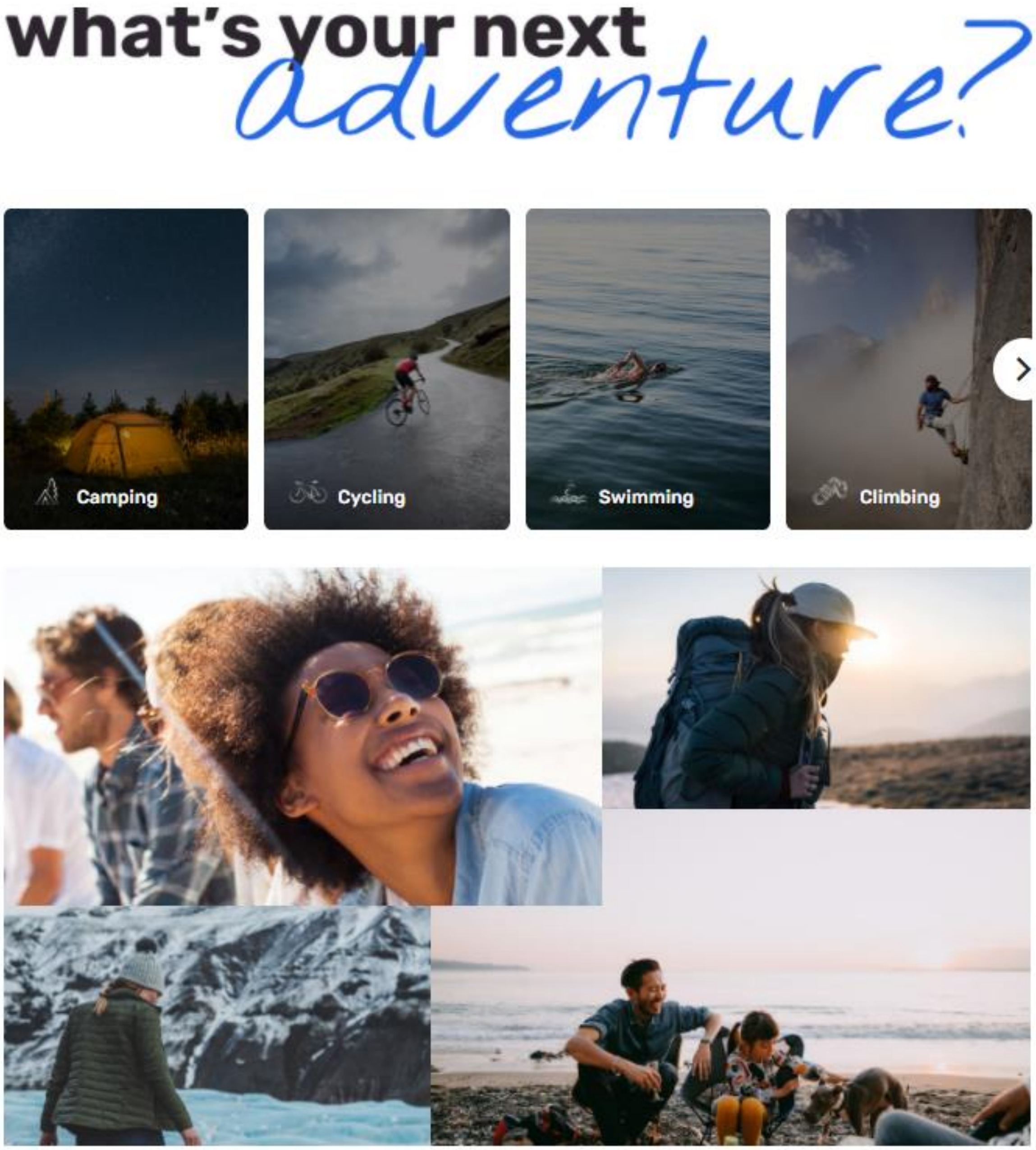


Version	Date	Notes

0.01	15/03/2020	Initial Draft
0.02	30/03/2020	Converting sections to Units and updated content
0.03	7/04/2020	Minor updates throughout doc
0.04	14/4/2020	Complete update of Unit 2
0.05	15/04/2020	Added Extensibility section on creating modules
0.06	6/5/2020	Added a bit more clarity on validating products.
0.07	28/07/2021	Adding additional information on Pick up times and slots.
0.08	10/09/2021	Added B2B options to the training

INTRODUCTION

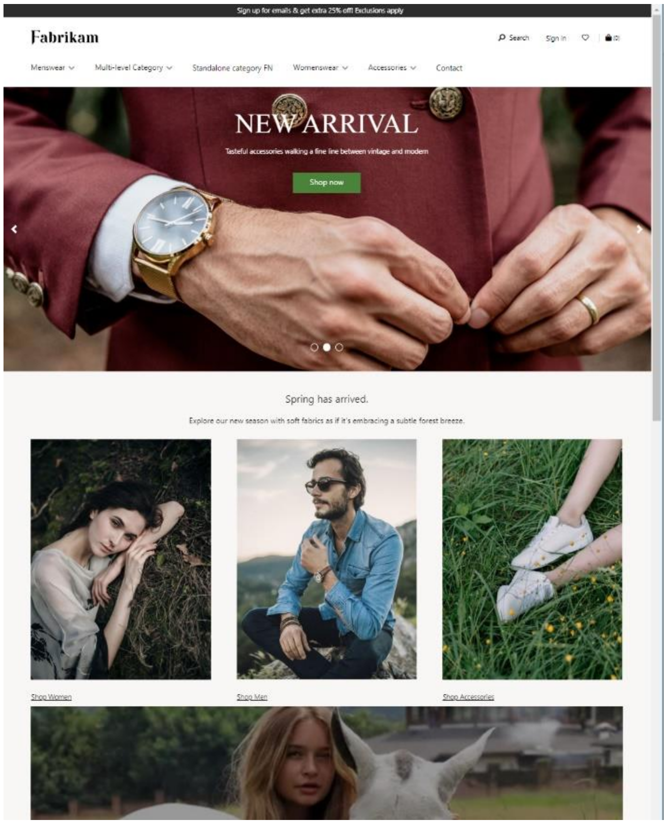
The Commerce workshop is meant to be a hands-on guide to making a branded e-Commerce site that could be used for general training or customer demos.

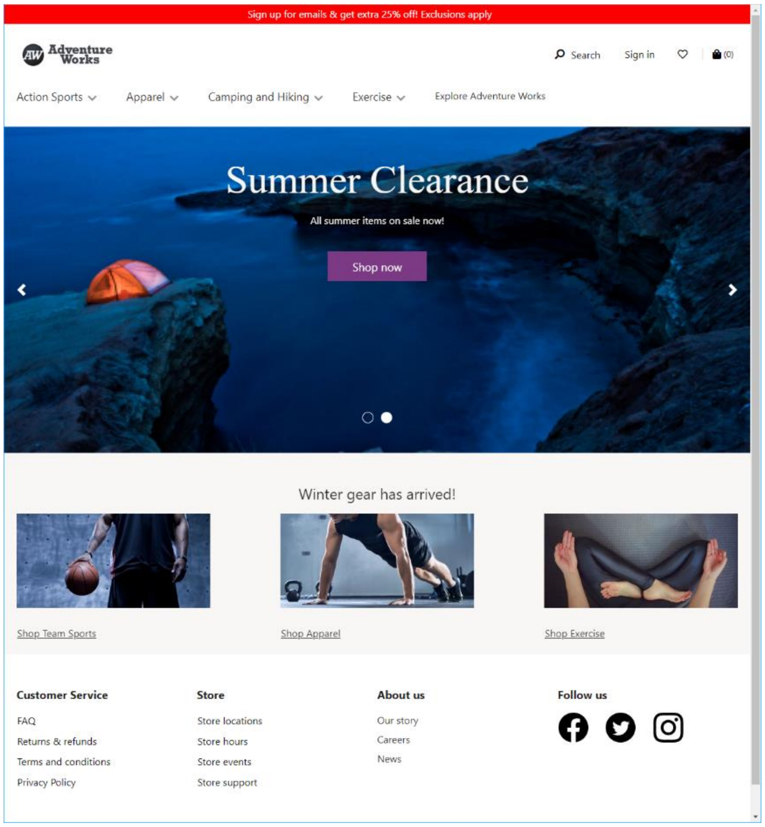
This document will cover the basics needed to get started with Dynamics 365 Commerce E-Commerce. It is not meant to be an exhaustive guide covering all the features of the Commerce product like merchandising, point of sale, etc... however the guide will touch on many features across the product.

This guide will provide examples of bringing up a fictitious sporting goods e-Commerce site called “Adventure Works” which will have its own set of products, navigation hierarchy and theme. This site is meant as an example only and can be replaced with any company / industry desired. Using the examples as a guide, you should be able to adjust to any company and set of products you desire.

When a Dynamics 365 Commerce environment is provisioned the option of starting with demo data can be selected. When this option is selected a starter “Fabrikam” fashion store is available with a pre-configured retail store (San Francisco), an e-Commerce site and a call center channel.

The goal of this exercise will be to turn the Fabrikam branded fashion store into the Adventure Works sporting goods branded store. The sporting goods store is just used as an example, feel free to use any ideas/brands you like as you follow the guide. We will also provide a zip file with Adventure Works assets if you choose to follow along with that brand.


Sign up for emails & get extra 25% off! Exclusions apply
Fabrikam
Search
Sign in
(0)
Menswear
Multi-level Category
Standalone category FN
Womenswear
Accessories
Contact
NEW ARRIVAL
Tasteful accessories walking a fine line between vintage and modern
Shop now
Spring has arrived.
Explore our new season with soft fabrics as if it's embracing a subtle forest breeze.
Shop Women
Shop Men
Shop Accessories



ASSUMPTIONS

The guide is meant as a getting started guide and no assumptions are made on the level of expertise on the product. Unfortunately, with the limited time for this exercise we do not have time to fully cover an overview of the tools being used.

The guide will assume you have access to a Dynamics 365 Commerce environment which already has the Commerce scale unit and e-Commerce environments initialized. The environments should be created with the demo data option which includes pre-defined channels, users, and customers. If you are using a production site without demo data, the instructions generally still apply but some differences may be needed including pre-setting up some settings such as business entities, warehouses, users, address books, etc… For more information, see the Channel setup prerequisites online doc.

OPTIONAL PREREQUISITES

- Optionally creating a B2C tenant will allow a full branded experience when signing in, otherwise during those screens the “Fabrikam” branded logo will appear. See the Set up a B2C tenant in Commerce doc for more information. This process can take some time, and thus do not recommend this unless step for this workshop

ADDITIONAL RESOURCES

- General Documentation

HIGH LEVEL OVERVIEW

In this workshop we will cover the following:

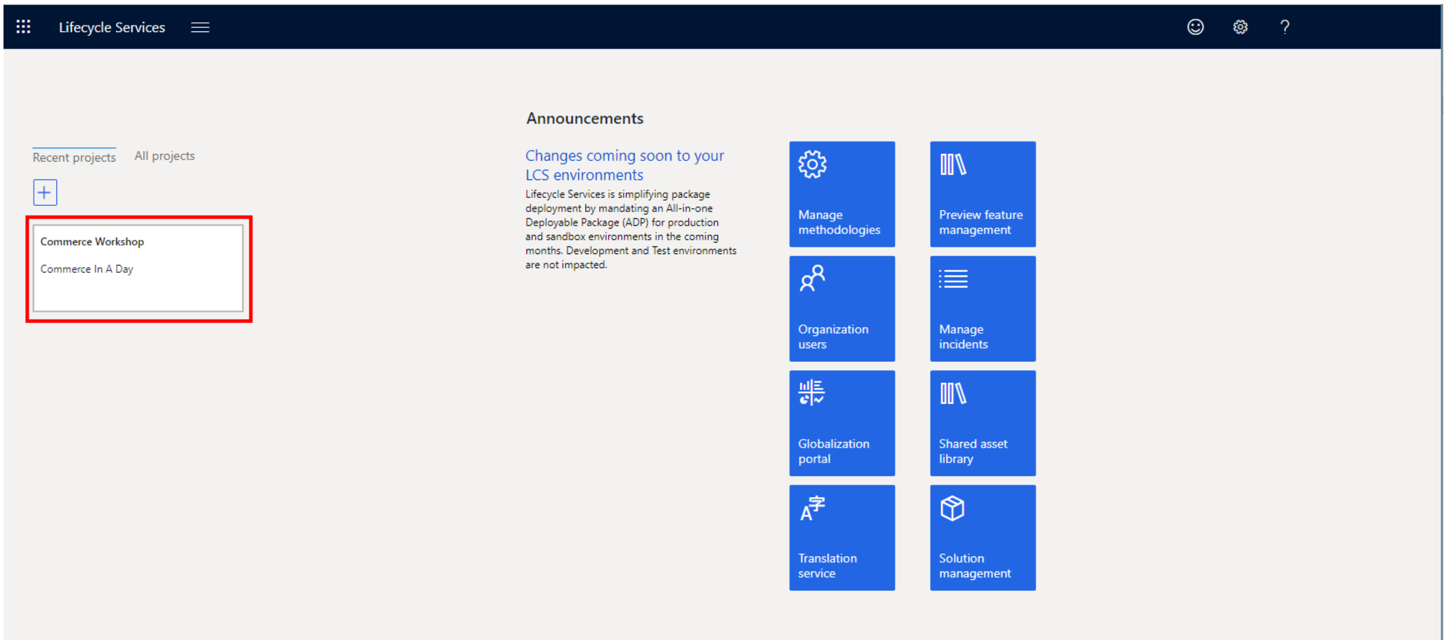
- Unit 1 - Setup and configure a new online channel and products in the Headquarters tool
    - Set up and configure a new online channel (payment provider, shipping, fulfillment groups,…)
    - Add new products (with variants) into new retail hierarchy nodes
    - Release new products, assort them to the new online channel and configure product attributes
    - Create a channel navigation hierarchy and associate with new channel
    - Configure channel refiners
    - Run appropriate jobs needed for Commerce Scale Unit support to see the new channel/products
- Unit 2 - Setup and configure an e-Commerce site using the site builder tool
    - Run the first time run experience (FRE) to create a new site based on new channel
    - Test the new site – new nav hierarchy and products with Fabrikam theme and marketing content
    - Tooling overview
    - Upload product images to the digital asset manager
    - Rebrand the header, footer, and homepage marketing content
    - Setup B2C tenant (optional)
    - CSS theme changes using theme overrides feature
- Unit 2 – E-Commerce Business to Business (B2B) configure new online channel
    - Set up and configure a new B2B online channel (payment provider, shipping, fulfillment groups,…)
    - Run the first time run experience (FRE) to create a new site based on new channel
    - Test the new site – new nav hierarchy and products with Fabrikam theme and marketing content
- Unit 4 – Business to Business site setup
    - Create new B2B site
    - Business Partner request, Account Management, Customer balance, Order template, Invoice pages
- Unit 5 – Azure Active directory B2C
    - Setup AAD B2C
- Unit 6 - E-Commerce extensibility theme and module development
    - Overview and development environment setup
    - Creating a custom module
    - Creating a custom theme
    - Testing theme
    - Build configuration extension package and deploying

TOOLS

The main tools we will be using in this guide include the Lifecycle Services (LCS) tool, Commerce Headquarters and the e-Commerce Site Builder tool.

You should already have access and a link to an LCS environment as a prerequisite. The Commerce Headquarters and the Site Builder tool can be launched from there. To get to these tools, sign into LCS and follow the below steps.

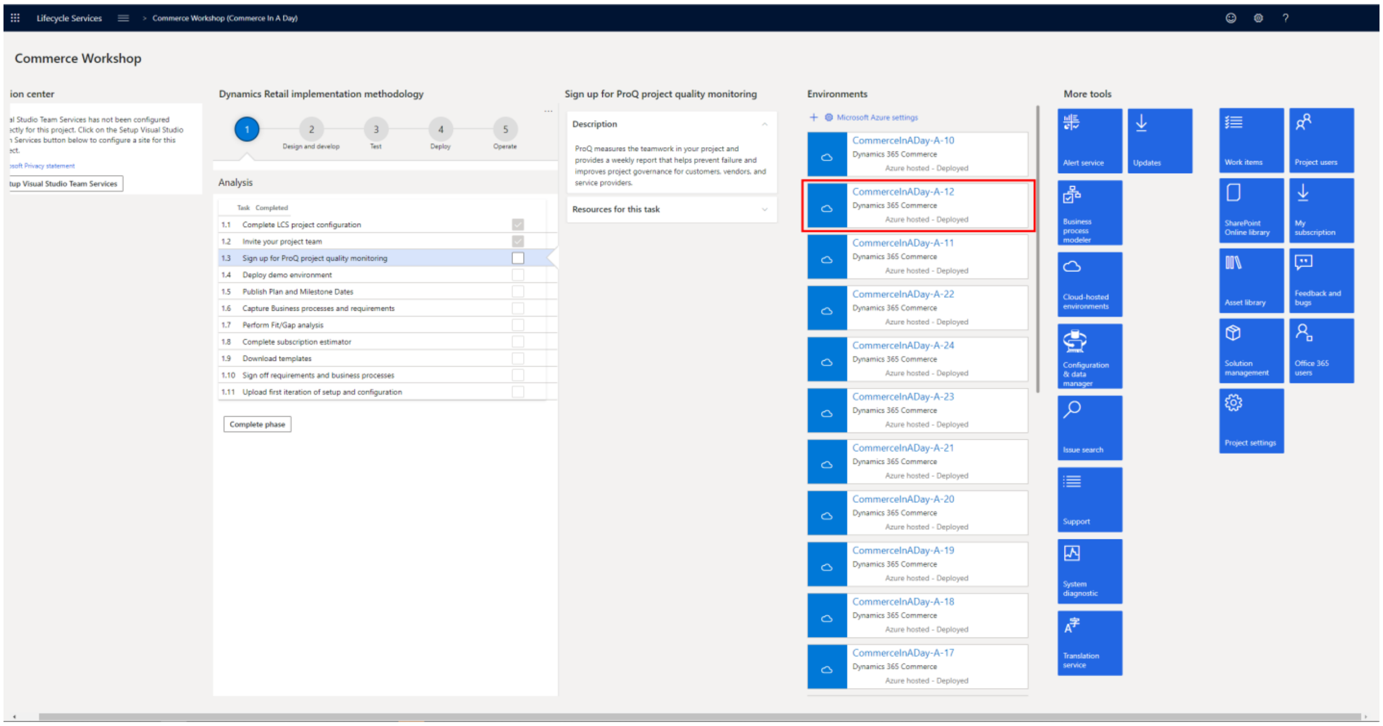
Start by selecting the environment you are working on.



Changes coming soon to your LCS environments

Lifecycle Services is simplifying package deployment by mandating an All-in-one Deployable Package (ADP) for production and sandbox environments in the coming months. Development and Test environments are not impacted.

Next, select the environment you are managing as in the below image.



Commerce Workshop

ion center

al Studio Team Services has not been configured ectly for this project. Click on the Setup Visual Studio n Services button below to configure a site for this ect.

Dynamics Retail implementation methodology

Analysis

Task		Completed
1.1	Complete LCS project configuration	☑
1.2	Invite your project team	☑
1.3	Sign up for ProQ project quality monitoring	☐
1.4	Deploy demo environment	☐
1.5	Publish Plan and Milestone Dates	☐
1.6	Capture Business processes and requirements	☐
1.7	Perform Fit/Gap analysis	☐
1.8	Complete subscription estimator	☐
1.9	Download templates	☐
1.10	Sign off requirements and business processes	☐
1.11	Upload first iteration of setup and configuration	☐

Sign up for ProQ project quality monitoring

Description

ProQ measures the teamwork in your project and provides a weekly report that helps prevent failure and improves project governance for customers, vendors, and service providers.

Resources for this task

Environments

CommerceInADay-A-10
Dynamics 365 Commerce
Azure hosted - Deployed

CommerceInADay-A-12
Dynamics 365 Commerce
Azure hosted - Deployed

CommerceInADay-A-11
Dynamics 365 Commerce
Azure hosted - Deployed

CommerceInADay-A-22
Dynamics 365 Commerce
Azure hosted - Deployed

CommerceInADay-A-24
Dynamics 365 Commerce
Azure hosted - Deployed

CommerceInADay-A-23
Dynamics 365 Commerce
Azure hosted - Deployed

CommerceInADay-A-21
Dynamics 365 Commerce
Azure hosted - Deployed

CommerceInADay-A-20
Dynamics 365 Commerce
Azure hosted - Deployed

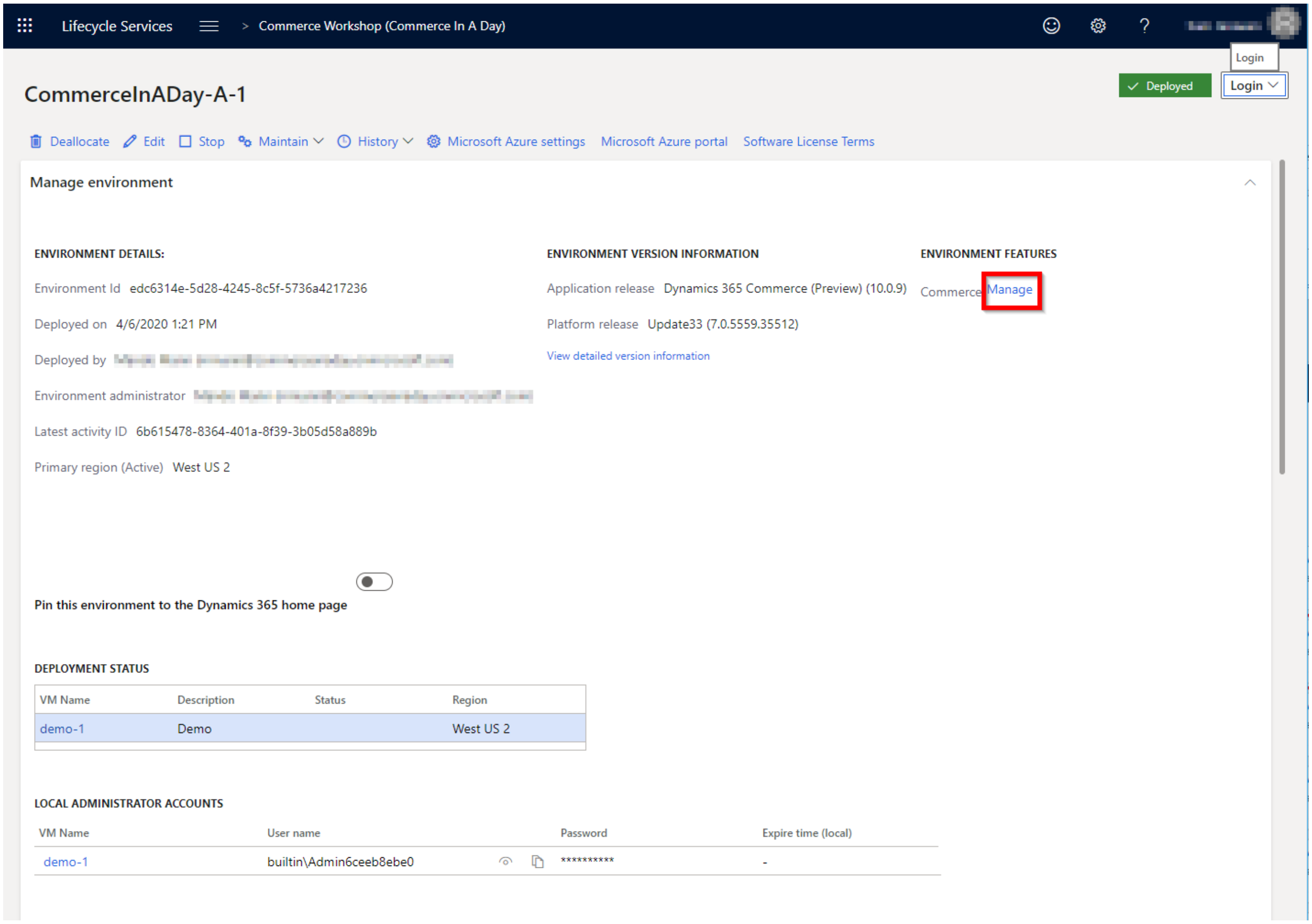
CommerceInADay-A-19
Dynamics 365 Commerce
Azure hosted - Deployed

CommerceInADay-A-18
Dynamics 365 Commerce
Azure hosted - Deployed

CommerceInADay-A-17
Dynamics 365 Commerce
Azure hosted - Deployed

More tools

Select the “**Manage**” link under the “ENVIRONMENT FEATURES” section.



CommerceInADay-A-1

Manage environment

ENVIRONMENT DETAILS:

Environment Id edc6314e-5d28-4245-8c5f-5736a4217236

Deployed on 4/6/2020 1:21 PM

Deployed by [redacted]

Environment administrator [redacted]

Latest activity ID 6b615478-8364-401a-8f39-3b05d58a889b

Primary region (Active) West US 2

ENVIRONMENT VERSION INFORMATION

Application release Dynamics 365 Commerce (Preview) (10.0.9)

Platform release Update33 (7.0.5559.35512)

View detailed version information

ENVIRONMENT FEATURES

Commerce Manage

Pin this environment to the Dynamics 365 home page

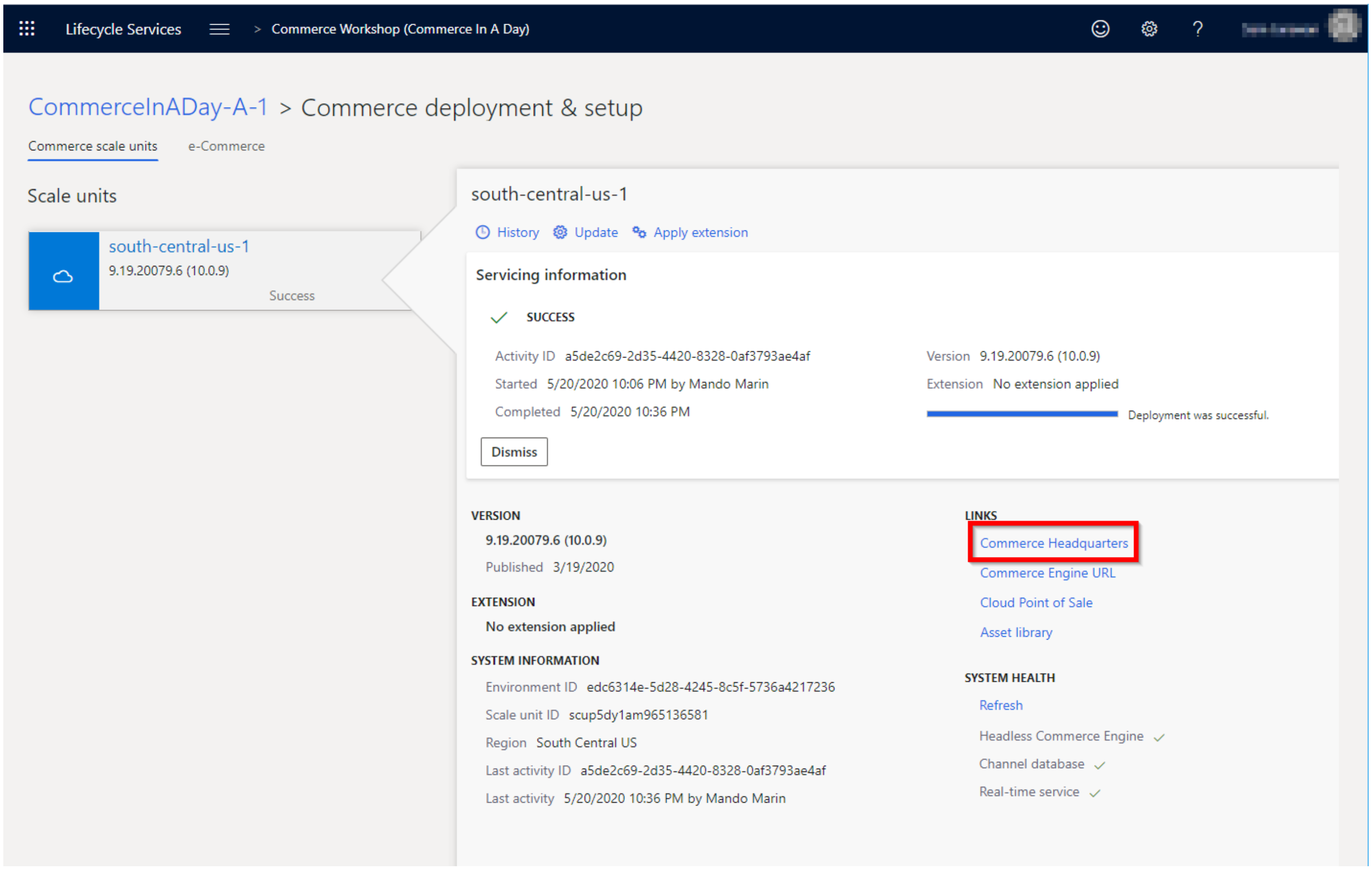
DEPLOYMENT STATUS

VM Name	Description	Status	Region
demo-1	Demo		West US 2

LOCAL ADMINISTRATOR ACCOUNTS

VM Name	User name	Password	Expire time (local)
demo-1	builtin\Admin6ceeb8ebe0	**********	-

To load the Commerce **Headquarters** tool, select the link as shown in the image below.



south-central-us-1
9.19.20079.6 (10.0.9)
Success

south-central-us-1

Servicing information

SUCCESS

Activity ID a5de2c69-2d35-4420-8328-0af3793ae4af
Started 5/20/2020 10:06 PM by Mando Marin
Completed 5/20/2020 10:36 PM

Version 9.19.20079.6 (10.0.9)
Extension No extension applied
Deployment was successful.

VERSION
9.19.20079.6 (10.0.9)
Published 3/19/2020

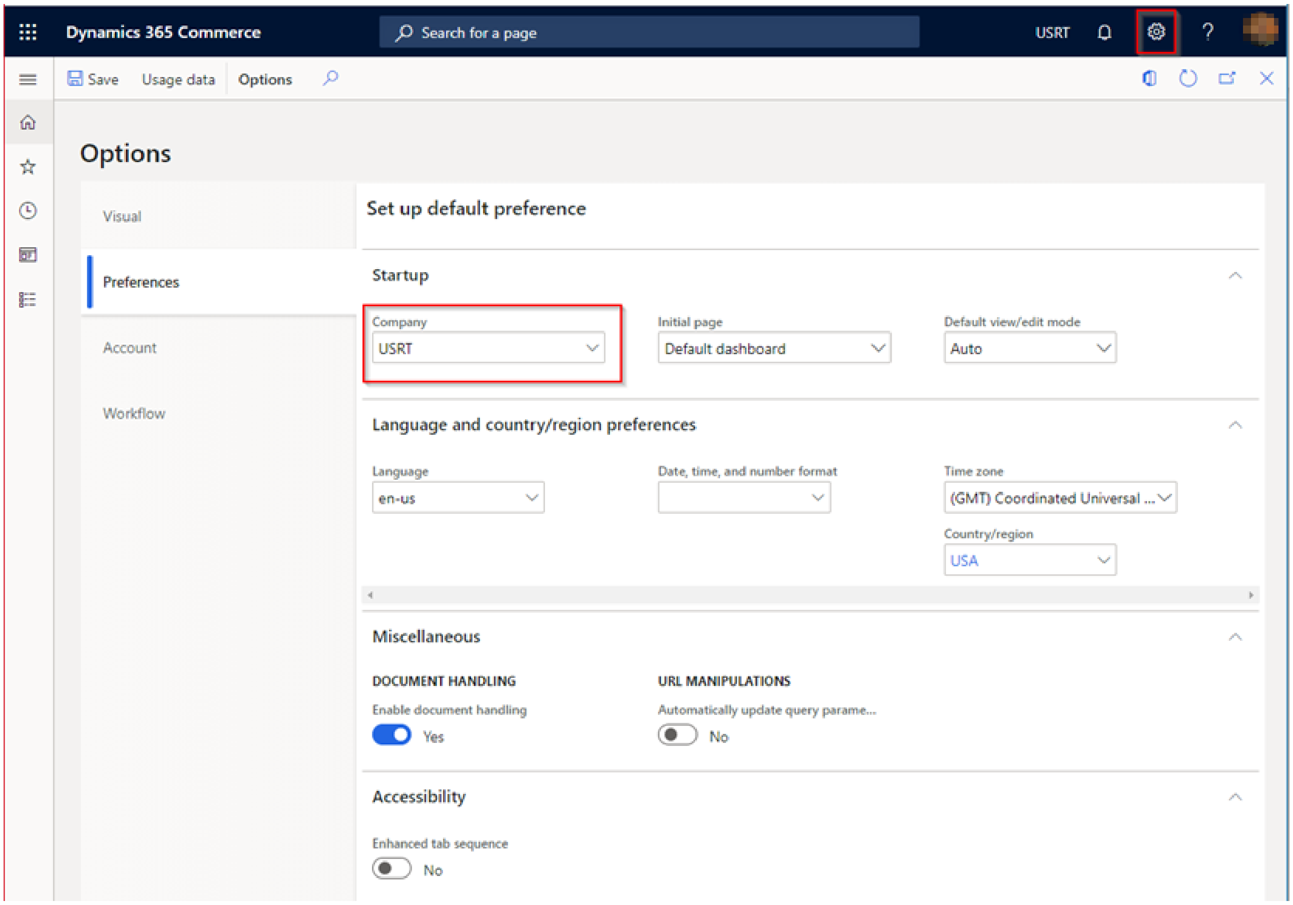
EXTENSION
No extension applied

SYSTEM INFORMATION
Environment ID edc6314e-5d28-4245-8c5f-5736a4217236
Scale unit ID scup5dy1am965136581
Region South Central US
Last activity ID a5de2c69-2d35-4420-8328-0af3793ae4af
Last activity 5/20/2020 10:36 PM by Mando Marin

LINKS

SYSTEM HEALTH
Headless Commerce Engine ✓
Channel database ✓
Real-time service ✓

Note: when using demo data you'll want to make sure you're using the **USRT**, to ensure HQ always launches with USRT you can optionally set it under the "User options" in the settings (gear at top of screen) as shown below.



Options

Set up default preference

Startup

Company: USRT

Initial page: Default dashboard

Default view/edit mode: Auto

Language and country/region preferences

Language: en-us

Date, time, and number format:

Time zone: (GMT) Coordinated Universal ...

Country/region: USA

Miscellaneous

DOCUMENT HANDLING

Enable document handling: Yes

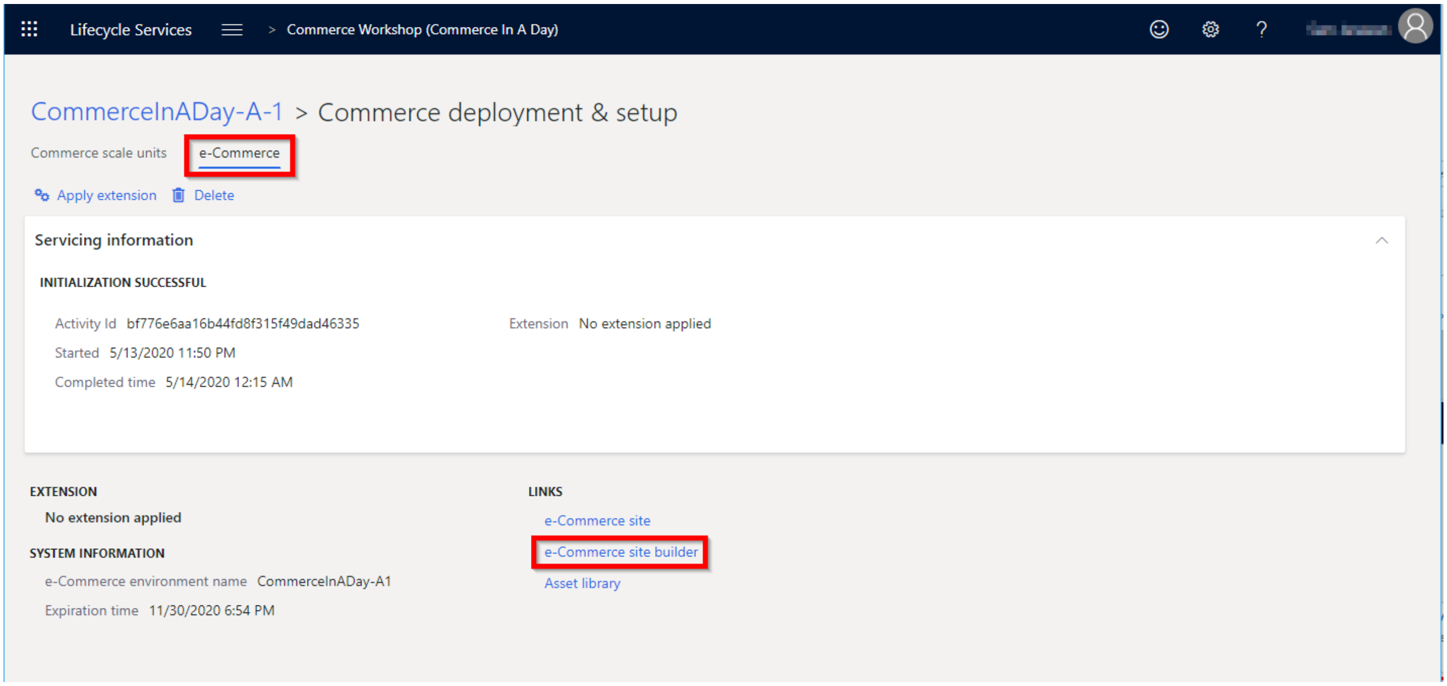
URL MANIPULATIONS

Automatically update query parame...: No

Accessibility

Enhanced tab sequence: No

To load the **Site Builder** tool, from **LCS,** select the "**e-Commerce**" tab at the top to load the e-Commerce **site builder** tool, select the appropriate link as shown below. You can also load the e-Commerce site from the appropriate link on this page.



CommerceInADay-A-1 > Commerce deployment & setup

Servicing information

INITIALIZATION SUCCESSFUL

Activity Id bf776e6aa16b44fd8f315f49dad46335

Extension No extension applied

Started 5/13/2020 11:50 PM

Completed time 5/14/2020 12:15 AM

EXTENSION

No extension applied

SYSTEM INFORMATION

e-Commerce environment name CommerceInADay-A1

Expiration time 11/30/2020 6:54 PM

LINKS

e-Commerce site

e-Commerce site builder

Asset library

UNIT 1 ONLINE STORE CHANNEL CONFIGURATION

ONLINE STORES

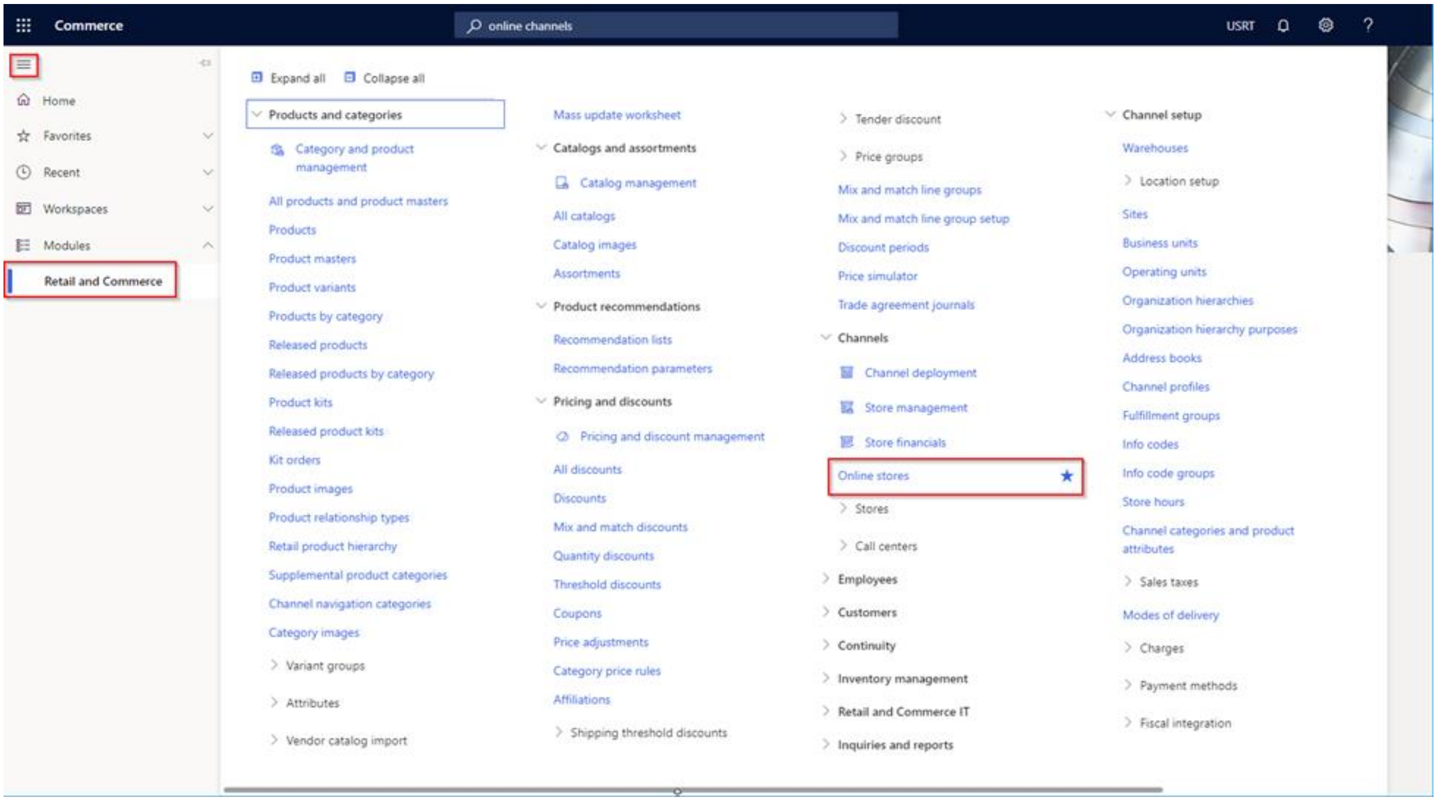
Dynamics 365 Commerce supports multiple channel types including **retail stores**, **call centers** and **online stores**. In a later step we will be using the Site Builder tool to initialize an e-Commerce site. As part of initialization it will need to associate to a Commerce online store channel. This section will cover how to create and configure an online store channel. We will also release and assort products to this channel and create a channel navigation hierarchy that will be shown on the e-Commerce site to navigate product categories.

To get started launch the **Commerce Headquarters** app.

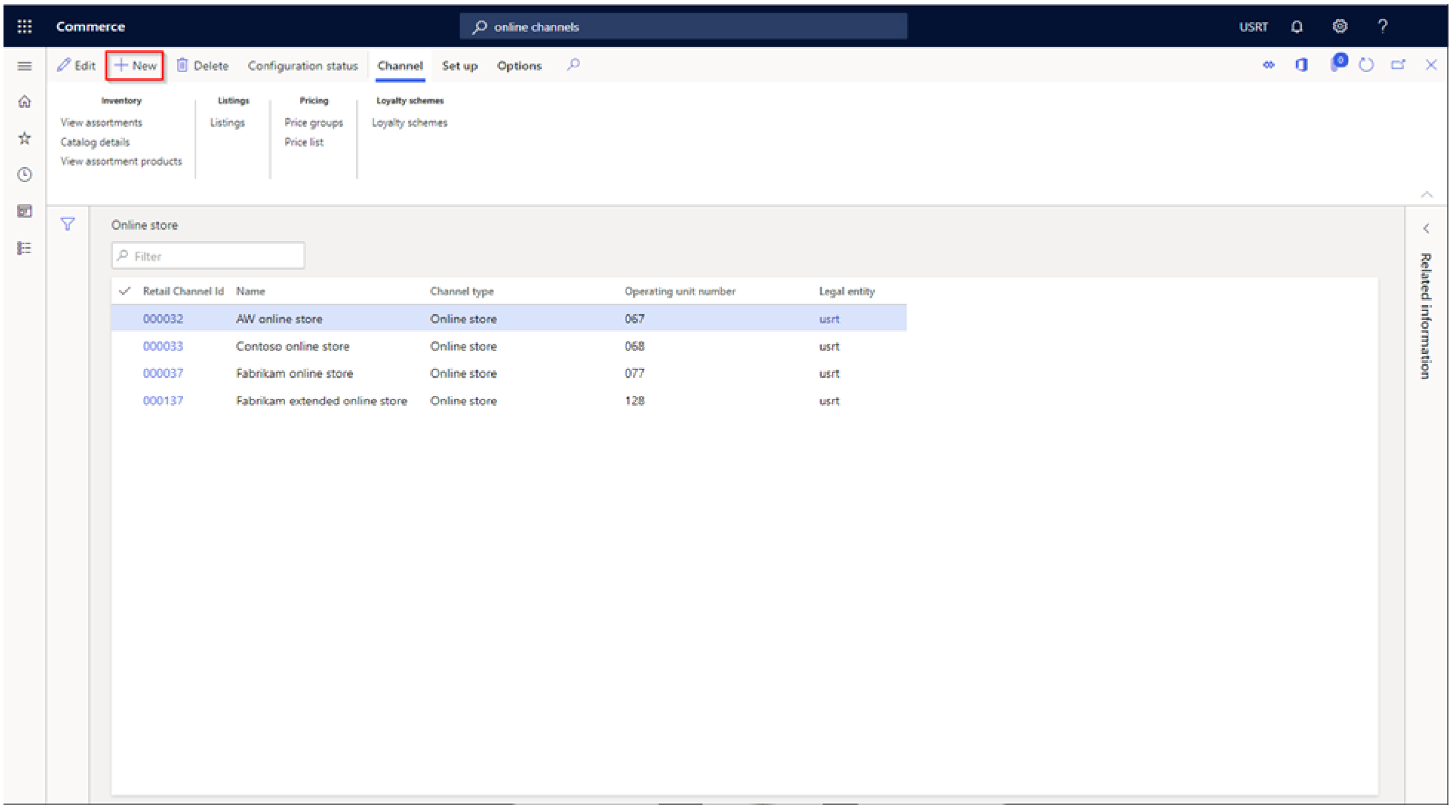
Please note: The demo environment you are using in this training has Adventure works demo data so just use another name. I.E AW Camping & Hiking.

CREATE AN ONLINE STORE CHANNEL IN HEADQUARTERS

From the "**Retail and Commerce**" menu, select "**Online stores**" (or alternatively search for "**Online stores**" in the search box at the top). Make sure "**USRT**" is selected at the top.



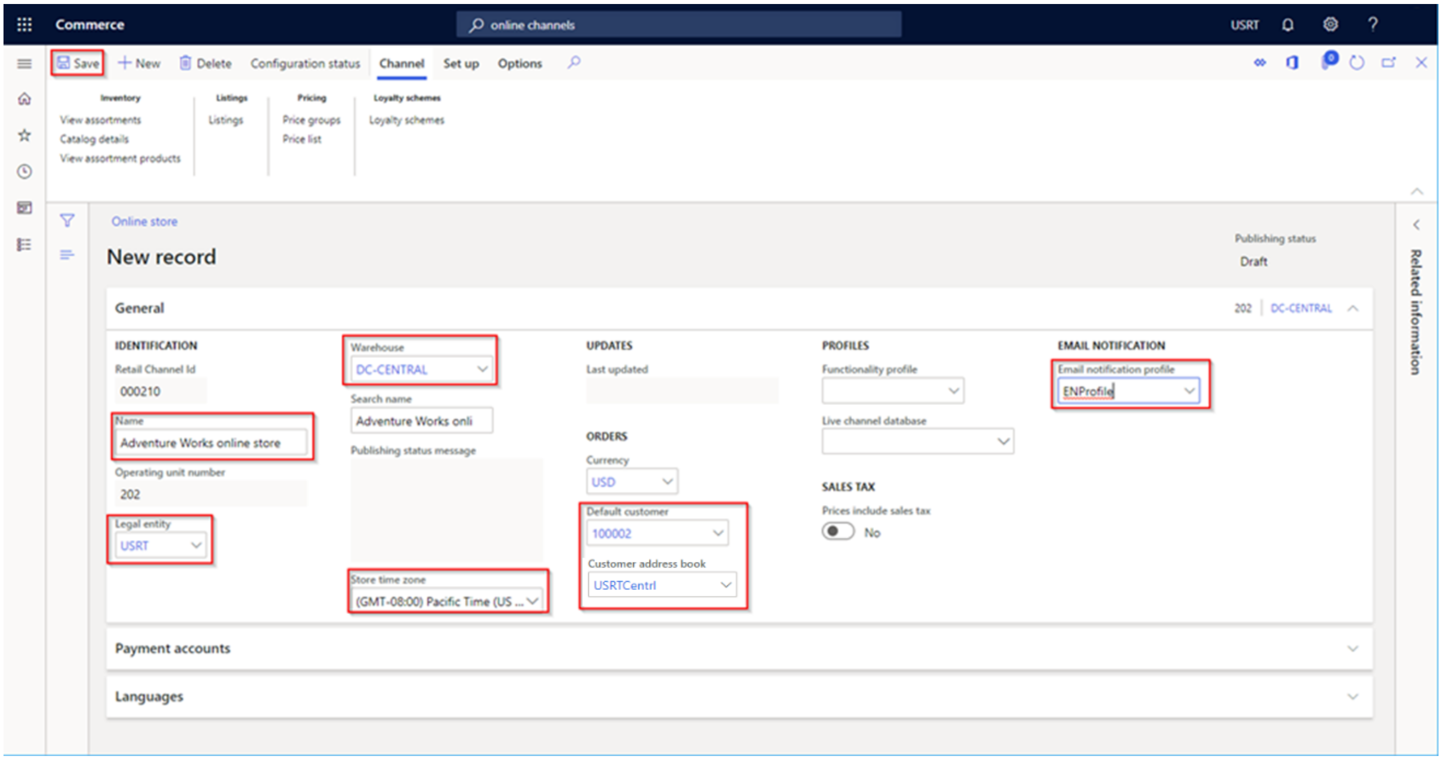
You should see a list of configured online stores if any exist, we will be adding a new one - select the “**New**” button.



Retail Channel Id	Name	Channel type	Operating unit number	Legal entity
000032	AW online store	Online store	067	usrt
000033	Contoso online store	Online store	068	usrt
000037	Fabrikam online store	Online store	077	usrt
000137	Fabrikam extended online store	Online store	128	usrt

Configure the following items for the new channel:

- **Name** – Provide the name for the store (example: “Adventure Works online store”)
- **Legal entity** – Select the legal entity the store will belong to (this will be “**USRT**” in the evaluation environment)
- **Warehouse** – Select the warehouse that will serve the online store (**DC-CENTRAL** can be used in the evaluation environment)
    - For basic warehouse setup see the online Warehouse set up document.
- **Store time zone** – Select the default time zone used for the store, this may map to the corporation’s headquarters time zone
- **Default customer** – This customer may need to be pre-configured, all sales orders through the site for anonymous users will be tied to this account in AX (**100002** can be used in the evaluation environment)
    - For more info see the online Create a default customer doc.
    - **Note**: for any new e-commerce customer account signups the default properties are copied from this account
    - **Note**: If you are creating a new default customer, ensure that the default customer is of a type “Person” and not “Organization”
- **Customer address book** – Set the address book, this should be pre-configured before creating the channel (**USRTCentrl** can be used from within the evaluation environment)
    - For more info see the online Create a default customer doc which covers creating an address book.
    - **Note**: The default address book must be associated with the default customer provided above
- **Email notifications** – Select the email notification profile, this may need to be pre-created before creating the channel (**ENProfile** can be used if you are using the evaluation environment)
    - For more information see the online Set up an email notification profile document.
- Select the “**Save**” button.



Online store

New record

Publishing status: Draft

General — 202 | DC-CENTRAL

IDENTIFICATION

Retail Channel Id: 000210

Name: Adventure Works online store

Operating unit number: 202

Legal entity: USRT

Warehouse: DC-CENTRAL

Search name: Adventure Works onli

Publishing status message:

Store time zone: (GMT-08:00) Pacific Time (US ...

UPDATES

Last updated:

ORDERS

Currency: USD

Default customer: 100002

Customer address book: USRTCentrl

PROFILES

Functionality profile:

Live channel database:

SALES TAX

Prices include sales tax: No

EMAIL NOTIFICATION

Email notification profile: ENProfile

SETUP PAYMENT ACCOUNTS

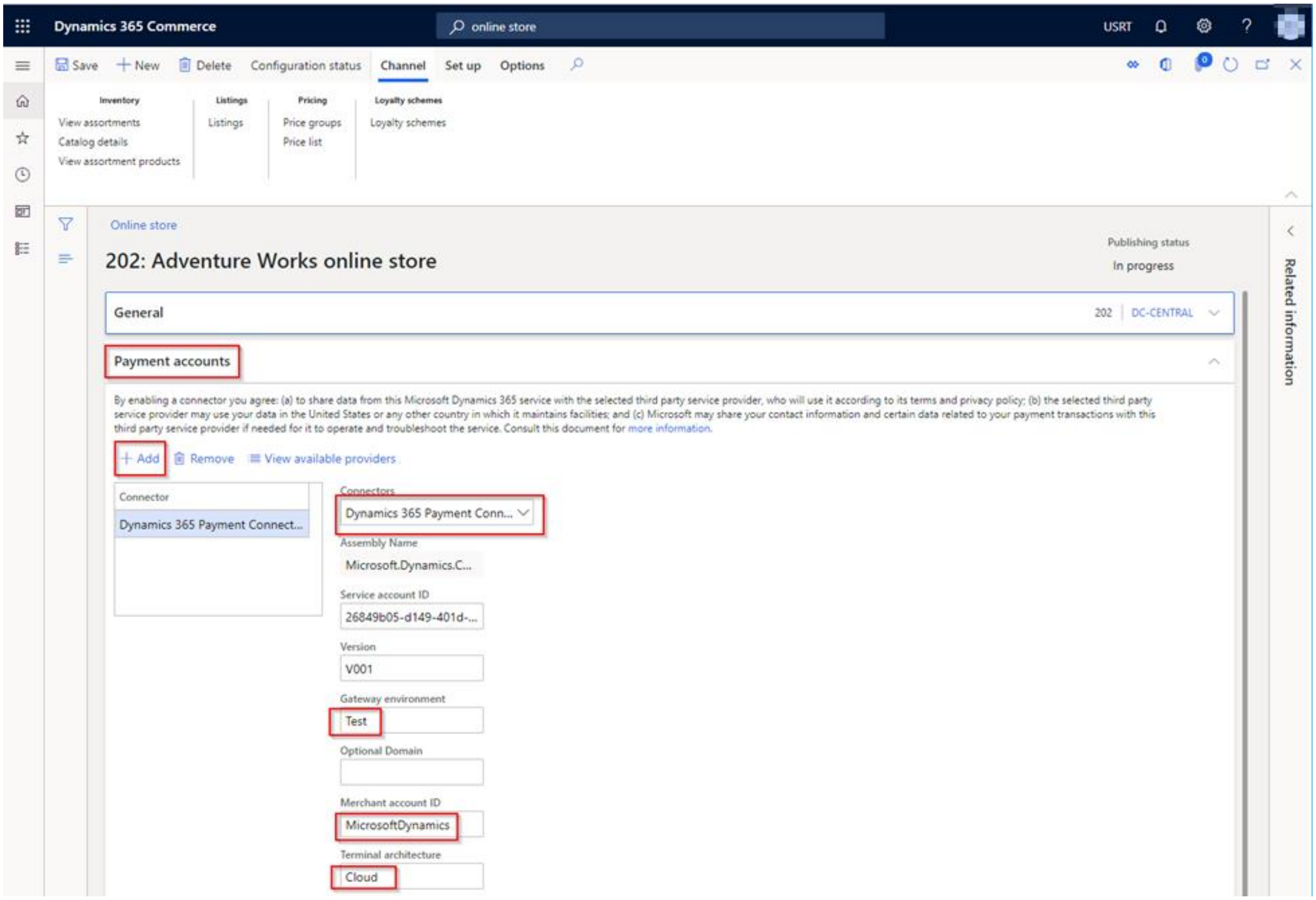
We will now setup the Adyen payment account needed to accept payments on the e-Commerce site. This section will guide you through setting up a payment connector for the Online site.

- Expand the "**Payment accounts**" section and select "**Add**" to add a payment connector.

Online channels are designed to be used out of the box with the **Dynamics 365 Payment Connector for Adyen.**

Configuration Steps **(Note: The Cloud API key will need to be supplied by your instructor for this training)**. If you plan to set this up in your own environment you will need an Adyen test account.

- Change "**Gateway Environment**" to '**Test'**
- Set the "**Merchant account ID**"
- Set "**Terminal Architecture**" to "**Cloud**"
- Change "**Allow saving payment information in e-Commerce**" to "**True"**
- All other values can be left with the defaults
- Click "**Save**"



202: Adventure Works online store

Publishing status
In progress

General 202 | DC-CENTRAL

Payment accounts

By enabling a connector you agree: (a) to share data from this Microsoft Dynamics 365 service with the selected third party service provider, who will use it according to its terms and privacy policy; (b) the selected third party service provider may use your data in the United States or any other country in which it maintains facilities; and (c) Microsoft may share your contact information and certain data related to your payment transactions with this third party service provider if needed for it to operate and troubleshoot the service. Consult this document for more information.

Connector
Dynamics 365 Payment Connect...

Connectors: Dynamics 365 Payment Conn...

Assembly Name: Microsoft.Dynamics.C...

Service account ID: 26849b05-d149-401d-...

Version: V001

Gateway environment: Test

Optional Domain:

Merchant account ID: MicrosoftDynamics

Terminal architecture: Cloud

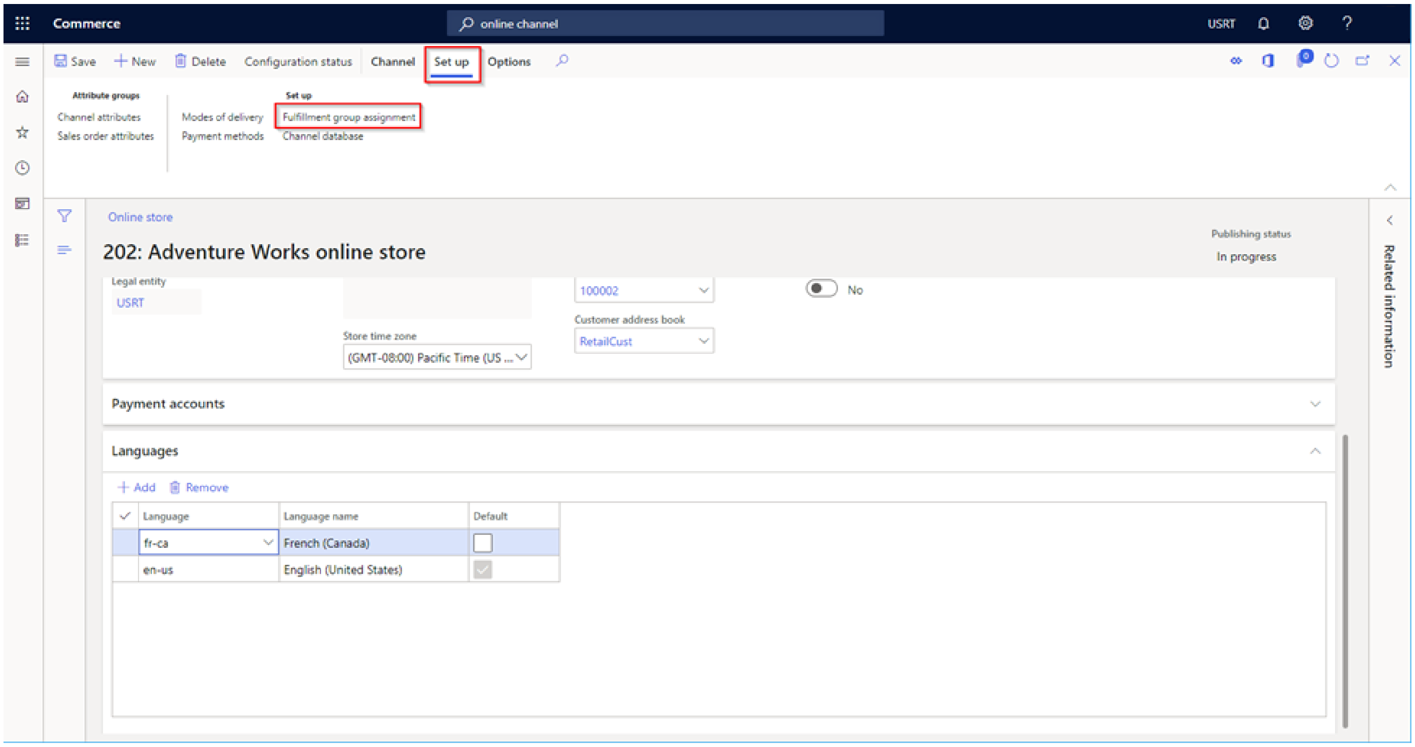
SETUP LANGUAGES

If you need to support multiple languages on your e-Commerce site, you first need to configure additional languages on the online store channel. For this workshop we will use en-us only, see appendix on how to add additional languages.

FULFILLMENT GROUP ASSIGNMENTS

In order for BOPIS (buy online, pickup in store) scenarios to work, you'll need to associate the channel with stores. Demo data already has a group of stores created called "All stores" that can be used here to ensure all stores retail stores show up when attempting the BOPIS scenario on the e-Commerce site. This section will show you how to set this up.

From within the selected online store, select the "**Setup**" tab at the top followed by the "**Fulfillment group assignment**" link.



Online store

202: Adventure Works online store

Publishing status: In progress

Legal entity: USRT

100002

No

Store time zone: (GMT-08:00) Pacific Time (US ...

Customer address book: RetailCust

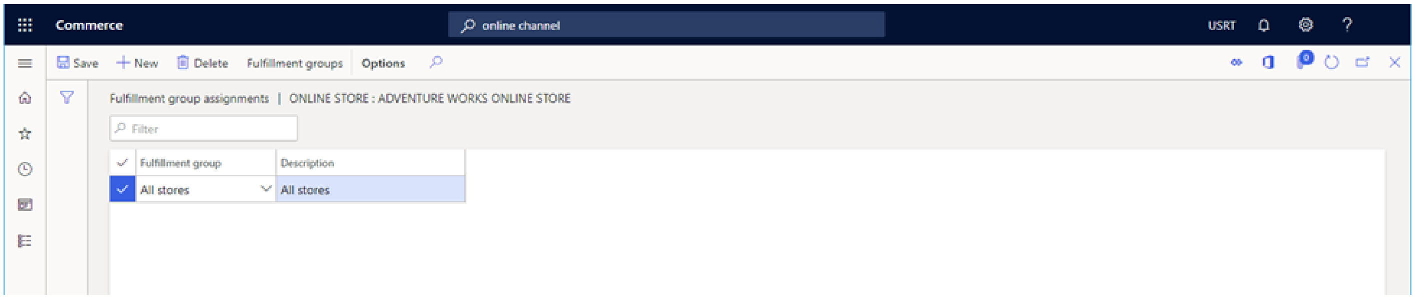
Payment accounts

Languages

+ Add   Remove

	Language	Language name	Default
	fr-ca	French (Canada)	□
	en-us	English (United States)	☑

Add "**All stores**" and close the form to get back to the previous channel page. You can also pick explicit stores here if desired.



Fulfillment group assignments | ONLINE STORE : ADVENTURE WORKS ONLINE STORE

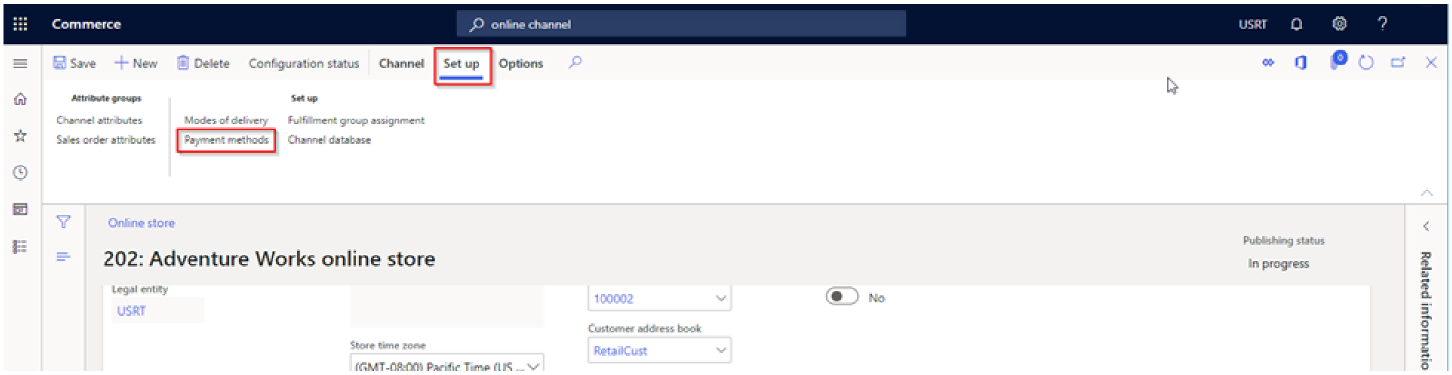
	Fulfillment group	Description
✓	All stores	All stores

SETUP PAYMENT METHODS

Each channel can accept various payment methods. In this section we will configure the online store channel to accept, loyalty cards, credit cards and gift cards. Choose the ones that are applicable to your needs.

Please note: **Card Setup** has been renamed to **Electronic payment setup**

From within the selected online store channel, select the "**Payment method**" button under the "**Setup**" tab.



Online store

202: Adventure Works online store

Legal entity: USRT

100002

No

Customer address book: RetailCust

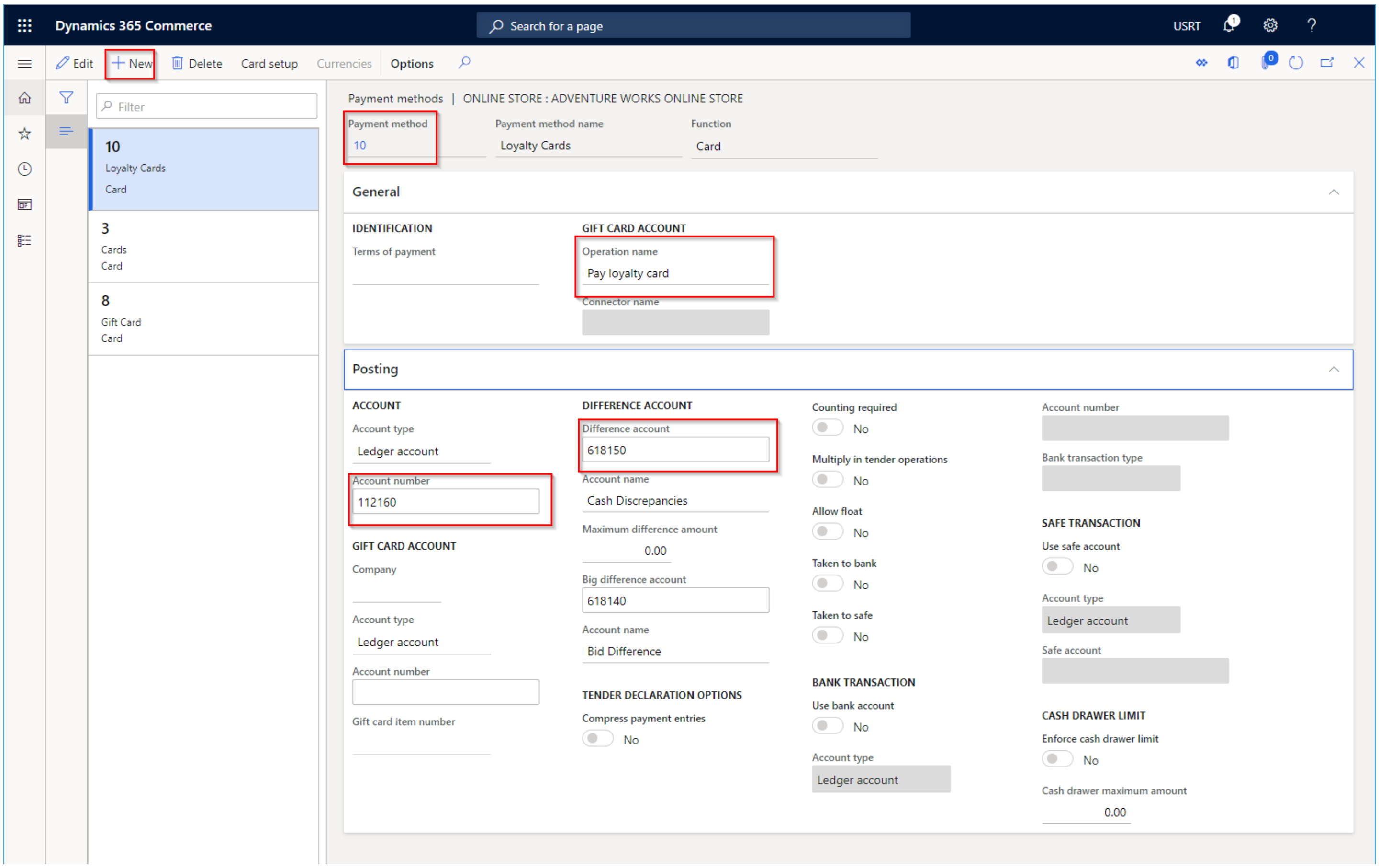
Store time zone: (GMT-08:00) Pacific Time (US ...

Publishing status: In progress

Click the new button to setup the below payment types. For each type you'll be adding account numbers that reference accounts setup in the demo data for posting purposes.

LOYALTY CARD

- Payment method: **10**
- In the "General" section set the Operation name to "**Pay loyalty card**"
- In the "Posting" section set
  - Under "ACCOUNT" section set Account number to **112160** (or other if not using demo data)
  - Under "DIFFERENCE ACCOUNT" section
    - Set difference account to **618150** (or other if not using demo data)
    - Set Big difference account to **618140**
- Select "**Save**"



10
Loyalty Cards
Card

3
Cards
Card

8
Gift Card
Card

Payment methods | ONLINE STORE : ADVENTURE WORKS ONLINE STORE

Payment method: 10
Payment method name: Loyalty Cards
Function: Card

General

IDENTIFICATION
Terms of payment:

GIFT CARD ACCOUNT
Operation name: Pay loyalty card
Connector name:

Posting

ACCOUNT
Account type: Ledger account
Account number: 112160

GIFT CARD ACCOUNT
Company:
Account type: Ledger account
Account number:
Gift card item number:

DIFFERENCE ACCOUNT
Difference account: 618150
Account name: Cash Discrepancies
Maximum difference amount: 0.00
Big difference account: 618140
Account name: Bid Difference

TENDER DECLARATION OPTIONS
Compress payment entries: No

Counting required: No
Multiply in tender operations: No
Allow float: No
Taken to bank: No
Taken to safe: No

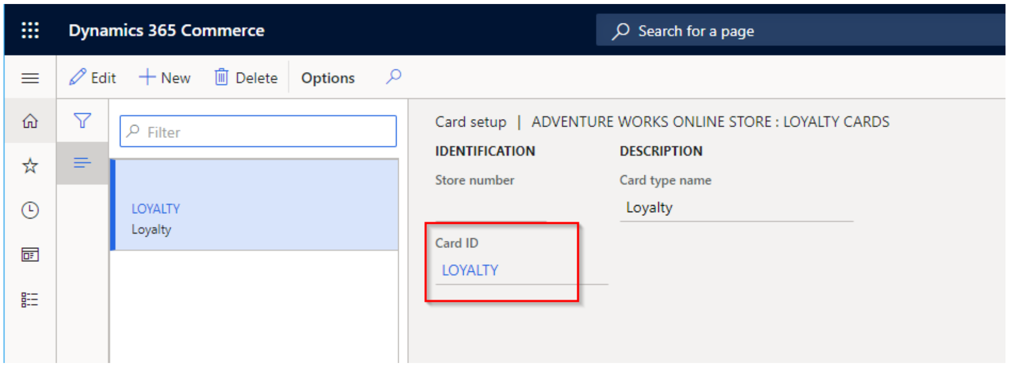
BANK TRANSACTION
Use bank account: No
Account type: Ledger account

Account number:
Bank transaction type:

SAFE TRANSACTION
Use safe account: No
Account type: Ledger account
Safe account:

CASH DRAWER LIMIT
Enforce cash drawer limit: No
Cash drawer maximum amount: 0.00

- Select "**Card Setup**" then "**New**"
- Set "**Card ID**" to "**LOYALTY**"
- Save and close to go back to previous screen



LOYALTY
Loyalty

Card setup | ADVENTURE WORKS ONLINE STORE : LOYALTY CARDS

IDENTIFICATION

Store number

Card ID
LOYALTY

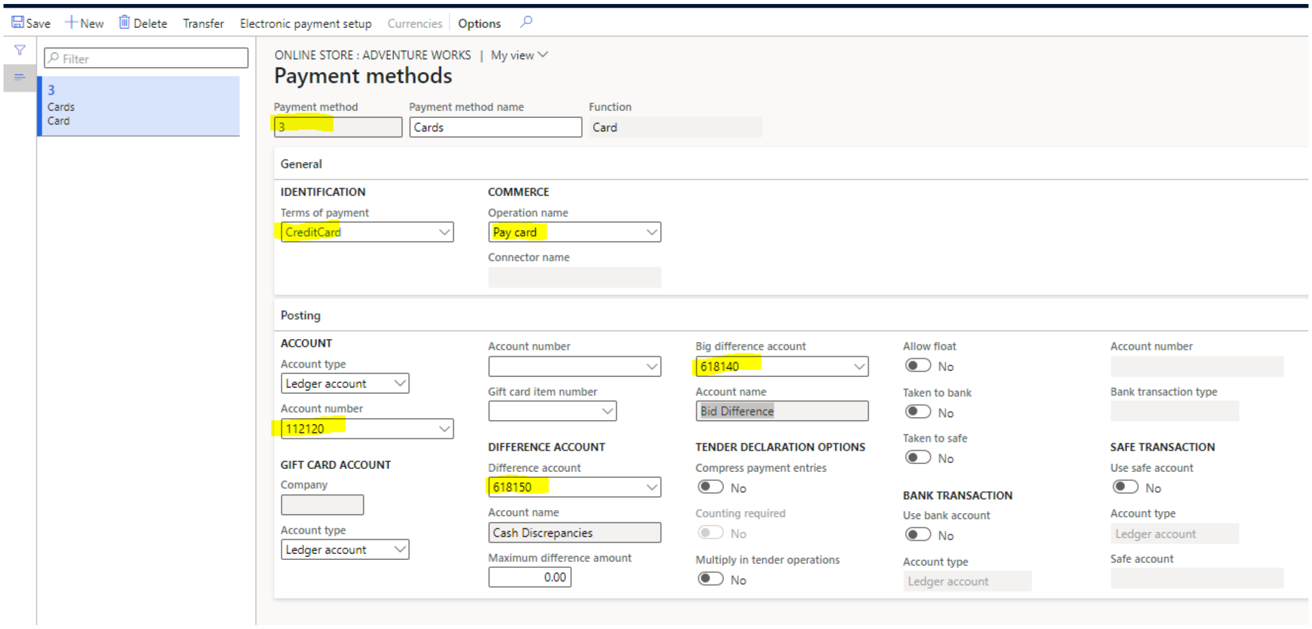
DESCRIPTION

Card type name
Loyalty

“CREDIT CARDS”

We’ll now add another card type. Select “New” again.

- Set “Payment method” to **3**
- In the “General” section set
  - Terms of payment: “**CreditCard**”
  - Operation name “**Pay card**”
- In the Posting section set
  - In the “ACCOUNT” section set Account number to **112120** (or other if not using demo data)
  - Difference account: **618150** (or other if not using demo data)
  - Big difference account: **618140**
- Click **Save**



3
Cards
Card

ONLINE STORE : ADVENTURE WORKS | My view

Payment methods

Payment method: 3
Payment method name: Cards
Function: Card

General

IDENTIFICATION
Terms of payment: CreditCard

COMMERCE
Operation name: Pay card
Connector name:

Posting

ACCOUNT
Account type: Ledger account
Account number: 112120

GIFT CARD ACCOUNT
Company:
Account type: Ledger account

Account number:
Gift card item number:

DIFFERENCE ACCOUNT
Difference account: 618150
Account name: Cash Discrepancies
Maximum difference amount: 0.00

Big difference account: 618140
Account name: Bid Difference

TENDER DECLARATION OPTIONS
Compress payment entries: No
Counting required: No
Multiply in tender operations: No

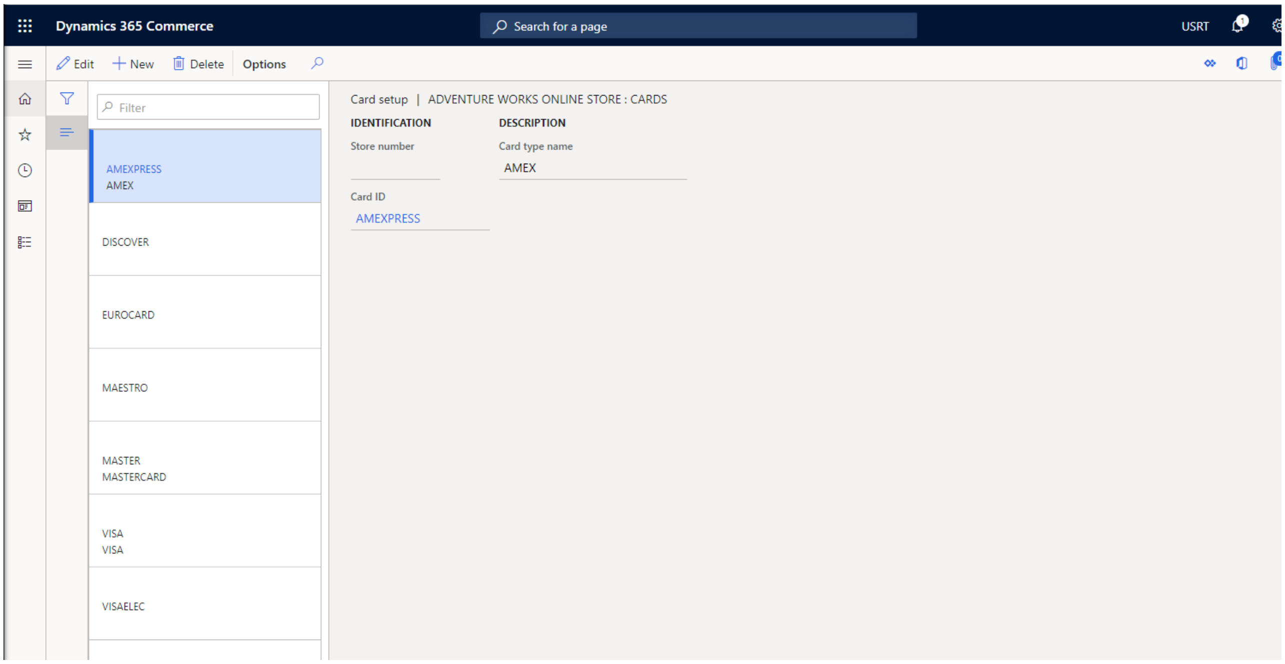
Allow float: No
Taken to bank: No
Taken to safe: No

BANK TRANSACTION
Use bank account: No
Account type: Ledger account

Account number:
Bank transaction type:

SAFE TRANSACTION
Use safe account: No
Account type: Ledger account
Safe account:

- Select “**Electronic payment setup**” and add cards to support, you should only need to set the “**Card ID**” for each, example add any of the following: AMEXPRESS, EUROCARD, MAESTRO, MASTER, VISA, VISAELEC (used for debit). Please include “Visa” to ensure the visa test card will work in Unit 2.



Card setup | ADVENTURE WORKS ONLINE STORE : CARDS

IDENTIFICATION

Store number:

Card ID: AMEXPRESS

DESCRIPTION

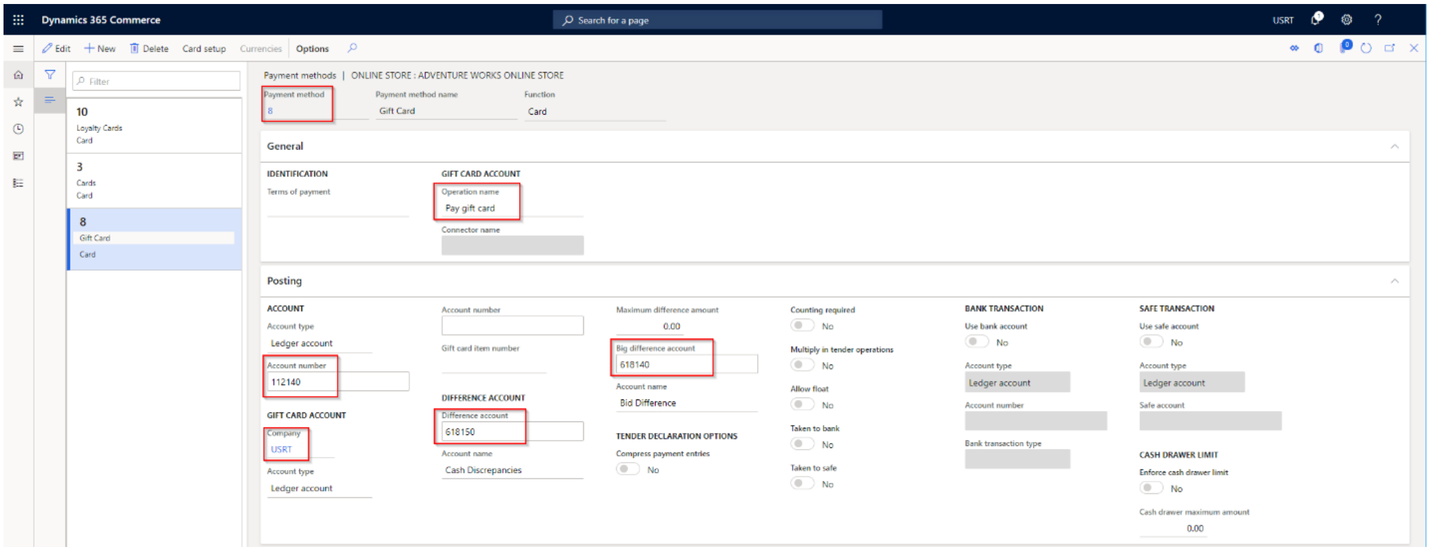
Card type name: AMEX

AMEXPRESS	AMEX
DISCOVER	
EUROCARD	
MAESTRO	
MASTER	MASTERCARD
VISA	VISA
VISAELEC	

“GIFT CARD”

We will now add gift card support. From the Payment methods page select “New” as done previously.

- Select Payment method: **8**
- Set Function to “Card”
- In the “General” section
    - Set “Operation name” to “**Pay gift card”**
- In the Posting section
    - Under “ACCOUNT” set Account number to **112140**
    - Gift Card Account company "**USRT**"
    - Difference account: **618150**
    - Big difference account: **618140**
- Click **Save**
- **Note: no card setup is required for this and should be disabled**



Payment methods | ONLINE STORE : ADVENTURE WORKS ONLINE STORE

Payment method: 8
Payment method name: Gift Card
Function: Card

General

IDENTIFICATION
Terms of payment:

GIFT CARD ACCOUNT
Operation name: Pay gift card
Connector name:

Posting

ACCOUNT
Account type: Ledger account
Account number: 112140

GIFT CARD ACCOUNT
Company: USRT
Account type: Ledger account

Account number:
Gift card item number:

DIFFERENCE ACCOUNT
Difference account: 618150
Account name: Cash Discrepancies

Maximum difference amount: 0.00
Big difference account: 618140
Account name: Bid Difference

TENDER DECLARATION OPTIONS
Compress payment entries: No

Counting required: No
Multiply in tender operations: No
Allow float: No
Taken to bank: No
Taken to safe: No

BANK TRANSACTION
Use bank account: No
Account type: Ledger account
Account number:
Bank transaction type:

SAFE TRANSACTION
Use safe account: No
Account type: Ledger account
Safe account:

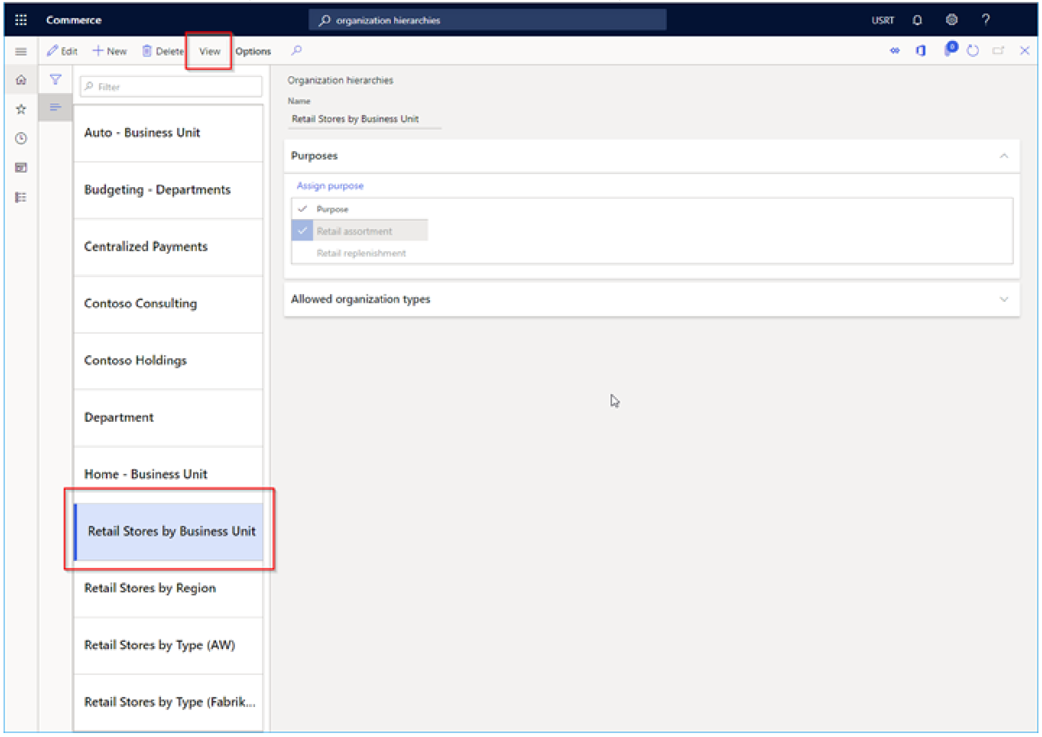
CASH DRAWER LIMIT
Enforce cash drawer limit: No
Cash drawer maximum amount: 0.00

ADD CHANNEL TO ORG HIERARCHY

Business units can be used to organize channels for simpler configurations. For example, instead of configuring modes of delivery for each store independently, or assorting products per channel, they can be done by org structure units.

In this section we will add our new channel to an existing organization hierarchy business unit that is included as part of the demo data. You may want to create a new "Business Unit" for your channel and below steps can be used as a general guide.

- Search for the "**Organization hierarchies"** page
  - Select "**Retail Stores by Business Unit**" and "**View**" on top



Auto - Business Unit
Budgeting - Departments
Centralized Payments
Contoso Consulting
Contoso Holdings
Department
Home - Business Unit
Retail Stores by Business Unit
Retail Stores by Region
Retail Stores by Type (AW)
Retail Stores by Type (Fabrik...

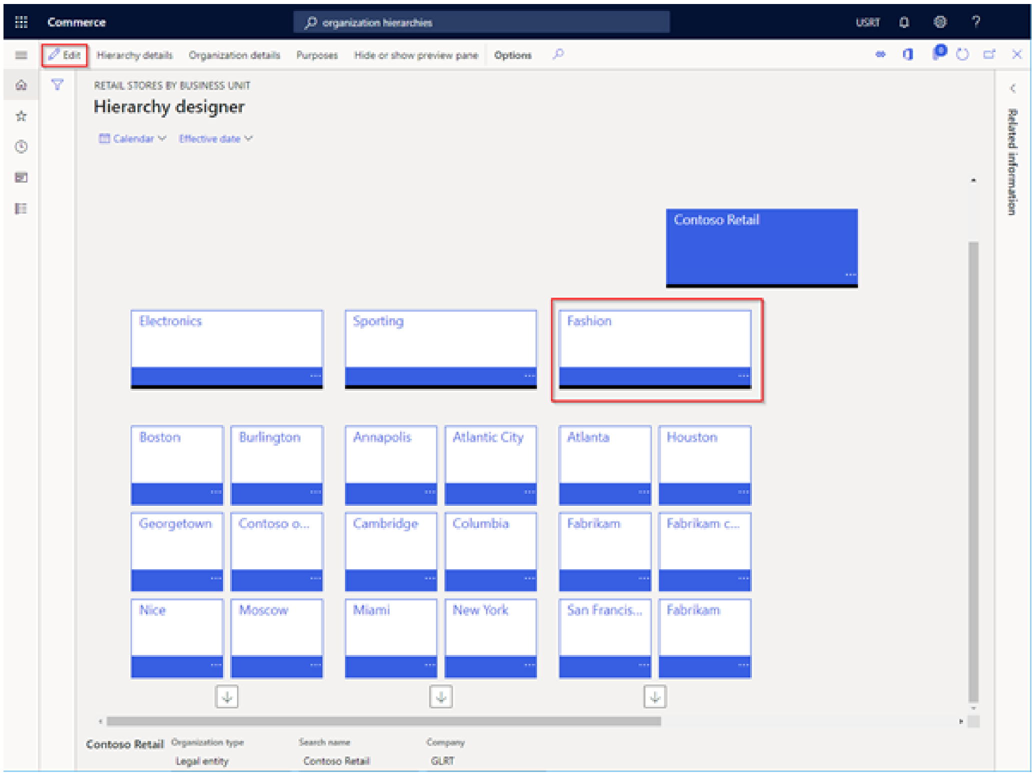
Organization hierarchies
Name
Retail Stores by Business Unit

Purposes

✓	Purpose
✓	Retail assortment
	Retail replenishment

Allowed organization types

  - Select "**Edit**" then the "**Fashion**" by hitting the "…" on fashion

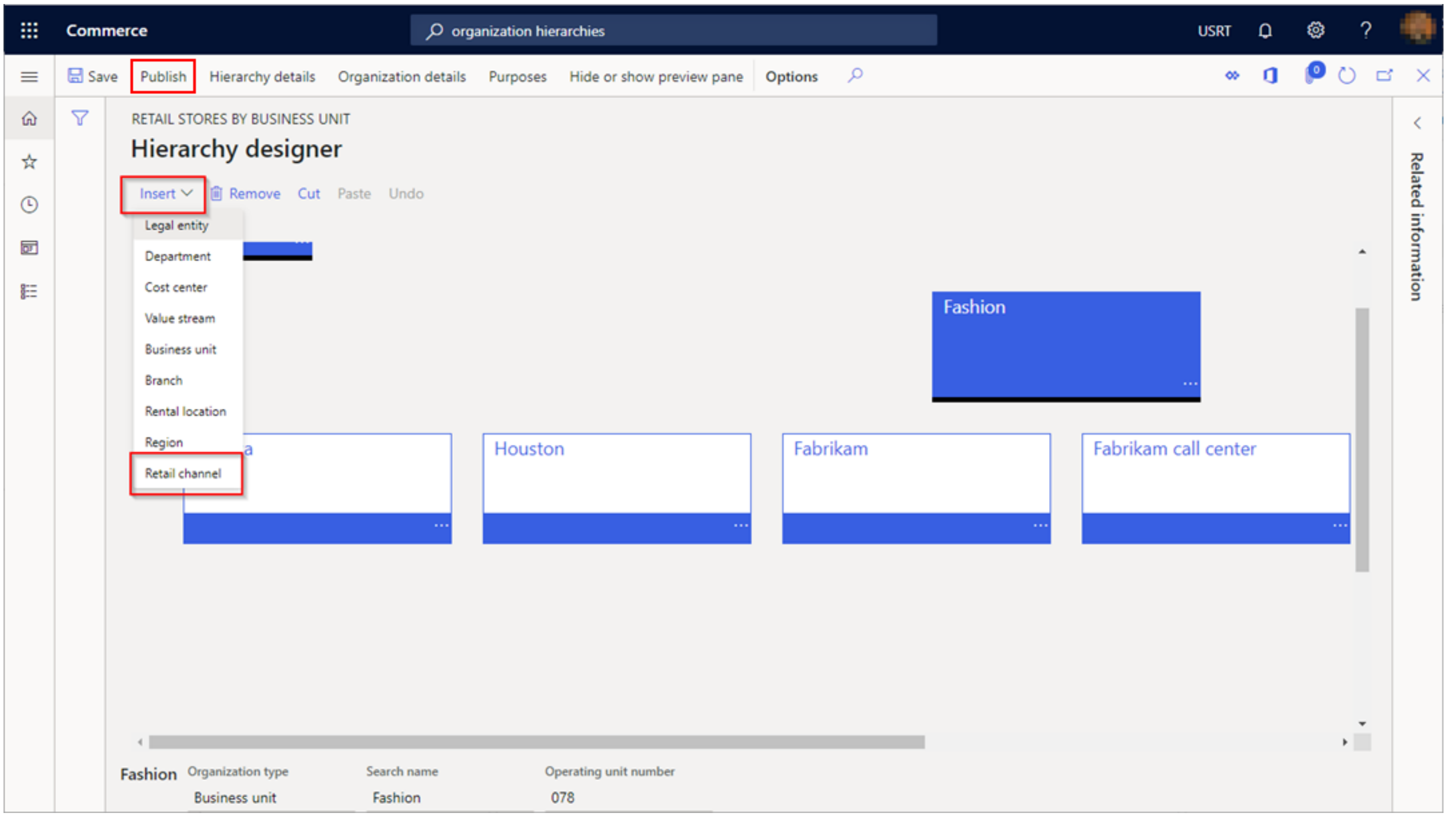


RETAIL STORES BY BUSINESS UNIT

Hierarchy designer

Contoso Retail	Organization type	Search name	Company
	Legal entity	Contoso Retail	GLRT

- Select "**Insert**" then "**Retail channel**" and pick the new channel to be added
- Note: You may need to select the arrow to see the new channel added as you scroll to the right.
- Save, **publish**, select an older date (example: 1st of last month since only 1 publish a day is allowed, if another publish is required you can pick a newer date)



RETAIL STORES BY BUSINESS UNIT

Hierarchy designer

Fashion

Organization type	Search name	Operating unit number
Business unit	Fashion	078

SETUP MODES OF DELIVERY

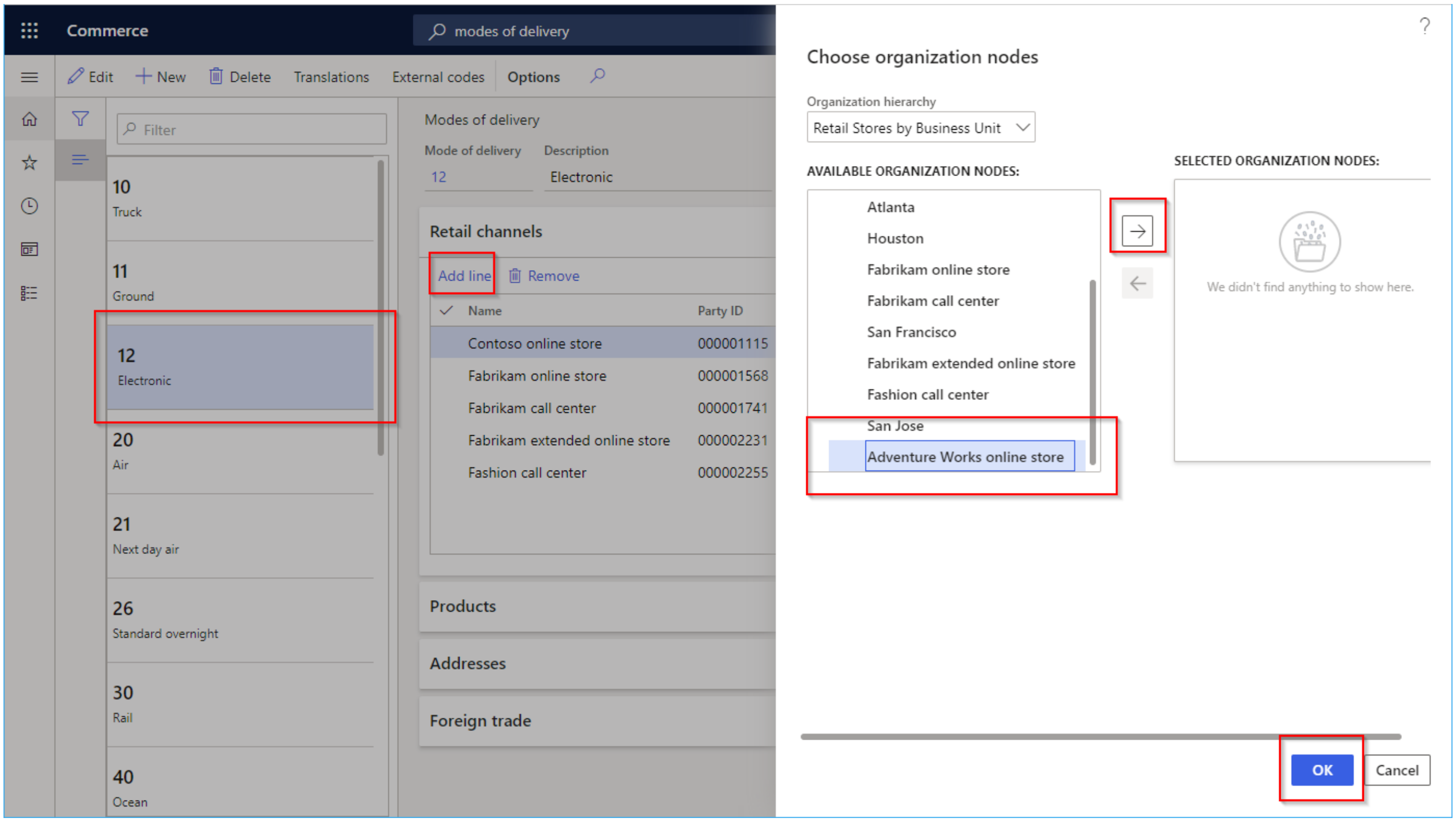
Modes of delivery are used at cart checkout when asking what delivery mode to use. We need to setup the modes of delivery available to the new online store channel.

Search for “**modes of delivery**” configuration in Headquarters.

Next, we will add this channel to the following modes of delivery.

- “12 - Electronic”
- “26 - Standard overnight”
- “60 - Customer pickup”
- “99 - Standard”
- “61 -Curbside pick up”

Select each one and under “**Retail channels**”, click the “**Add line**” link then add the new channel:



Modes of delivery

Mode of delivery	Description
12	Electronic

Name	Party ID
Contoso online store	000001115
Fabrikam online store	000001568
Fabrikam call center	000001741
Fabrikam extended online store	000002231
Fashion call center	000002255

Organization hierarchy: Retail Stores by Business Unit

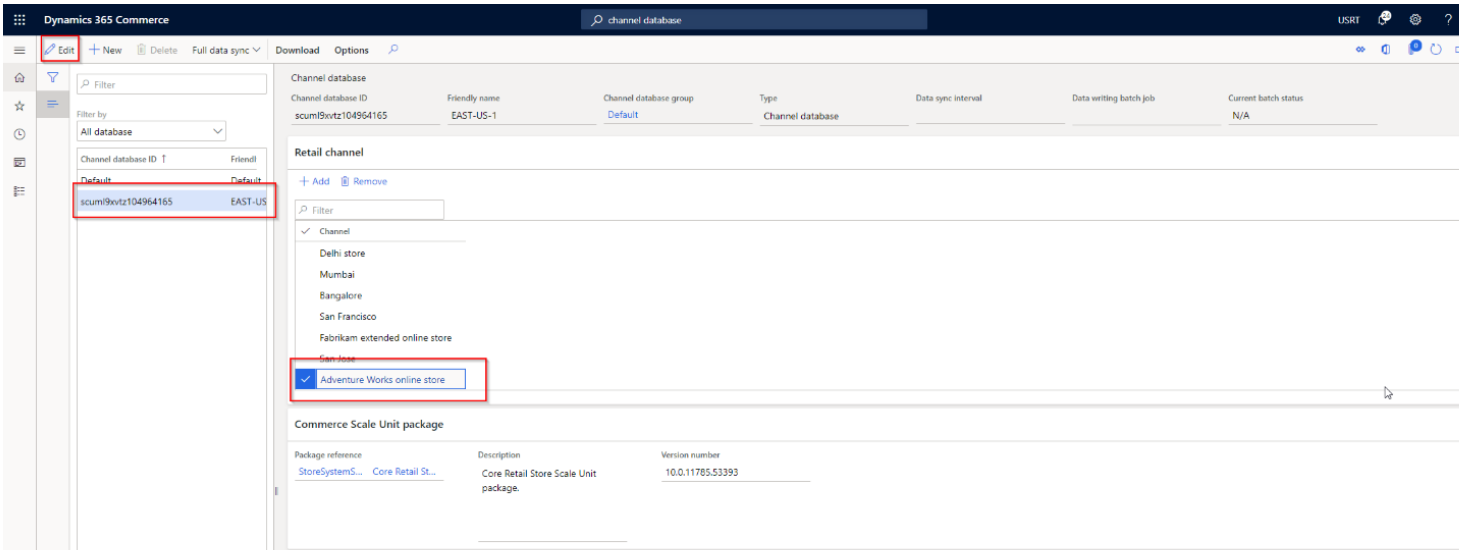
Atlanta
Houston
Fabrikam online store
Fabrikam call center
San Francisco
Fabrikam extended online store
Fashion call center
San Jose
Adventure Works online store

After any changes you must run “Process delivery modes”, we’ll run that in a later step.

CONFIGURE CHANNEL TO BE PART OF THE APPROPRIATE CHANNEL DATABASE

Now that the channel creation and configuration are complete, we need to ensure we pick the correct channel database. By default, the deployments will contain a scale unit that we should select.

- Search for “**Channel database**” and load the page
- Select the scale unit channel database (not the “Default”) as shown in the image below.
- Select “**Edit**” and add the new channel then click save



Channel database ID	Friendl
Default	Default
scuml9xvtz104964165	EAST-US

Channel database

Channel database ID	Friendly name	Channel database group	Type	Data sync interval	Data writing batch job	Current batch status
scuml9xvtz104964165	EAST-US-1	Default	Channel database			N/A

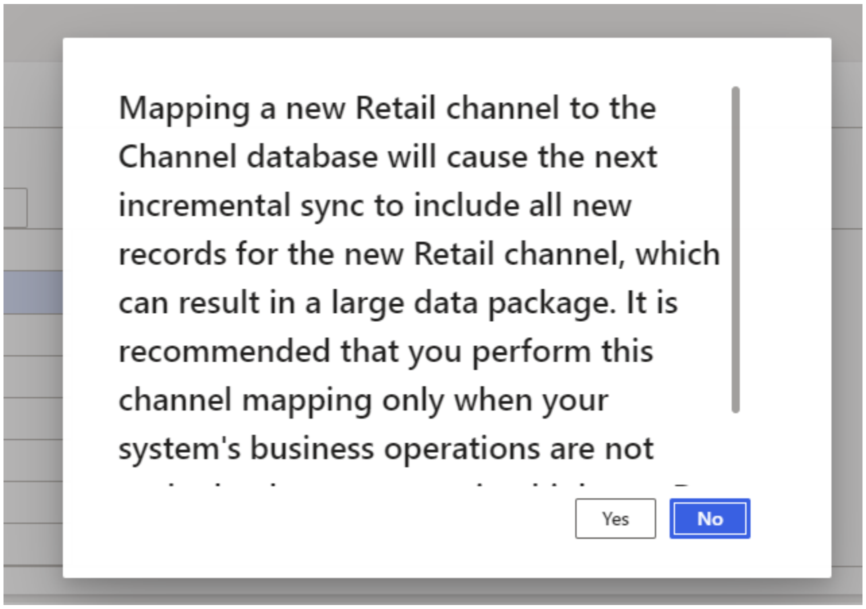
Retail channel

Channel
Delhi store
Mumbai
Bangalore
San Francisco
Fabrikam extended online store
San Jose
Adventure Works online store

Commerce Scale Unit package

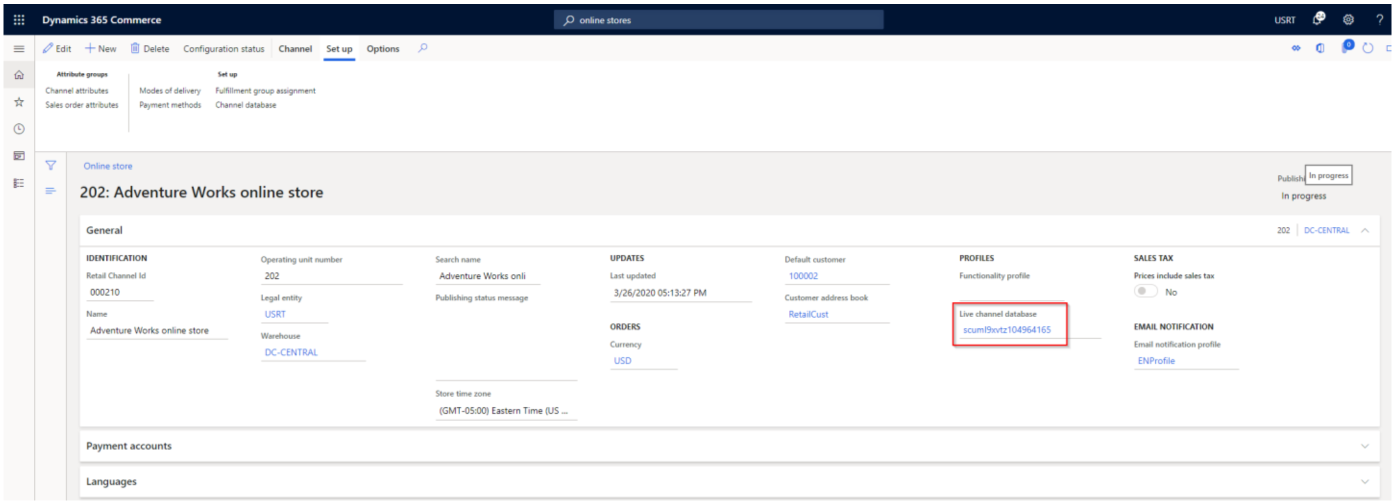
Package reference		Description	Version number
StoreSystemS...	Core Retail St...	Core Retail Store Scale Unit package.	10.0.11785.53393

Note: you should get a warning message as below, stating this could take some time, select “Yes”


Mapping a new Retail channel to the Channel database will cause the next incremental sync to include all new records for the new Retail channel, which can result in a large data package. It is recommended that you perform this channel mapping only when your system's business operations are not

- Note we will later run the “Full data sync” from here but not yet…

If you now go back to the online store channel you should see the channel database listed as shown below:



Online store

202: Adventure Works online store

Publishi In progress
In progress

General

202 | DC-CENTRAL

IDENTIFICATION

Retail Channel Id
000210

Name
Adventure Works online store

Operating unit number
202

Legal entity
USRT

Warehouse
DC-CENTRAL

Search name
Adventure Works onli

Publishing status message

Store time zone
(GMT-05:00) Eastern Time (US ...

UPDATES

Last updated
3/26/2020 05:13:27 PM

ORDERS

Currency
USD

Default customer
100002

Customer address book
RetailCust

PROFILES

Functionality profile

Live channel database
scuml9xvtz104964165

SALES TAX

Prices include sales tax
No

EMAIL NOTIFICATION

Email notification profile
ENProfile

Payment accounts

Languages

We now have a completed channel setup!

CREATING A NEW PRODUCT

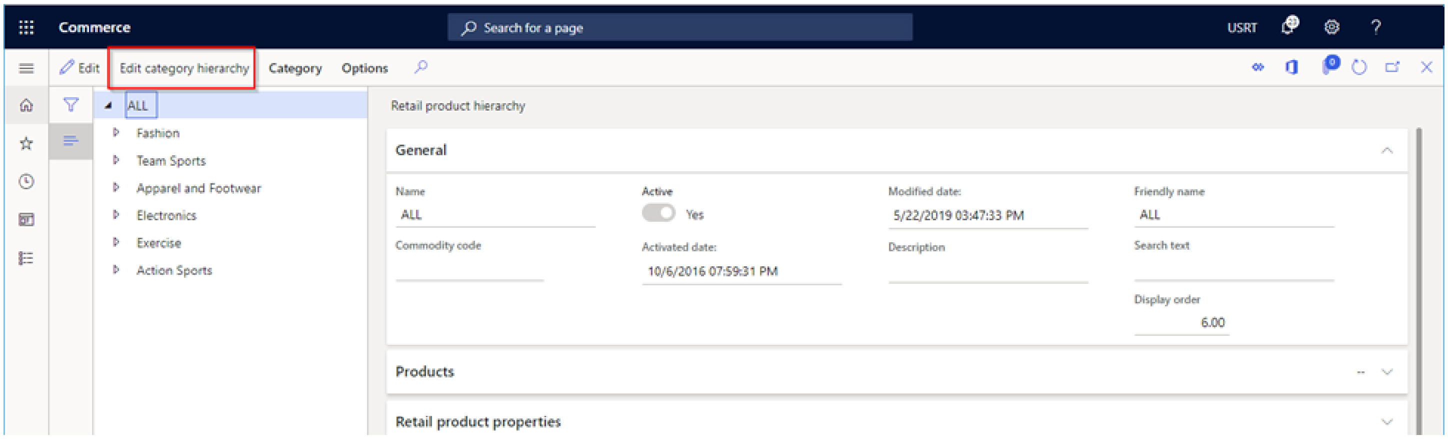
In this section we will create a single new "backpack" product, more products can be created following the same steps. To start we are going to create a new global category for the product.

CREATE A NEW PRODUCT HIERARCHY NODE

Since the demo data already has a product hierarchy defined, we are going to add a new node to it for our backpack example product we'll be adding. You could pick an existing category for your product or follow along and create a new one.

From within Headquarters, search for "**Commerce product hierarchy**".

Select the "**Edit category hierarchy**" link



- ALL
  - Fashion
  - Team Sports
  - Apparel and Footwear
  - Electronics
  - Exercise
  - Action Sports

Retail product hierarchy

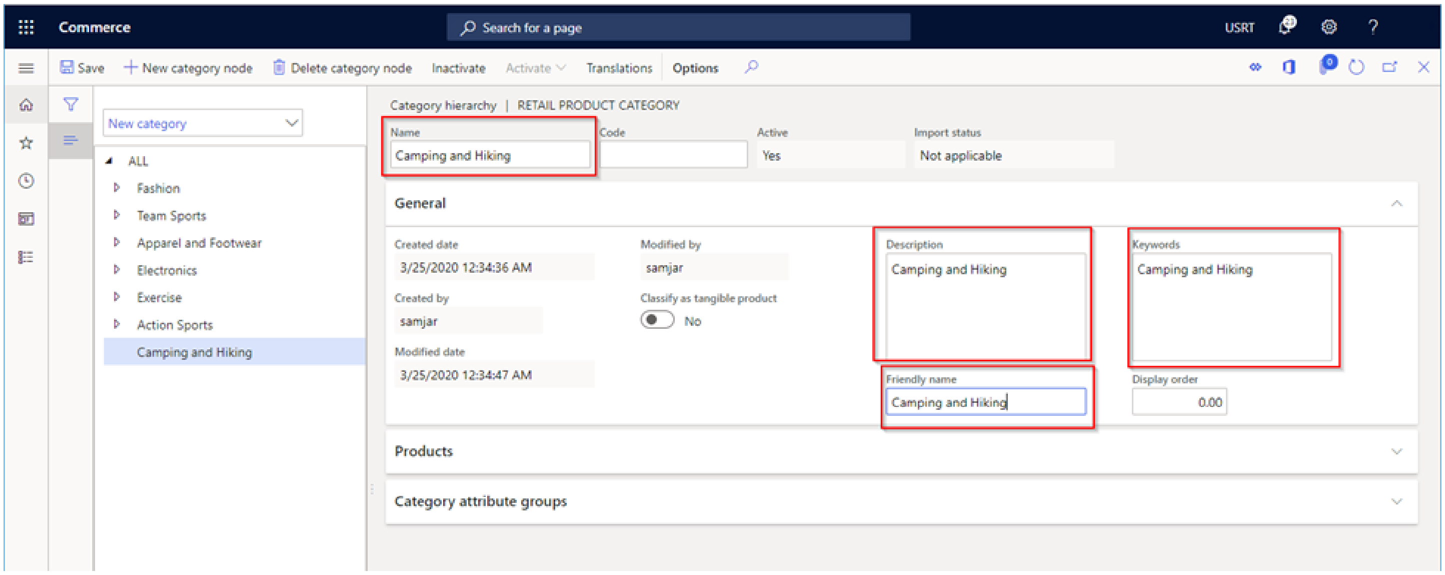
General

Name: ALL
Active: Yes
Modified date: 5/22/2019 03:47:33 PM
Friendly name: ALL
Commodity code:
Activated date: 10/6/2016 07:59:31 PM
Description:
Search text:
Display order: 6.00

Products

Retail product properties

- Select the node you want to add a new node to
  - Example we will add to the top level "**ALL**" node, select it and press the "**New category node"** button at the top
- Fill in the name, friendly name, description, and keywords



- ALL
  - Fashion
  - Team Sports
  - Apparel and Footwear
  - Electronics
  - Exercise
  - Action Sports
  - Camping and Hiking

Category hierarchy | RETAIL PRODUCT CATEGORY

Name: Camping and Hiking
Code:
Active: Yes
Import status: Not applicable

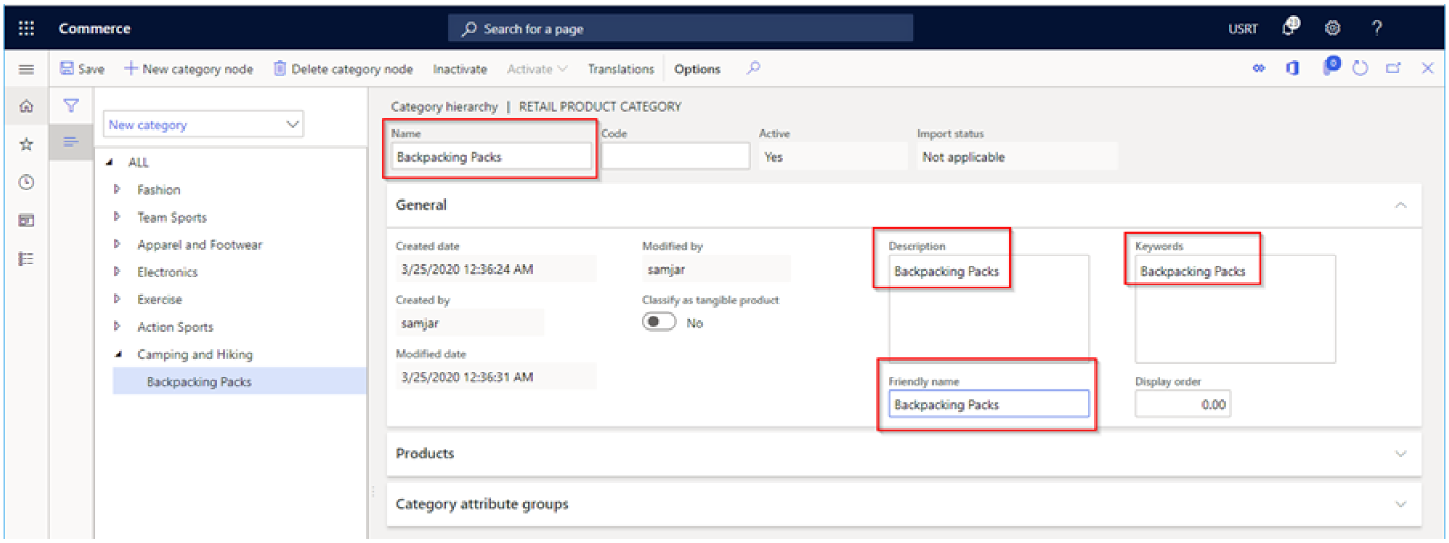
General

Created date: 3/25/2020 12:34:36 AM
Modified by: samjar
Description: Camping and Hiking
Keywords: Camping and Hiking
Created by: samjar
Classify as tangible product: No
Modified date: 3/25/2020 12:34:47 AM
Friendly name: Camping and Hiking
Display order: 0.00

Products

Category attribute groups

In our example we will now create a new node for the new product under “Camping and Hiking” called “Backpacking Packs”.



Category hierarchy | RETAIL PRODUCT CATEGORY

Name	Code	Active	Import status
Backpacking Packs		Yes	Not applicable

General

Created date: 3/25/2020 12:36:24 AM

Created by: samjar

Modified date: 3/25/2020 12:36:31 AM

Modified by: samjar

Classify as tangible product: No

Description: Backpacking Packs

Friendly name: Backpacking Packs

Keywords: Backpacking Packs

Display order: 0.00

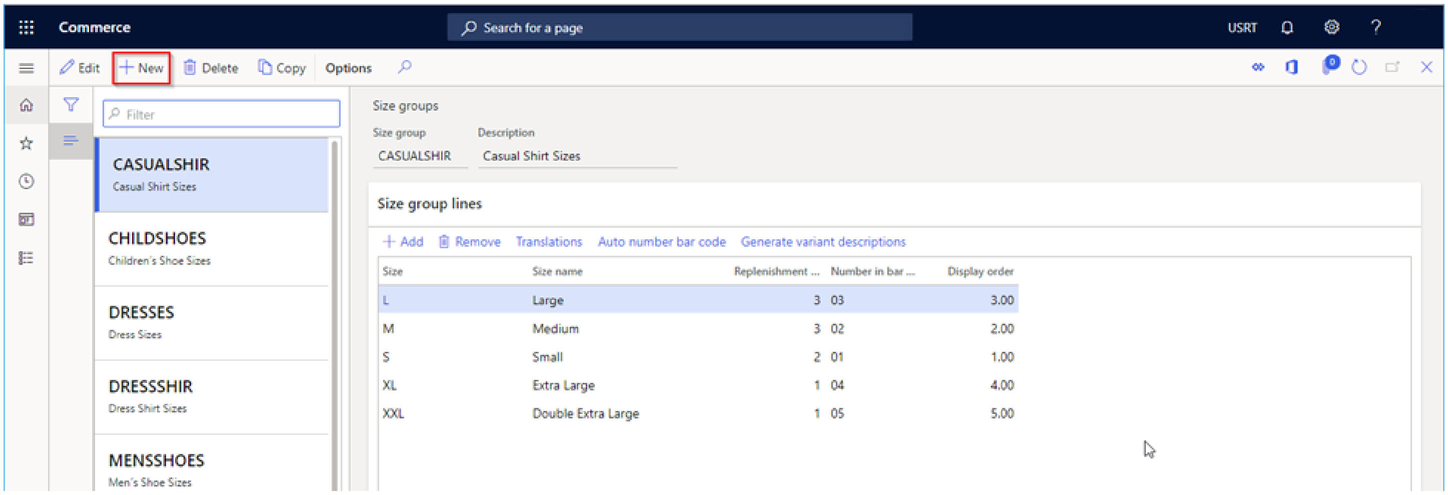
Products

Category attribute groups

CREATE NEW SIZE GROUP

Products can contain variants such as size, color and style. The example product we are going to add will be a backpack that is available in 3 sizes, we need to create the size group first. The size group example will have 3 sizes called Small, Medium and Large. The different variants of a product are important later when adding product images since variant images will need to be used along with the variant name in the file name. This step is needed before we add the product since we need to create the product with the groups. The following steps can generally be followed to add color or style groups as well if needed.

- Search for the "**size groups**" page
- Select the "**+ New**" button at the top

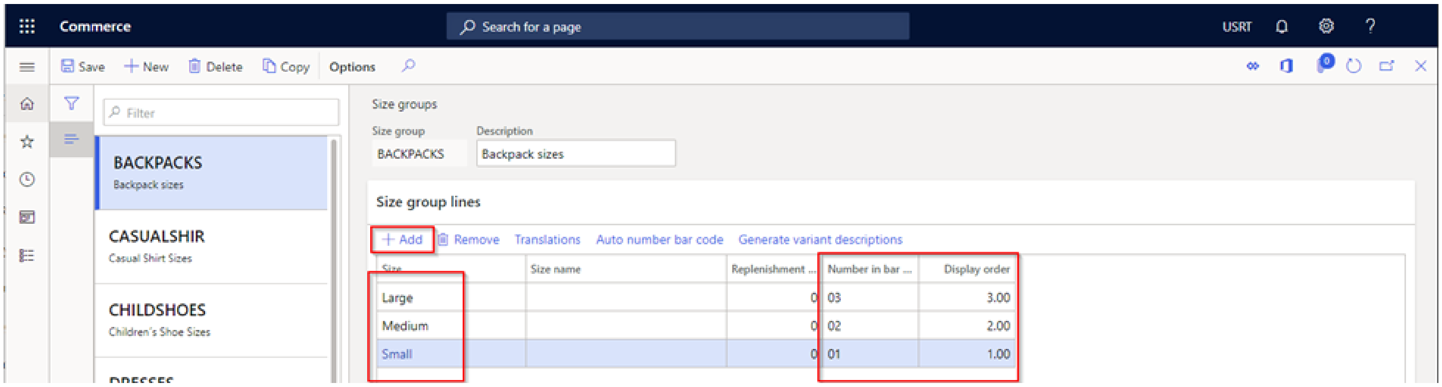


Size group: CASUALSHIR
Description: Casual Shirt Sizes

Size	Size name	Replenishment ...	Number in bar ...	Display order
L	Large	3	03	3.00
M	Medium	3	02	2.00
S	Small	2	01	1.00
XL	Extra Large	1	04	4.00
XXL	Double Extra Large	1	05	5.00

The below example adds a "**BACKPACKS**" size group with Description "**Backpack sizes**"

- Add **Small**, **Medium** and **Large** sizes with display order **1,2,3** respectively and **01,02,03** as Number in barcode and replenishment of 1 (Name and description are not required)



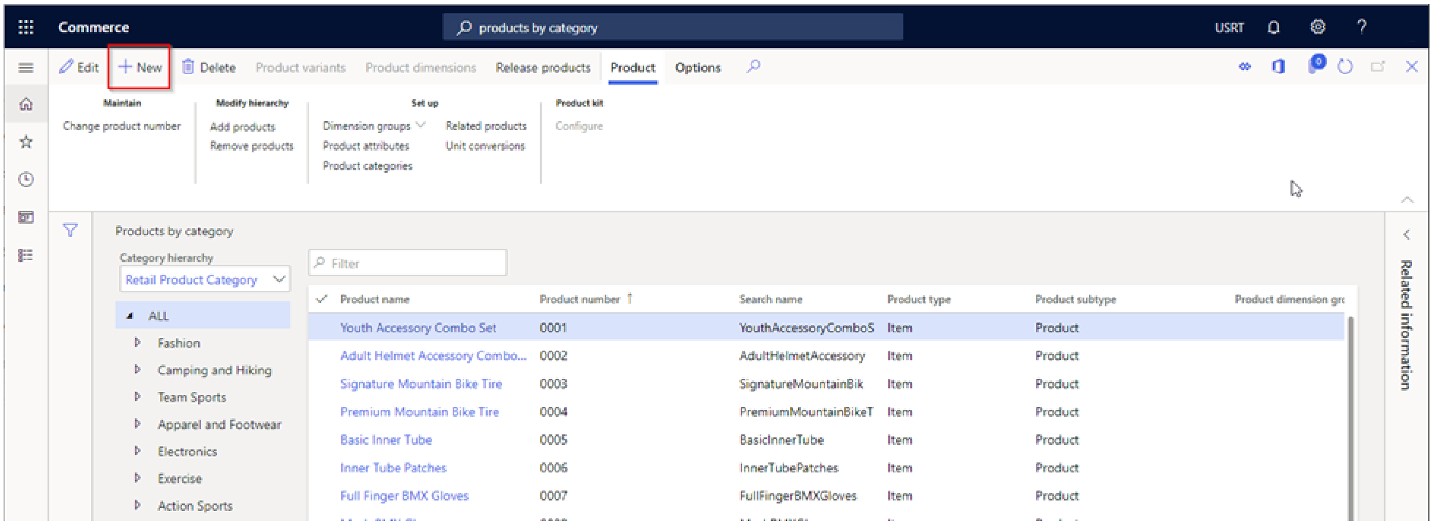
Size group: BACKPACKS
Description: Backpack sizes

Size	Size name	Replenishment ...	Number in bar ...	Display order
Large		0	03	3.00
Medium		0	02	2.00
Small		0	01	1.00

ADDING A NEW PRODUCT

We are now ready to manually add a product. The below example shows how to add a new backpack with size variants using the size group we just created.

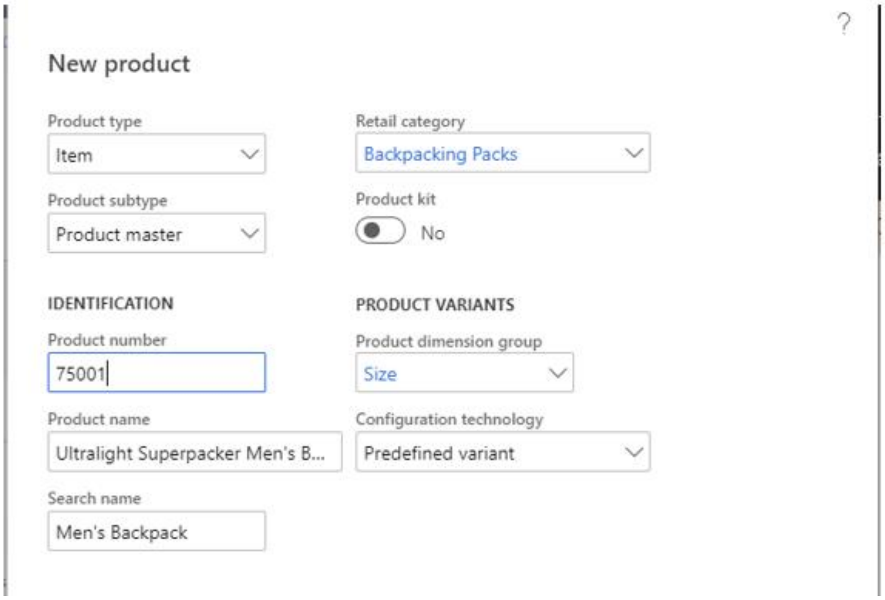
- Search for the "**Products by category**" page
- **Note**: make sure you are not in “Released products by category”
- Select "**+ New**" button from the top



Product name	Product number	Search name	Product type	Product subtype	Product dimension grc
Youth Accessory Combo Set	0001	YouthAccessoryComboS	Item	Product	
Adult Helmet Accessory Combo...	0002	AdultHelmetAccessory	Item	Product	
Signature Mountain Bike Tire	0003	SignatureMountainBik	Item	Product	
Premium Mountain Bike Tire	0004	PremiumMountainBikeT	Item	Product	
Basic Inner Tube	0005	BasicInnerTube	Item	Product	
Inner Tube Patches	0006	InnerTubePatches	Item	Product	
Full Finger BMX Gloves	0007	FullFingerBMXGloves	Item	Product	

Fill out product details, below is an example:

- **Product type** = "**Item**"
- **Product subtype** = "**Product Master**"
- **Product number** = “**75001**” (this can be any number that is unique)
- **Product Name** = "**Ultralight Superpacker Men's Backpack**"
- **Product dimension group** = "**Size**"
- **Retail category** = "**Backpacking Packs**" (note: pick the category you created earlier with your name in it)
- **Search name** = "**Men's Backpack**"
- Click "**OK**"



New product

Product type: Item

Product subtype: Product master

IDENTIFICATION

Product number: 75001

Product name: Ultralight Superpacker Men's B...

Search name: Men's Backpack

Retail category: Backpacking Packs

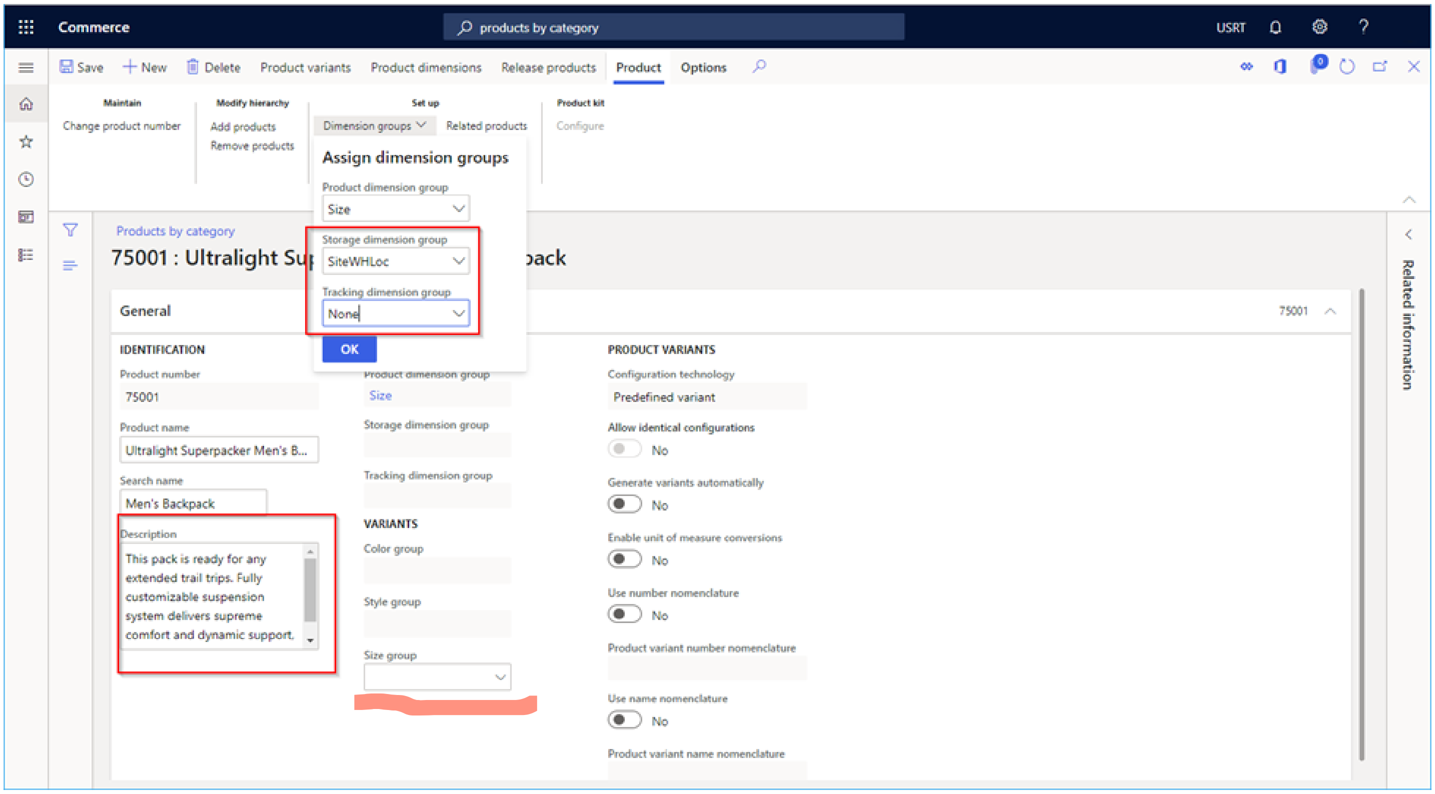
Product kit: No

PRODUCT VARIANTS

Product dimension group: Size

Configuration technology: Predefined variant

- Next add a **Product Description** (below example uses: “This pack is ready for any extended trail trip. Fully customizable suspension system delivers supreme comfort and dynamic support.”)
- Select "**Dimension groups**"
    - Select "**Storage dimension group**" = **SiteWHLoc**
        - This will ensure the product inventory can be added to a warehouse with specific defined locations (you will see this later when we add inventory)
    - Select "**Tracking dimension group**" = "**None**"



Assign dimension groups

Product dimension group: Size

Storage dimension group: SiteWHLoc

Tracking dimension group: None

Products by category

75001 : Ultralight Su[...]pack

General 75001

IDENTIFICATION

Product number: 75001

Product name: Ultralight Superpacker Men's B...

Search name: Men's Backpack

Description: This pack is ready for any extended trail trips. Fully customizable suspension system delivers supreme comfort and dynamic support.

Product dimension group: Size

Storage dimension group:

Tracking dimension group:

VARIANTS

Color group:

Style group:

Size group:

PRODUCT VARIANTS

Configuration technology: Predefined variant

Allow identical configurations: No

Generate variants automatically: No

Enable unit of measure conversions: No

Use number nomenclature: No

Product variant number nomenclature:

Use name nomenclature: No

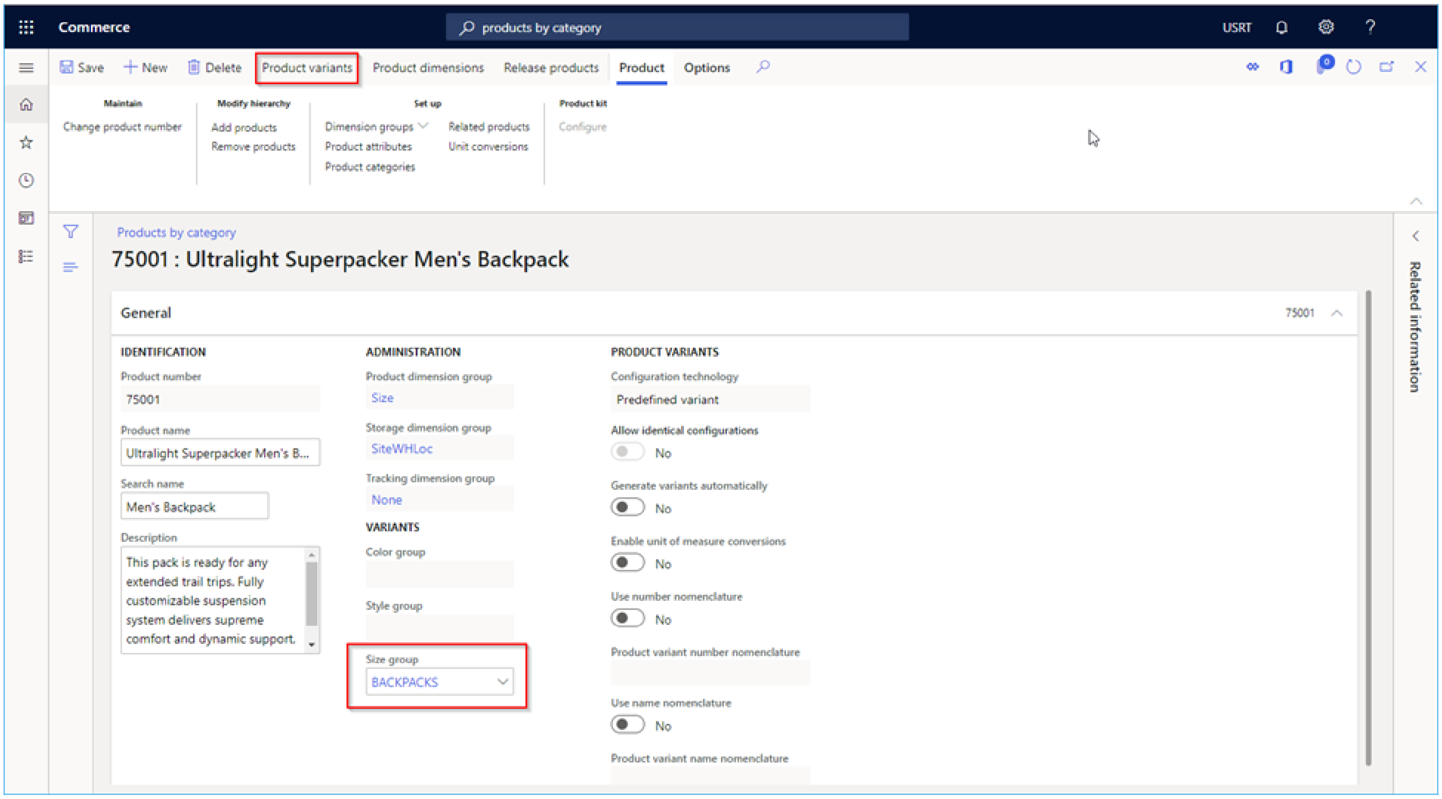
Product variant name nomenclature:

Now we can select the “**Size group**” we created earlier called “**BACKPACKS**”

Define Product variants

Now we need to define the specific variants our product will support, for example a color group could be chosen with a generic list of colors, now you can pick the specific ones available on the product.

- Select "**Product variants**"



Products by category

75001 : Ultralight Superpacker Men's Backpack

General 75001

IDENTIFICATION

Product number: 75001

Product name: Ultralight Superpacker Men's B...

Search name: Men's Backpack

Description: This pack is ready for any extended trail trips. Fully customizable suspension system delivers supreme comfort and dynamic support.

ADMINISTRATION

Product dimension group: Size

Storage dimension group: SiteWHLoc

Tracking dimension group: None

VARIANTS

Color group

Style group

Size group: BACKPACKS

PRODUCT VARIANTS

Configuration technology: Predefined variant

Allow identical configurations: No

Generate variants automatically: No

Enable unit of measure conversions: No

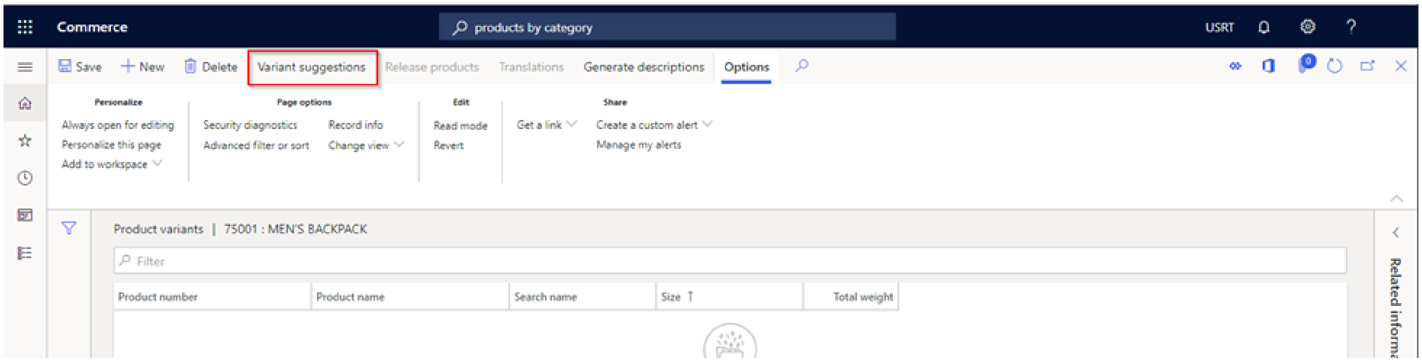
Use number nomenclature: No

Product variant number nomenclature

Use name nomenclature: No

Product variant name nomenclature

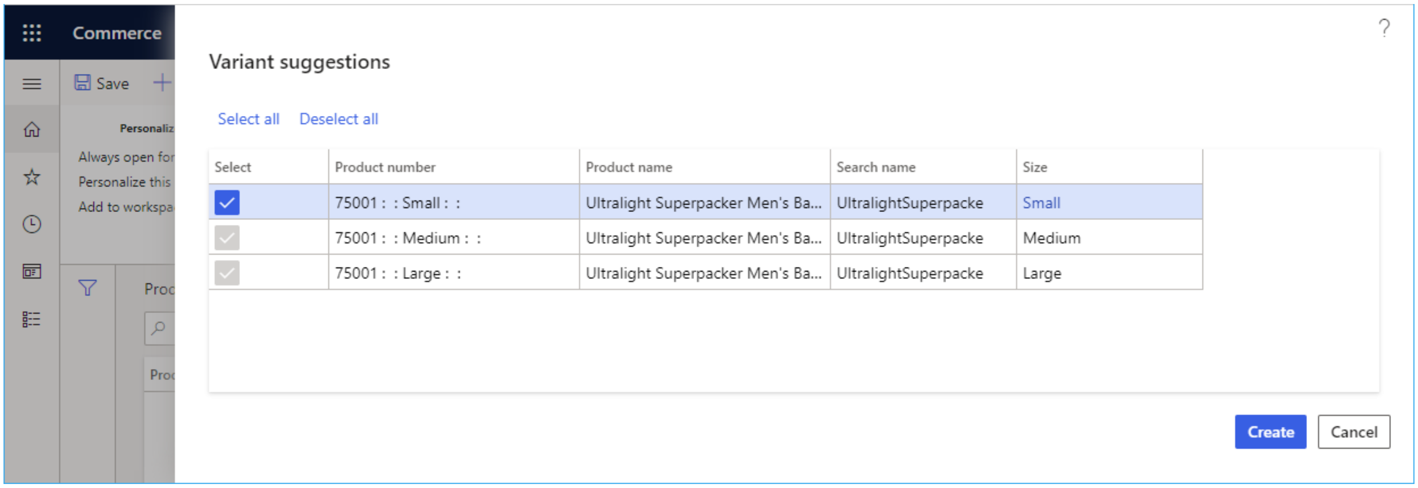
- Select "**Variant suggestions**"



Product variants | 75001 : MEN'S BACKPACK

Product number	Product name	Search name	Size ↑	Total weight

- Select "**Select all**"
- Select "**Create**” button
- Click “**X**” in upper right to close to take you back to product page, no need to click save it should auto save



Variant suggestions

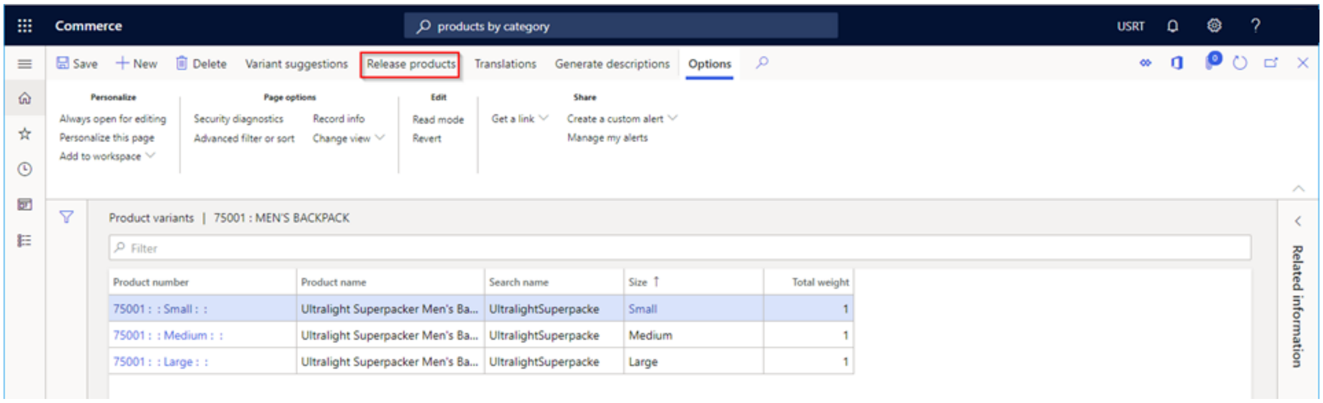
Select	Product number	Product name	Search name	Size
☑	75001 : : Small : :	Ultralight Superpacker Men's Ba...	UltralightSuperpacke	Small
✓	75001 : : Medium : :	Ultralight Superpacker Men's Ba...	UltralightSuperpacke	Medium
✓	75001 : : Large : :	Ultralight Superpacker Men's Ba...	UltralightSuperpacke	Large

We have now finished creating a product and its variants.

RELEASE PRODUCTS TO A LEGAL ENTITY

Now that the product is added to our global list of products, we must release it to a legal entity to sell it. In our example below we will release to USRT where our online channel is defined.

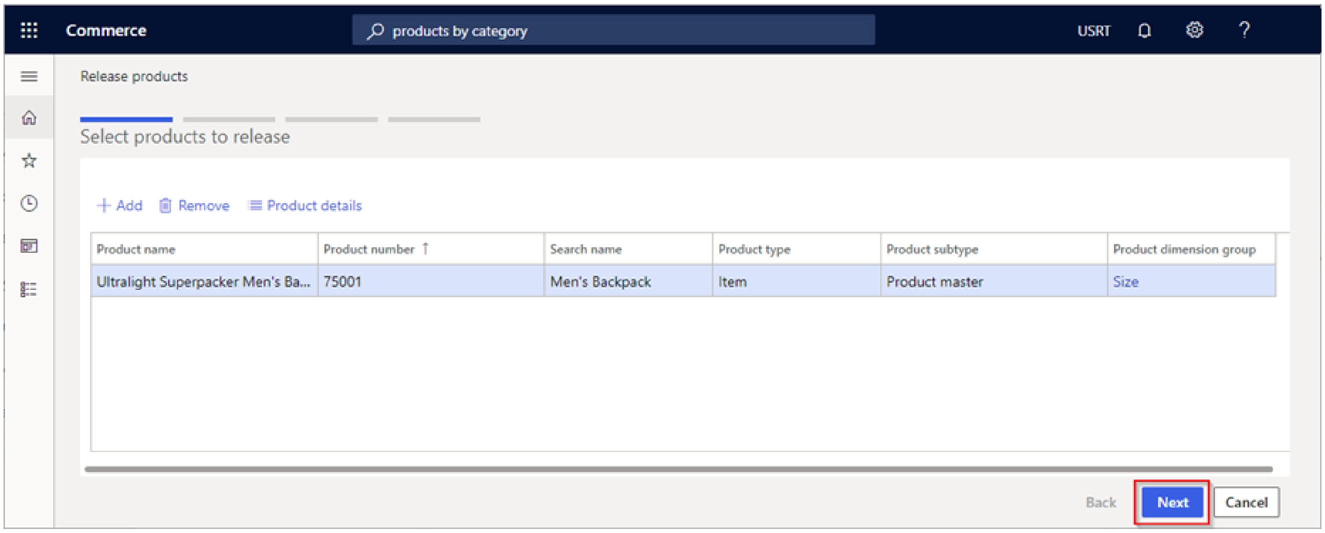
Select "**Release products**"



Product variants | 75001 : MEN'S BACKPACK

Product number	Product name	Search name	Size ↑	Total weight
75001 : : Small : :	Ultralight Superpacker Men's Ba...	UltralightSuperpacke	Small	1
75001 : : Medium : :	Ultralight Superpacker Men's Ba...	UltralightSuperpacke	Medium	1
75001 : : Large : :	Ultralight Superpacker Men's Ba...	UltralightSuperpacke	Large	1

Select "**Next**"

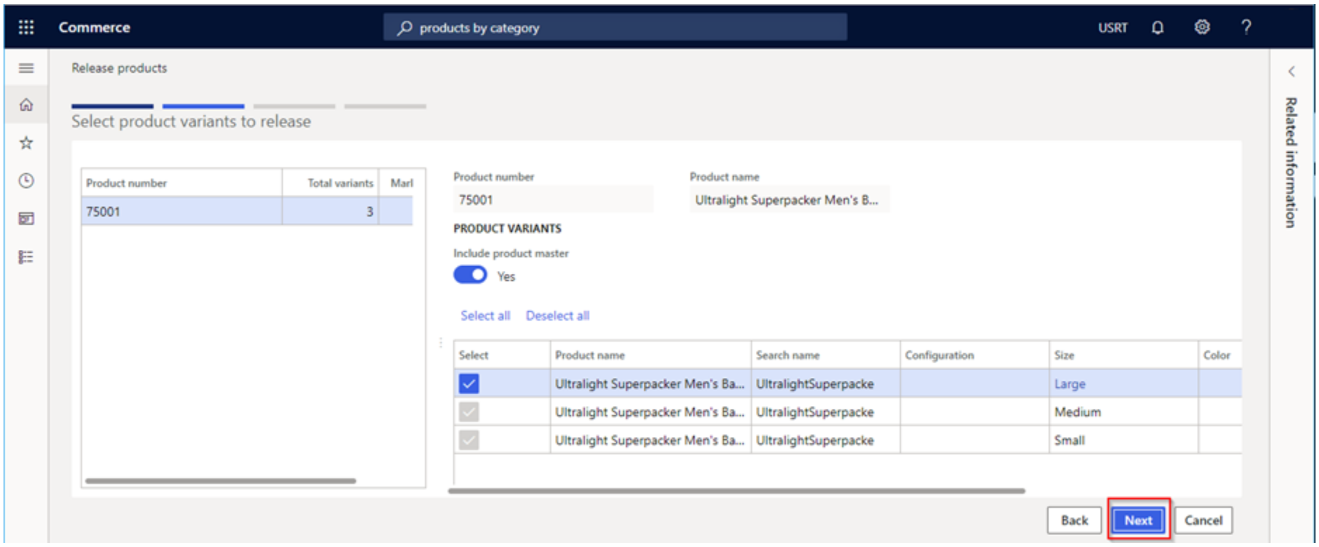


Release products

Select products to release

Product name	Product number ↑	Search name	Product type	Product subtype	Product dimension group
Ultralight Superpacker Men's Ba...	75001	Men's Backpack	Item	Product master	Size

Select "**Next**"



Release products

Select product variants to release

Product number	Total variants	Marl
75001	3	

Product number: 75001

Product name: Ultralight Superpacker Men's B...

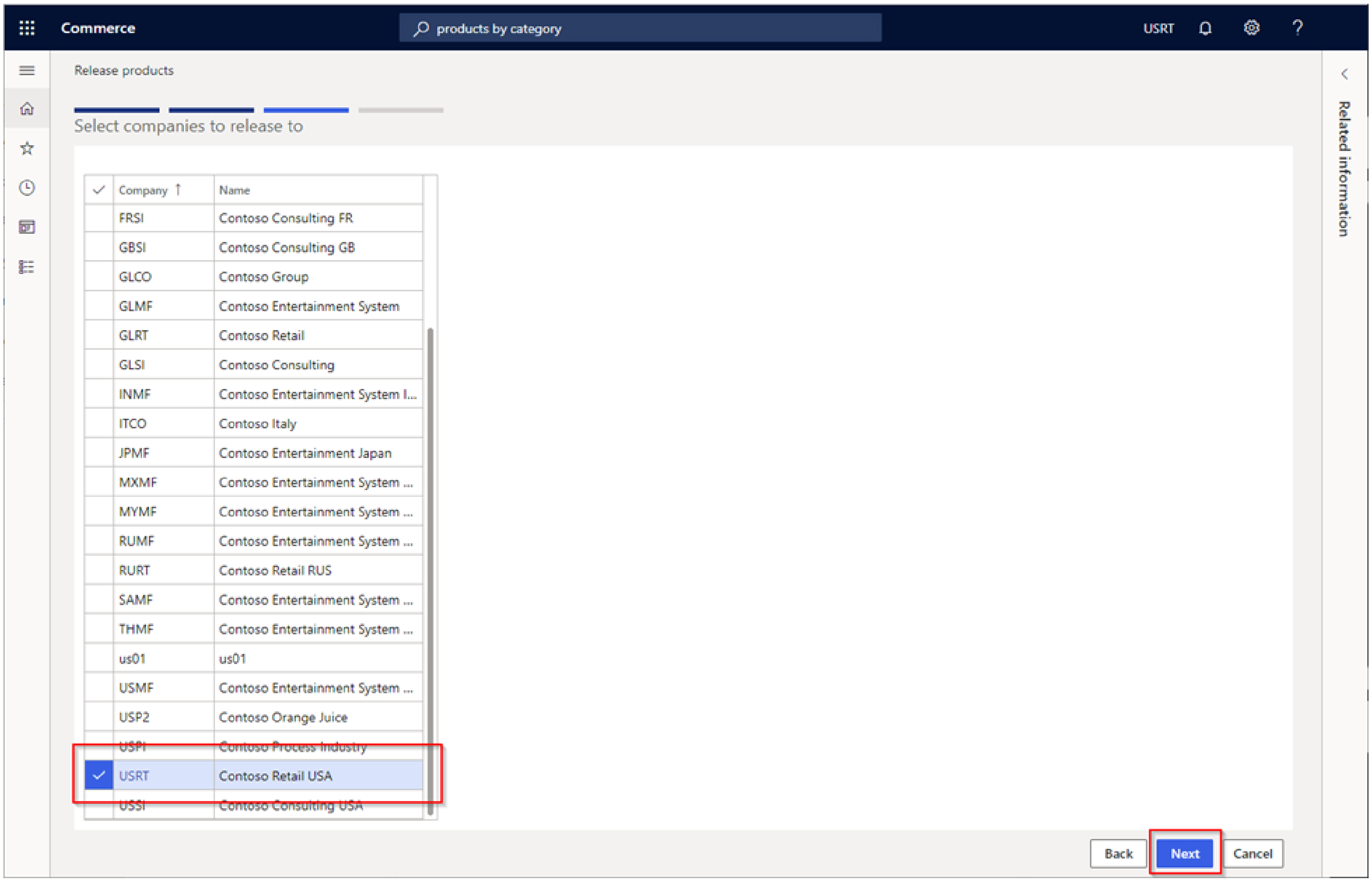
PRODUCT VARIANTS

Include product master: Yes

Select all Deselect all

Select	Product name	Search name	Configuration	Size	Color
☑	Ultralight Superpacker Men's Ba...	UltralightSuperpacke		Large	
☐	Ultralight Superpacker Men's Ba...	UltralightSuperpacke		Medium	
☐	Ultralight Superpacker Men's Ba...	UltralightSuperpacke		Small	

Select "**USRT**" and "**Next**"

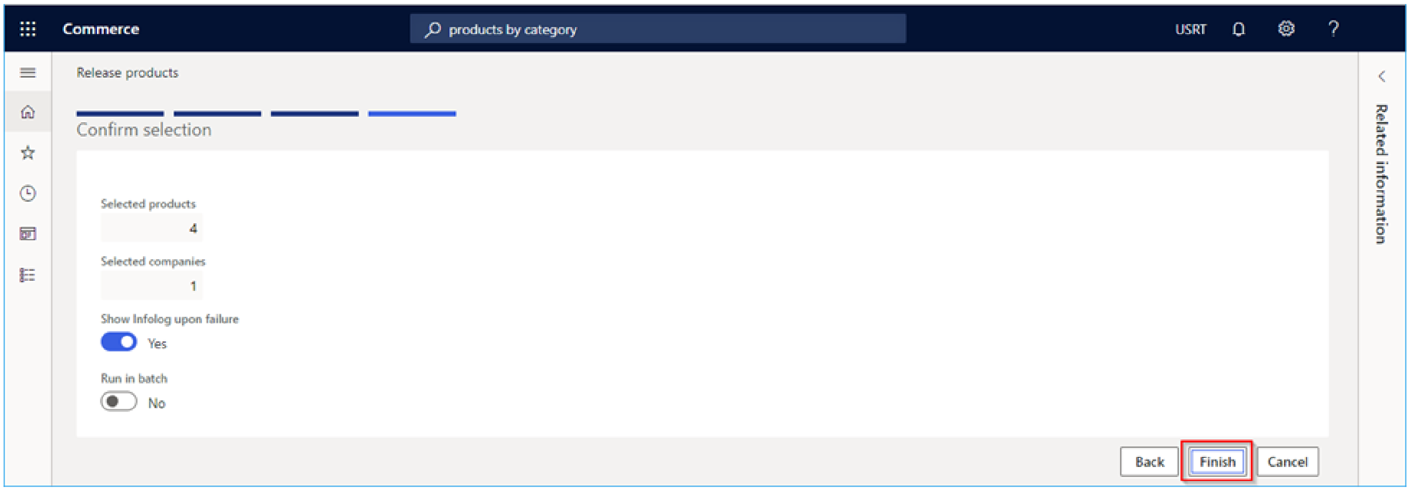


Release products

Select companies to release to

✓	Company ↑	Name
	FRSI	Contoso Consulting FR
	GBSI	Contoso Consulting GB
	GLCO	Contoso Group
	GLMF	Contoso Entertainment System
	GLRT	Contoso Retail
	GLSI	Contoso Consulting
	INMF	Contoso Entertainment System I...
	ITCO	Contoso Italy
	JPMF	Contoso Entertainment Japan
	MXMF	Contoso Entertainment System ...
	MYMF	Contoso Entertainment System ...
	RUMF	Contoso Entertainment System ...
	RURT	Contoso Retail RUS
	SAMF	Contoso Entertainment System ...
	THMF	Contoso Entertainment System ...
	us01	us01
	USMF	Contoso Entertainment System ...
	USP2	Contoso Orange Juice
	USPI	Contoso Process Industry
✓	USRT	Contoso Retail USA
	USSI	Contoso Consulting USA

Select "**Finish**"



Release products

Confirm selection

Selected products
4

Selected companies
1

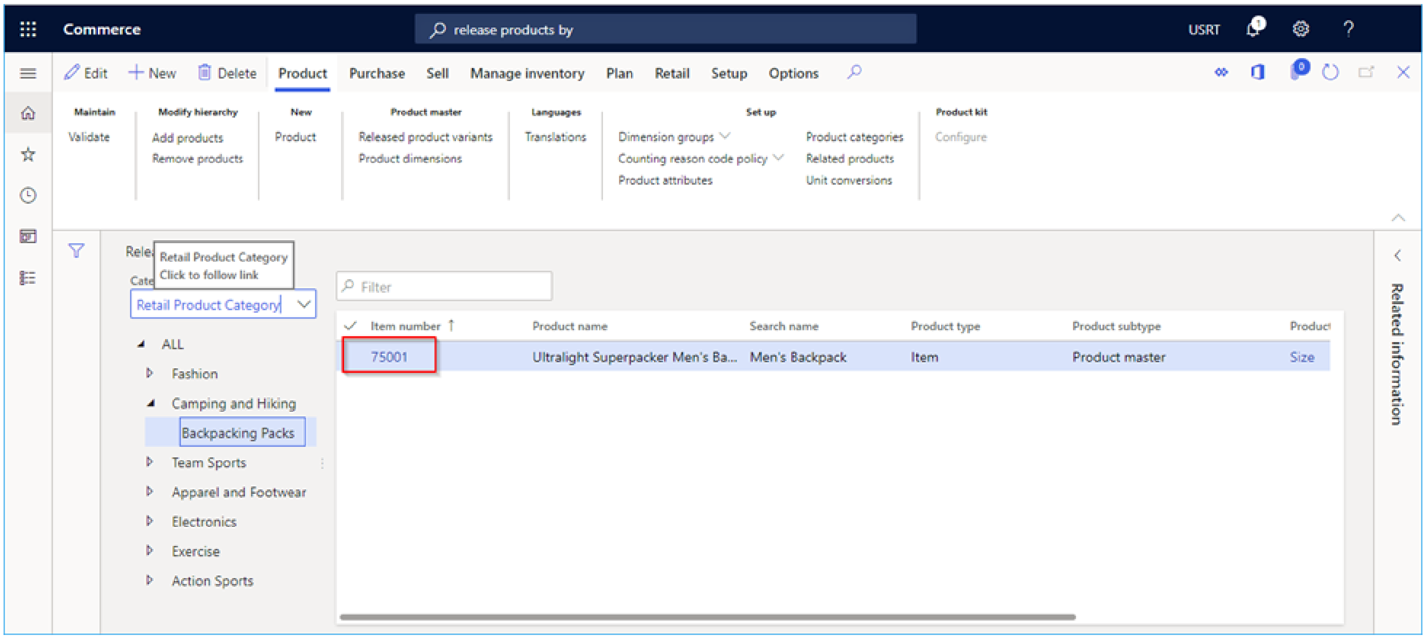
Show Infolog upon failure
● Yes

Run in batch
○ No

CONFIGURE RELEASED PRODUCT PROPERTIES

Once products are released, we can now configure more of the product information such as selling price. Below will show how to configure the example product created earlier.

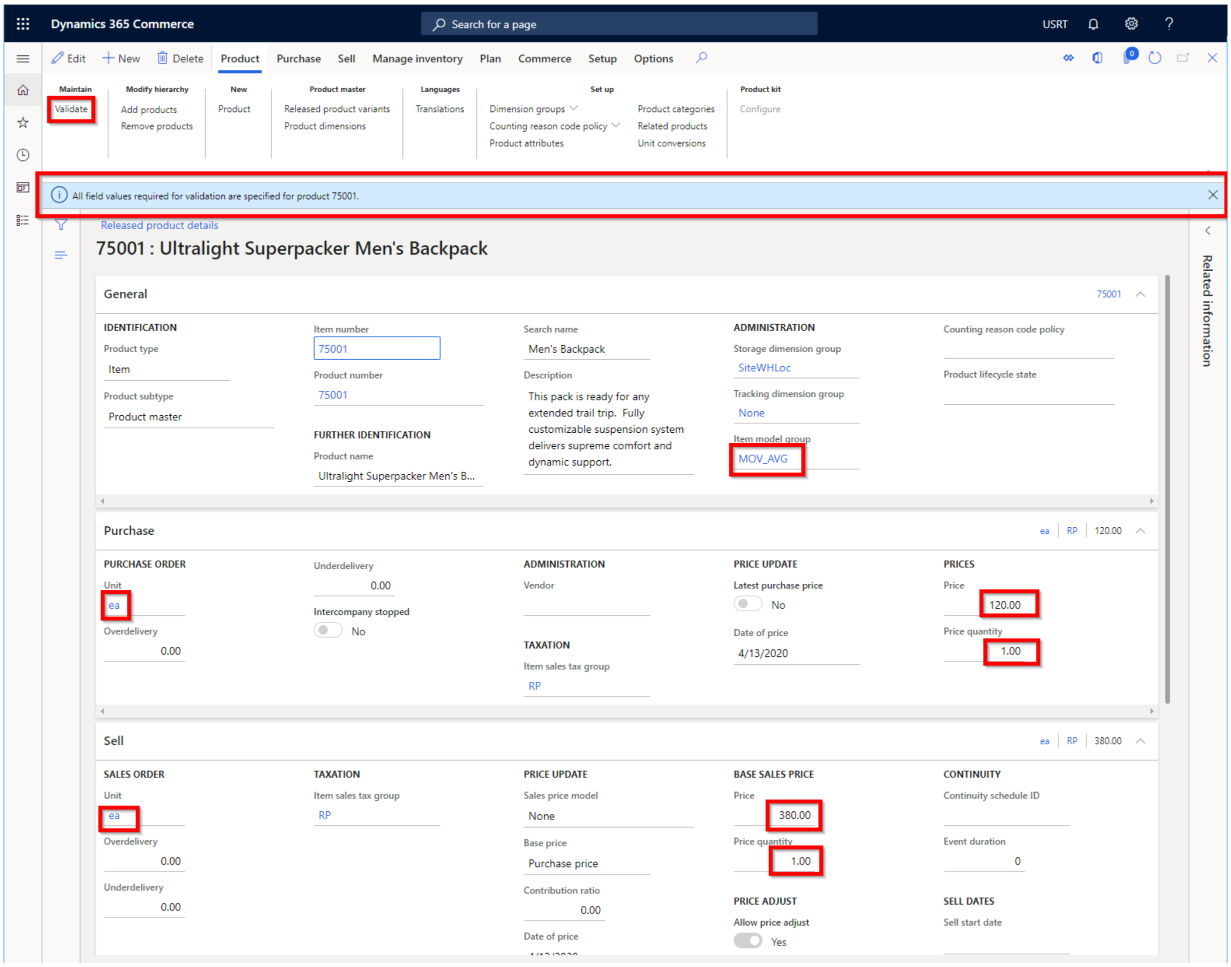
- Search for the "**Released Products by Category**" page.
- Select the new product number link under "**Camping and Hiking**" -> "**Backpacking Packs**" category



Item number	Product name	Search name	Product type	Product subtype	Product
75001	Ultralight Superpacker Men's Ba...	Men's Backpack	Item	Product master	Size

We will now modify some of the data, there is a minimal set of data that needs to be set before the product will successfully validate. Ensure the product validates successfully before proceeding or it will not show up on the site.

- Select "**Edit**" and make below changes
    - "**General**" section set
        - **Item model group** : **Mov_AVG**
    - "**Purchase**" section add
        - **Unit**: "ea",
        - **Price=**120.00**, Quantity=**1.00
    - "**Sell**" section add
        - **Unit**: **ea,**
        - **Price=**380.00, **Quantity=1.00**
    - "**Manage Inventory**" section add
        - "**Unit**": **ea**
    - "**Manage Costs**" section pick an
        - "**Item Group**" = "**Action Sports**"
    - Select "**Validate**" then confirm there are no errors and click "**Save**"
        - Once validated you should see a message at the top saying "All field values required for validation are specified for product 75001.". As shown below.



All field values required for validation are specified for product 75001.

Released product details

75001 : Ultralight Superpacker Men's Backpack

General — 75001

IDENTIFICATION
Product type: Item
Product subtype: Product master
Item number: 75001
Product number: 75001
FURTHER IDENTIFICATION
Product name: Ultralight Superpacker Men's B...
Search name: Men's Backpack
Description: This pack is ready for any extended trail trip. Fully customizable suspension system delivers supreme comfort and dynamic support.
ADMINISTRATION
Storage dimension group: SiteWHLoc
Tracking dimension group: None
Item model group: MOV_AVG
Counting reason code policy:
Product lifecycle state:

Purchase — ea | RP | 120.00

PURCHASE ORDER
Unit: ea
Overdelivery: 0.00
Underdelivery: 0.00
Intercompany stopped: No
ADMINISTRATION
Vendor:
TAXATION
Item sales tax group: RP
PRICE UPDATE
Latest purchase price: No
Date of price: 4/13/2020
PRICES
Price: 120.00
Price quantity: 1.00

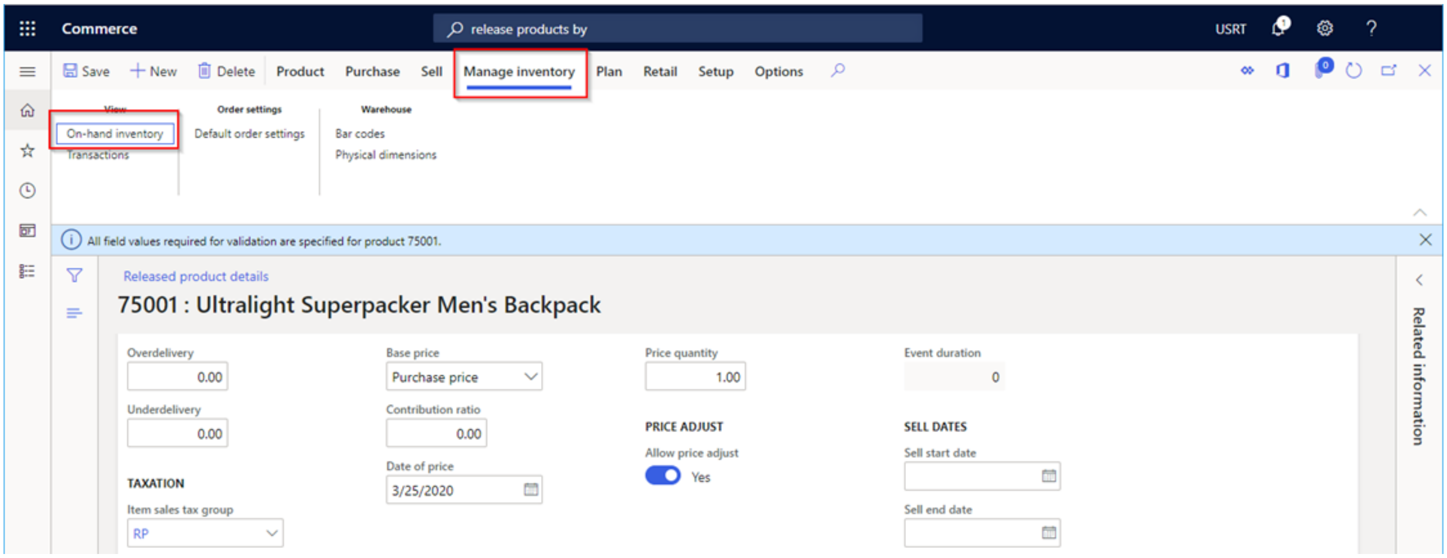
Sell — ea | RP | 380.00

SALES ORDER
Unit: ea
Overdelivery: 0.00
Underdelivery: 0.00
TAXATION
Item sales tax group: RP
PRICE UPDATE
Sales price model: None
Base price: Purchase price
Contribution ratio: 0.00
Date of price:
BASE SALES PRICE
Price: 380.00
Price quantity: 1.00
PRICE ADJUST
Allow price adjust: Yes
CONTINUITY
Continuity schedule ID:
Event duration: 0
SELL DATES
Sell start date:

ADD PRODUCT INVENTORY

We will now add product inventory for each variant and for both the warehouse for the online store (DC-Central) as well as San Francisco for BOPIS scenarios. San Francisco retail is the most updated POS so it best used for demos.

- Select "**Manage Inventory**" and "**On-hand inventory**"



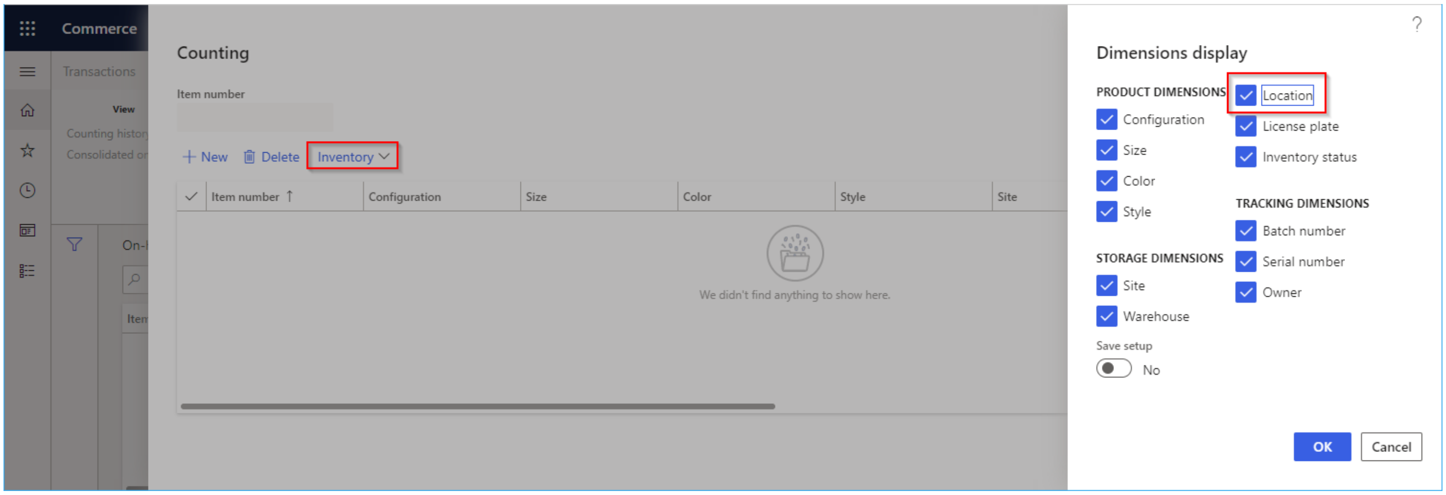
All field values required for validation are specified for product 75001.

Released product details

75001 : Ultralight Superpacker Men's Backpack

Overdelivery: 0.00	Base price: Purchase price	Price quantity: 1.00	Event duration: 0
Underdelivery: 0.00	Contribution ratio: 0.00	PRICE ADJUST	SELL DATES
TAXATION	Date of price: 3/25/2020	Allow price adjust: Yes	Sell start date:
Item sales tax group: RP			Sell end date:

- Select "**Quantity Adjustment**" at the top
- Open "**Inventory**" dropdown menu then “**Display Dimensions**” and check the "**Location**" checkbox and click “**OK**”



Dimensions display

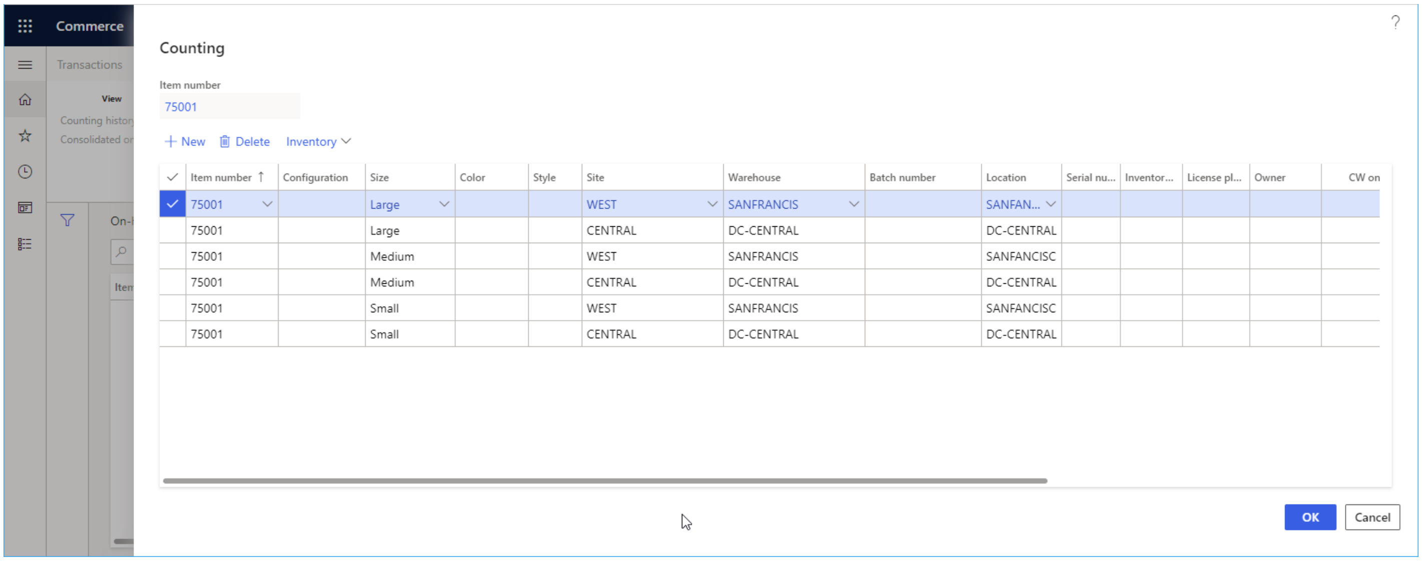
PRODUCT DIMENSIONS
☑ Configuration
☑ Size
☑ Color
☑ Style

STORAGE DIMENSIONS
☑ Site
☑ Warehouse
☑ Location
☑ License plate
☑ Inventory status

TRACKING DIMENSIONS
☑ Batch number
☑ Serial number
☑ Owner

Save setup: No

- "+**New**"
  - choose **Size=Small**
  - choose **Site**="**Central**"
  - **Warehouse**="**DC-Central**"
  - **Location**="**DC-Central**" (this is the warehouse where our Online channel is hooked up to)
  - Add a quantity = some number ex: 55 for each
- Repeat for Size = **Medium**
- Repeat for Size = **Large**
- Optional: Add inventory for Site = ”West” and Warehouse=”SanFrancis” this can be useful for BOPIS scenarios (Buy online pickup in store)
- Select “**OK**” to save



Counting

Item number
75001

✓	Item number ↑	Configuration	Size	Color	Style	Site	Warehouse	Batch number	Location	Serial nu...	Inventor...	License pl...	Owner	CW on
✓	75001		Large			WEST	SANFRANCIS		SANFAN...					
	75001		Large			CENTRAL	DC-CENTRAL		DC-CENTRAL					
	75001		Medium			WEST	SANFRANCIS		SANFANCISC					
	75001		Medium			CENTRAL	DC-CENTRAL		DC-CENTRAL					
	75001		Small			WEST	SANFRANCIS		SANFANCISC					
	75001		Small			CENTRAL	DC-CENTRAL		DC-CENTRAL					

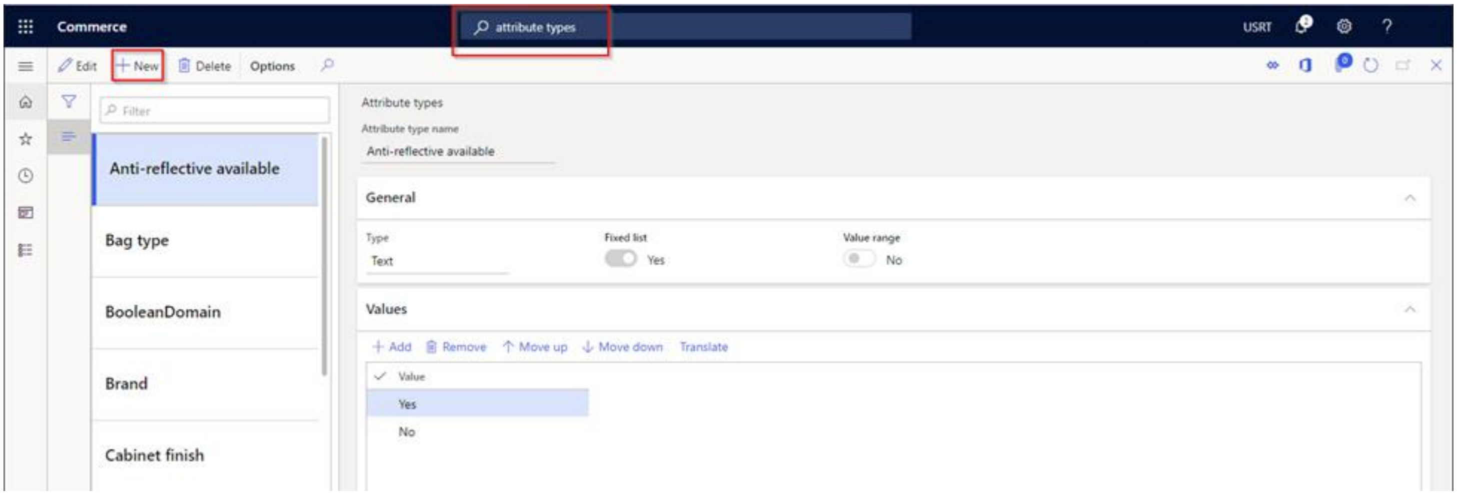
CREATE NEW ATTRIBUTES

We are going to create two new attributes for our backpack, one for the usage and one for backpack frame type. This will allow us to see refiners on the category page. First, we will create the two new attribute types, the attributes and an attribute group to group them together.

CREATE NEW ATTRIBUTE TYPES

In search box search for "**Attribute types**" and select the page

- Select "**New**" to create a new attribute type



Anti-reflective available
Bag type
BooleanDomain
Brand
Cabinet finish

Attribute types
Attribute type name
Anti-reflective available

General
Type
Text
Fixed list
Yes
Value range
No

Values
Value
Yes
No

Below example shows setting a “Backpack usage”:

- "**Attribute type name**" = "**Backpack usage**"
- **Type** = "**Text**"
- **Fixed list** = "**Yes**"
- Values select "**Add**" and add the following 4 values

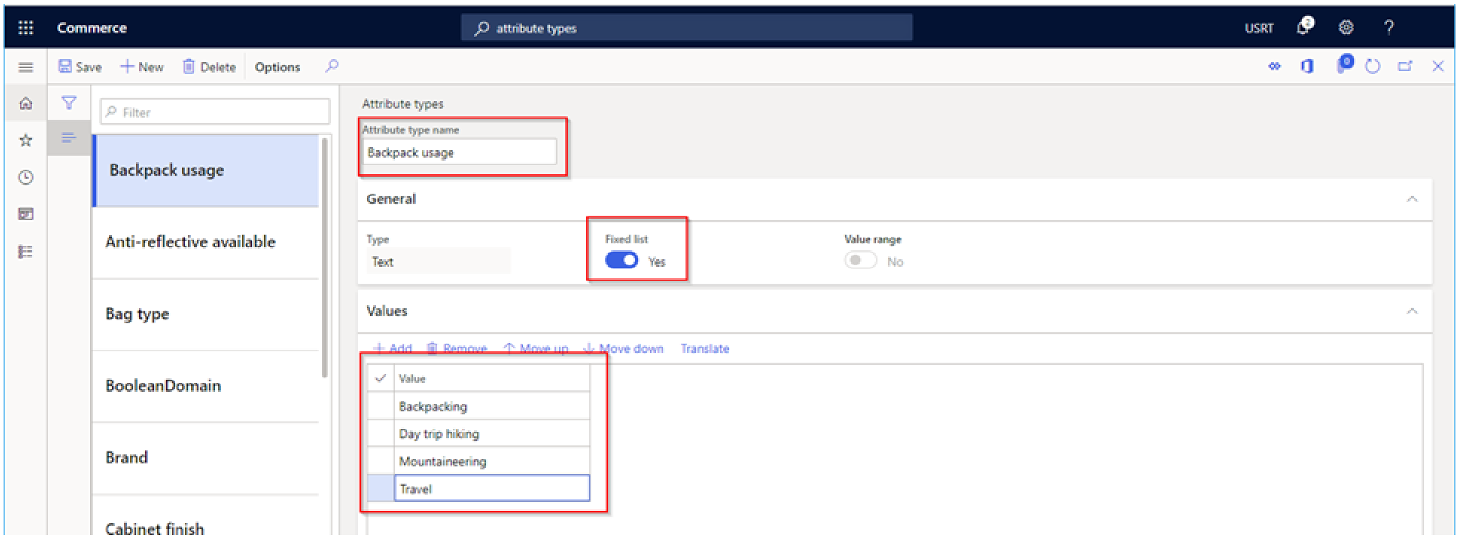
Backpacking

Day trip hiking

Mountaineering

Travel

- Click **“Save”**



Attribute types

Attribute type name: Backpack usage

General

Type: Text

Fixed list: Yes

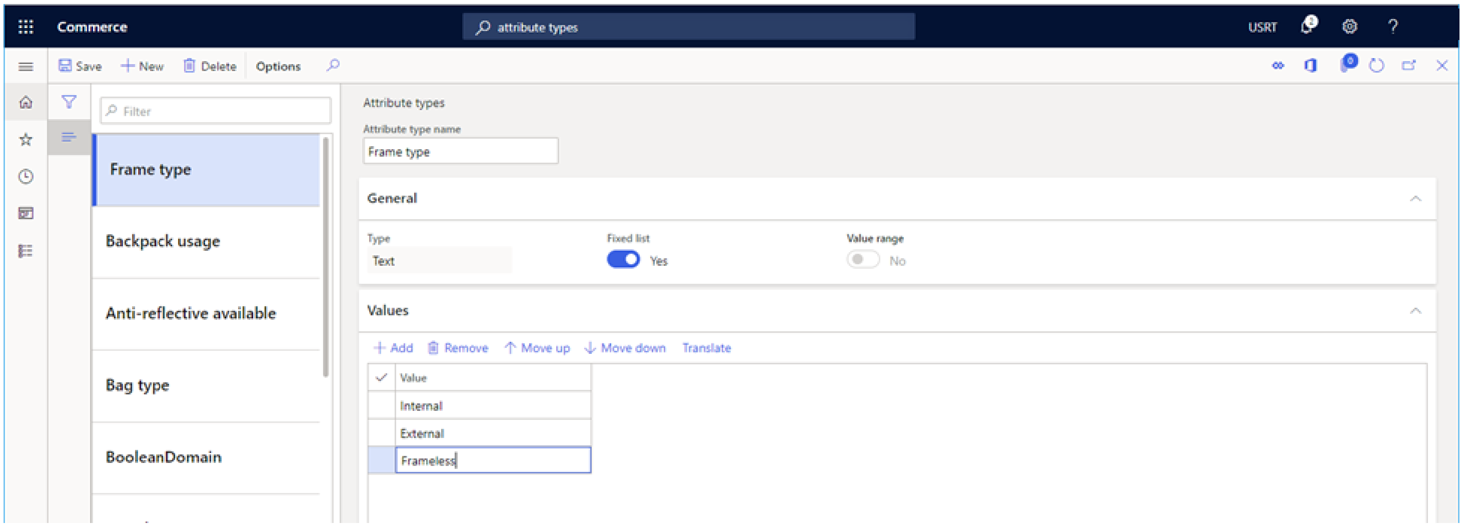
Value range: No

Values

✓	Value
	Backpacking
	Day trip hiking
	Mountaineering
	Travel

Below example shows setting a “Frame type”:

- Select "**New**" to create a new attribute type
- Set "**Attribute type name**" = "**Frame type**"
- **Type** = "**Text**"
- **Fixed list** = "**Yes**"
- Values select "**Add**" and add the following 4 values
    - **Internal**
    - **External**
    - **Frameless**
- Click **“Save”**



Frame type
Backpack usage
Anti-reflective available
Bag type
BooleanDomain

Attribute types

Attribute type name: Frame type

General

Type: Text
Fixed list: Yes
Value range: No

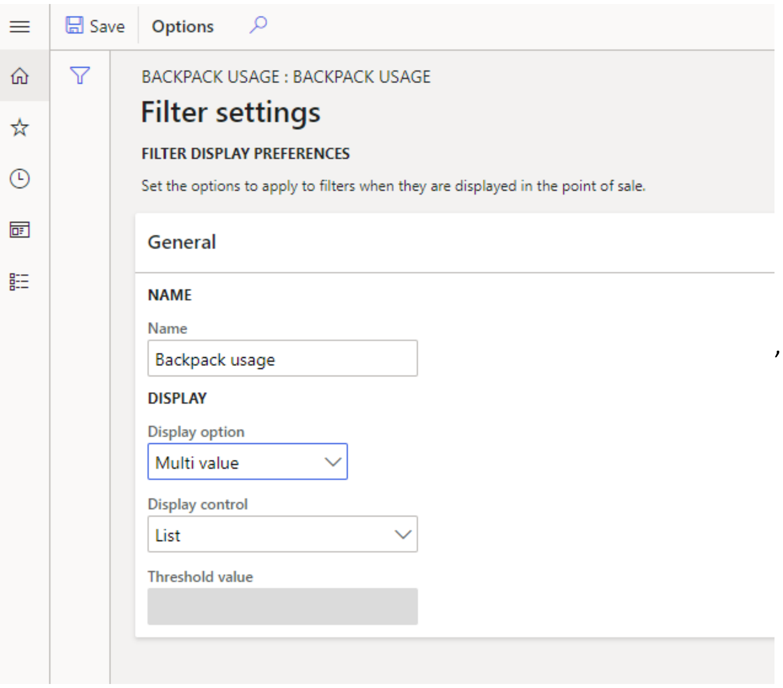
Values

Value
Internal
External
Frameless

CREATE ATTRIBUTES

The below example shows how to create a new attribute using the type we created above.

- Search and navigate to "**Attributes**" page
- Select "**New**" to create a new attribute
- **Name** = "**Backpack usage**"
- **Attribute type** = "**Backpack usage**"
- **Friendly name** = "**Usage**"
    - Click "**Filter Settings**" button and change “**Display option**” to “**Multi value**” then save and close



BACKPACK USAGE : BACKPACK USAGE

Filter settings

FILTER DISPLAY PREFERENCES

Set the options to apply to filters when they are displayed in the point of sale.

General

NAME

Name: Backpack usage

DISPLAY

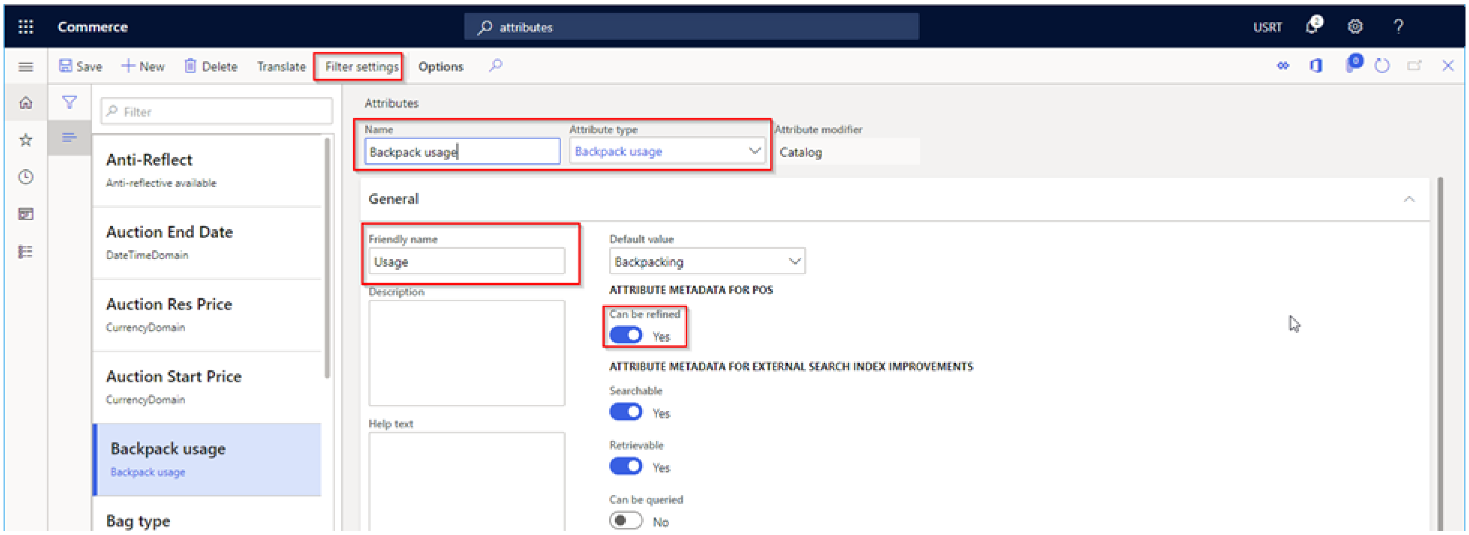
Display option: Multi value

Display control: List

Threshold value:

,

- **Default value** = "**Backpacking**"
- **Can be refined** = "**Yes**" (will show error if above step is not done)
- Click “**Save**”



Anti-Reflect
Anti-reflective available

Auction End Date
DateTimeDomain

Auction Res Price
CurrencyDomain

Auction Start Price
CurrencyDomain

Backpack usage
Backpack usage

Bag type

Attributes

Name: Backpack usage

Attribute type: Backpack usage

Attribute modifier: Catalog

General

Friendly name: Usage

Description

Help text

Default value: Backpacking

ATTRIBUTE METADATA FOR POS

Can be refined: Yes

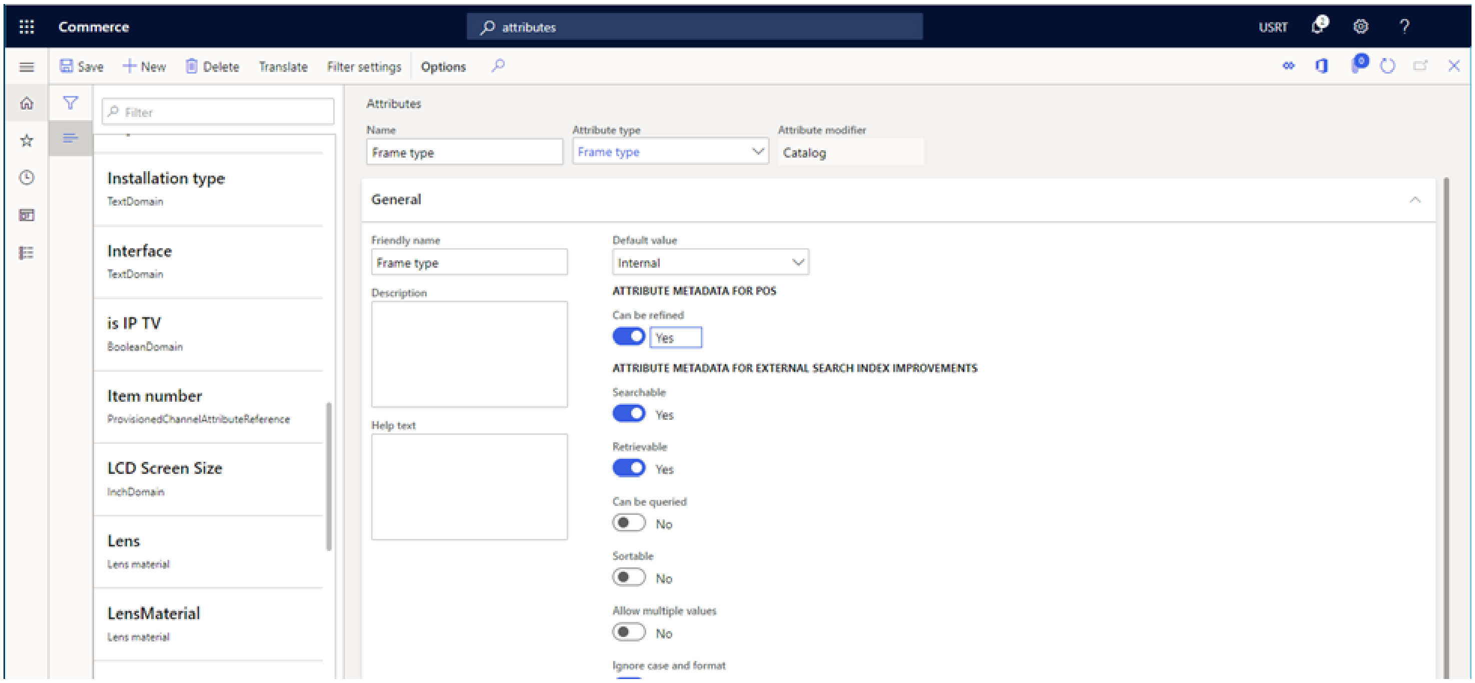
ATTRIBUTE METADATA FOR EXTERNAL SEARCH INDEX IMPROVEMENTS

Searchable: Yes

Retrievable: Yes

Can be queried: No

- Check if “**Frame type**” exists if not create it
- Select "**New**" to create another attribute
- **Name** = "**Frame Type**"
- **Attribute type** = "**Frame Type**"
- **Friendly name** = “**Frame Type**"
- Click "**Filter Settings**" button and change “Display option” to “Multi value” then save and close
- **Default value** = "**Internal**"
- **Can be refined** = "**Yes**" (will show error if above step is not done)
- **Save**



Installation type
TextDomain

Interface
TextDomain

is IP TV
BooleanDomain

Item number
ProvisionedChannelAttributeReference

LCD Screen Size
InchDomain

Lens
Lens material

LensMaterial
Lens material

Attributes

Name	Attribute type	Attribute modifier
Frame type	Frame type	Catalog

General

Friendly name: Frame type

Description

Help text

Default value: Internal

ATTRIBUTE METADATA FOR POS

Can be refined: Yes

ATTRIBUTE METADATA FOR EXTERNAL SEARCH INDEX IMPROVEMENTS

Searchable: Yes

Retrievable: Yes

Can be queried: No

Sortable: No

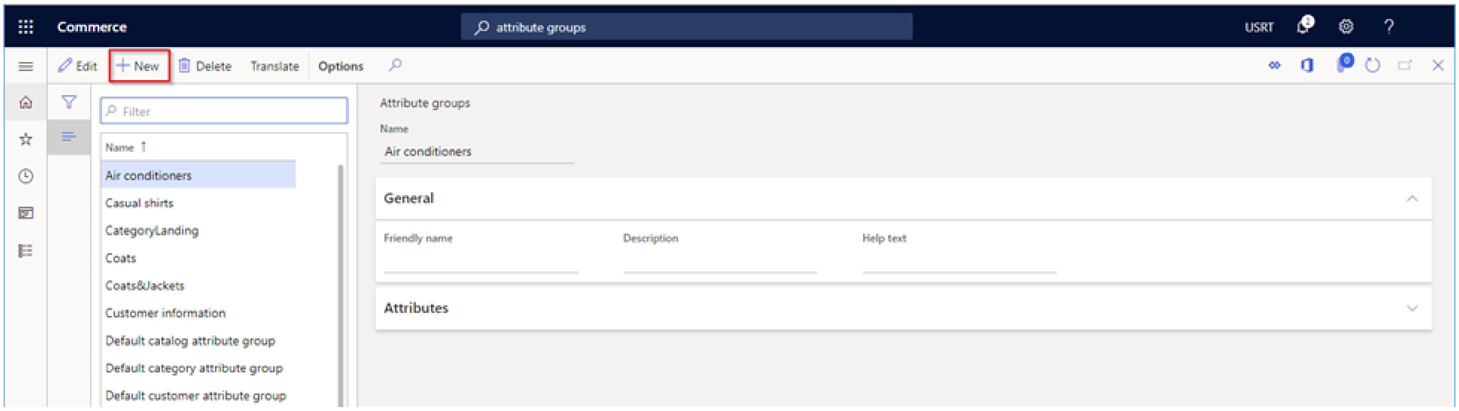
Allow multiple values: No

Ignore case and format

CREATE NEW ATTRIBUTE GROUP

We can group attributes together; this will make it easier later to apply a group of attributes to a single product and a hierarchy node. Below is an example on creating an attribute group of the two attributes created above.

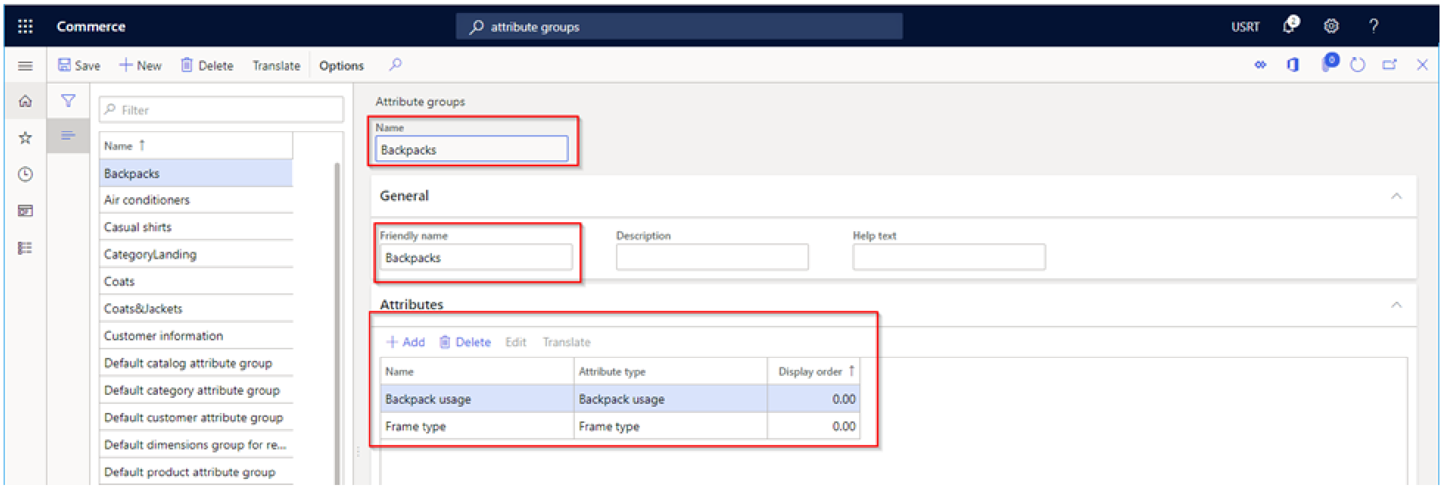
- Search and navigate to "**Attribute groups**"
- Select "**New**" button



Attribute groups

Name: Air conditioners

- Set **Name**="**Backpacks**"
- **Friendly name** = "**Backpacks**"
- Add the two previously created Attributes using the “**Add**” button in the Attributes section.
- Save and close



Attribute groups

Name: Backpacks

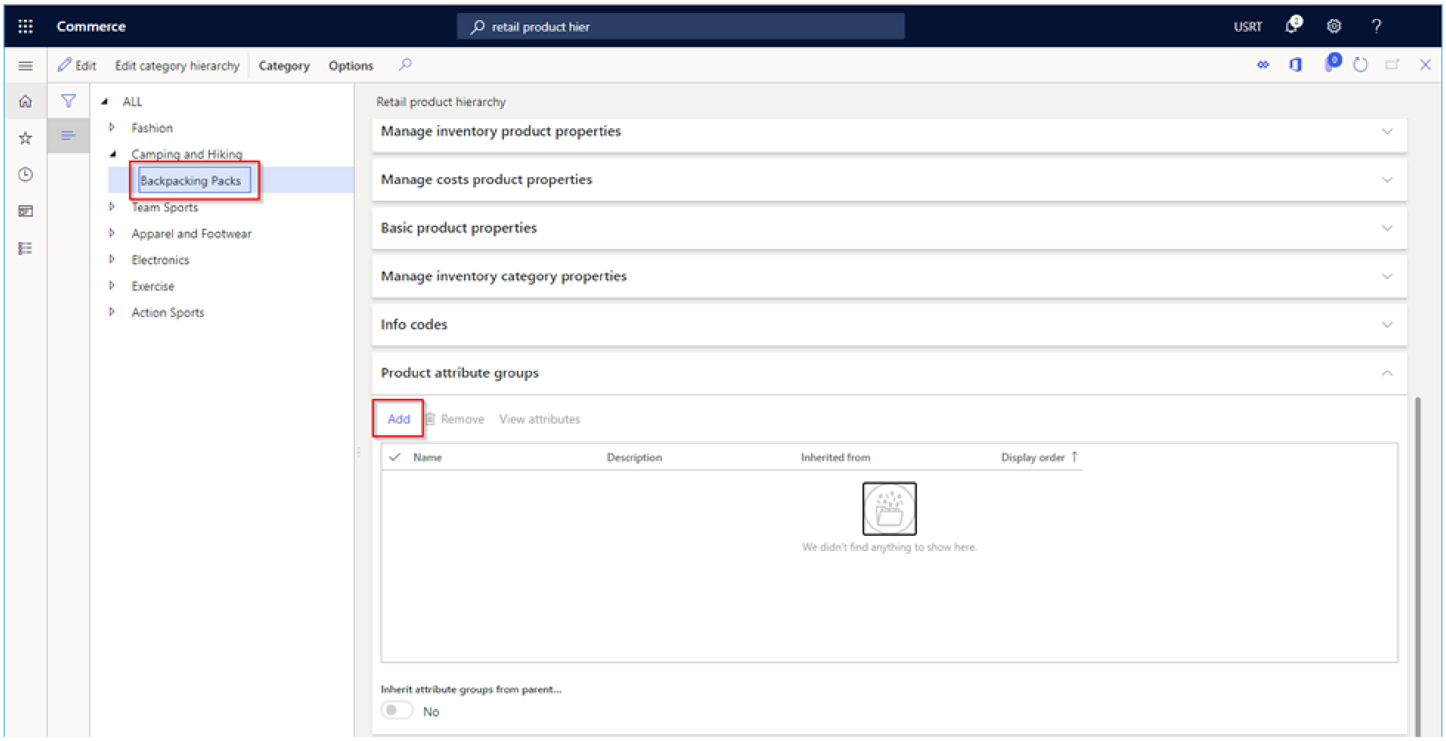
Friendly name: Backpacks

Name	Attribute type	Display order
Backpack usage	Backpack usage	0.00
Frame type	Frame type	0.00

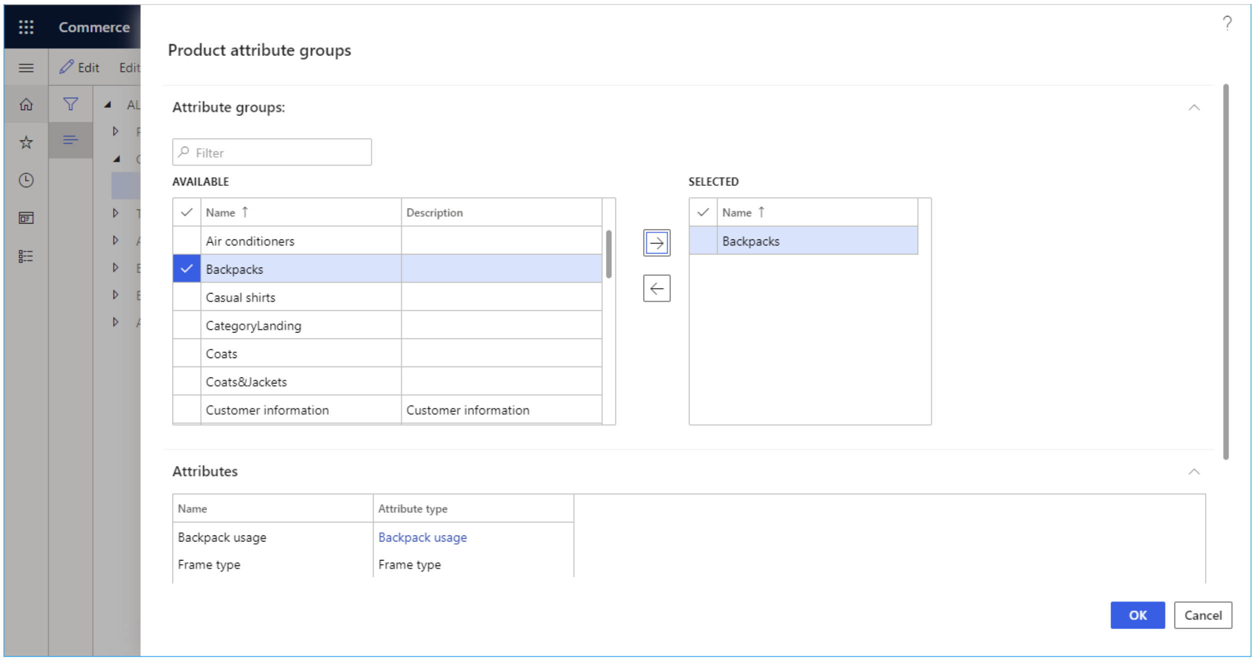
ADD ATTRIBUTES TO COMMERCE PRODUCT HIERARCHY

This section will now show how to add an attribute group to a product hierarchy, this will allow all products to inherit the attributes.

- Search and navigate to "**Commerce product hierarchy**"
- Select a category, for example below the "**Backpacking Packs**" node has been selected
- Under "**Product attribute groups**" section select "**Add**" button



- Select an attribute group to add, the below example shows a "**Backpacks**" attribute group being added.
- Select "**OK**"



Product attribute groups

Attribute groups:

Filter

AVAILABLE

✓	Name ↑	Description
	Air conditioners	
✓	Backpacks	
	Casual shirts	
	CategoryLanding	
	Coats	
	Coats&Jackets	
	Customer information	Customer information

SELECTED

✓	Name ↑
	Backpacks

Attributes

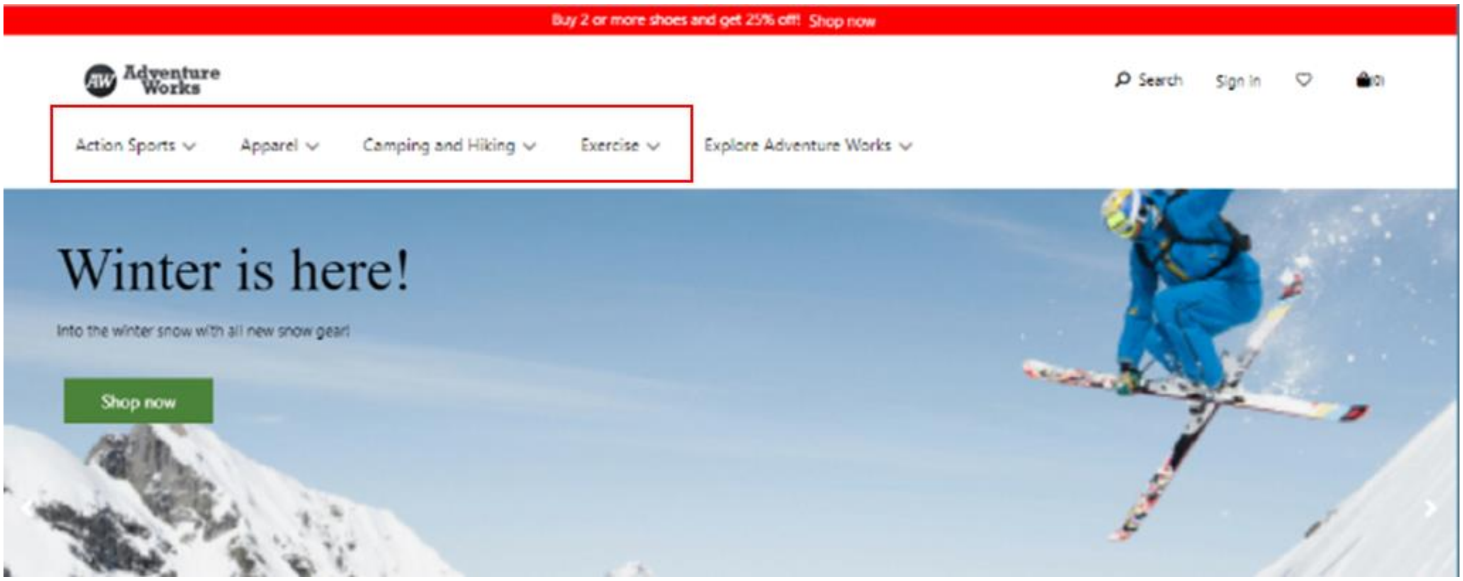
Name	Attribute type
Backpack usage	Backpack usage
Frame type	Frame type

ADDING ADDITIONAL PRODUCTS

Optionally you can follow the above steps to create more products. The products could leverage existing attributes and attribute groups if desired.

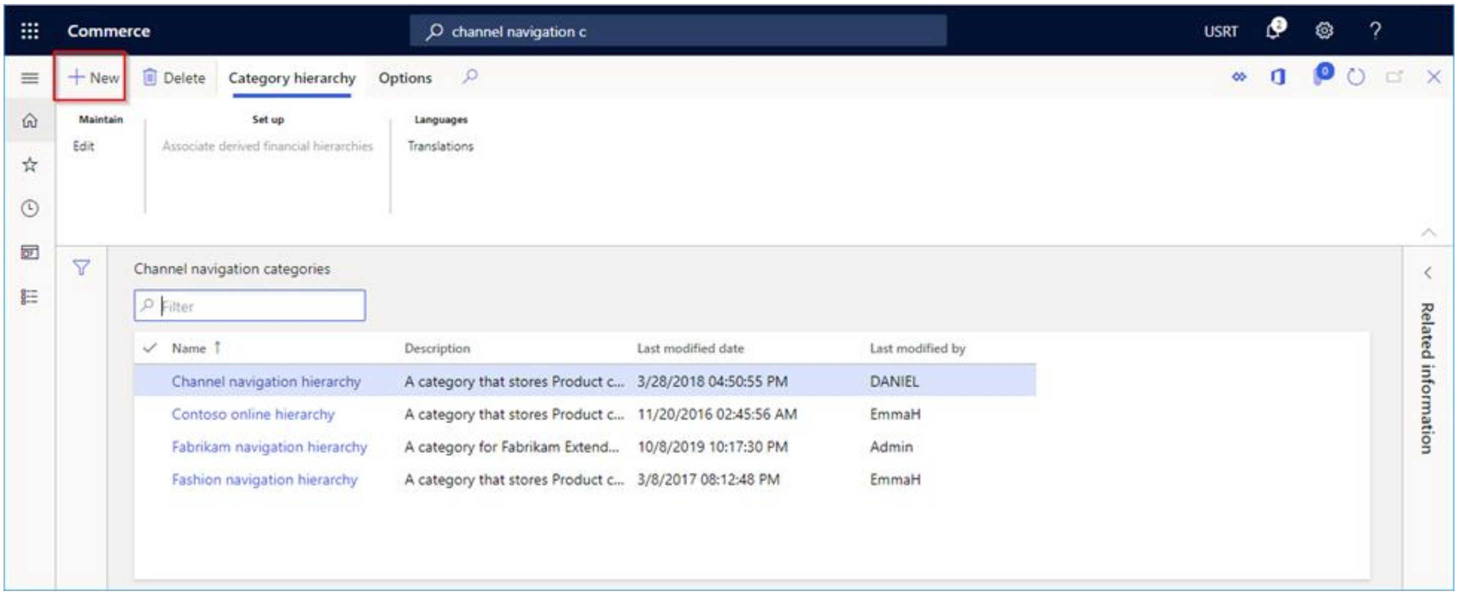
CREATE NEW CHANNEL NAVIGATION HIERARCHY

Our new channel still needs a product navigation hierarchy that will be presented on our e-Commerce site which is different than the global product hierarchy we have used up until this point. Notice this hierarchy in the below image on the e-Commerce site.



Follow the below steps to create a new navigation hierarchy. Each channel can have a unique channel hierarchy with.

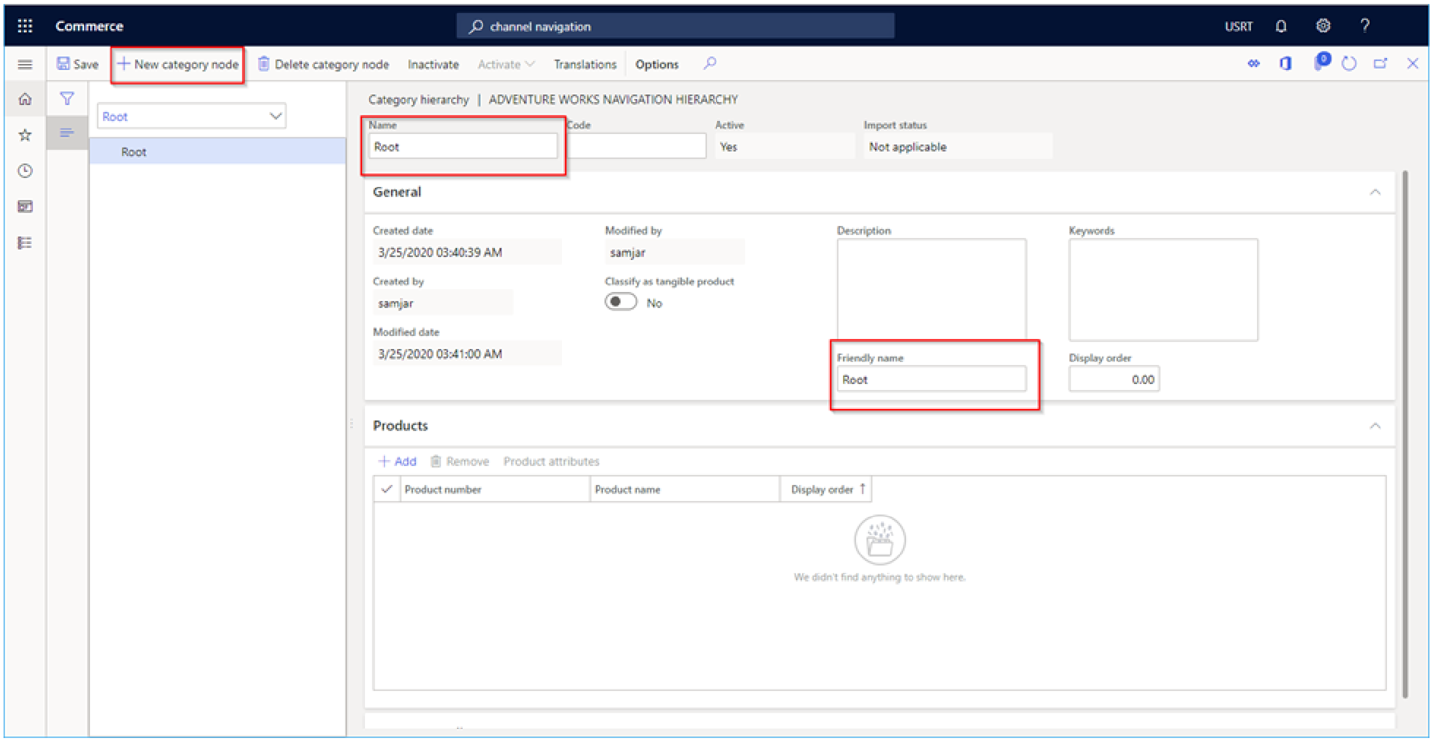
- Search for the "**Channel navigation categories**" page
- Select the "**New**" button



Channel navigation categories

Name ↑	Description	Last modified date	Last modified by
Channel navigation hierarchy	A category that stores Product c...	3/28/2018 04:50:55 PM	DANIEL
Contoso online hierarchy	A category that stores Product c...	11/20/2016 02:45:56 AM	EmmaH
Fabrikam navigation hierarchy	A category for Fabrikam Extend...	10/8/2019 10:17:30 PM	Admin
Fashion navigation hierarchy	A category that stores Product c...	3/8/2017 08:12:48 PM	EmmaH

- Provide a name. In the below example we will use "**Adventure Works navigation hierarchy**" and description “**A category for Adventure Works online store.**” then click the “**Save**” button
- We will create a root node first – this root node will not show up in our e-Commerce site, select "**New category node**"
- Set a name such as "Root" and a friendly name such as "Root" (feel free to use any name here – it will only show up here not in the end e-commerce site)



Root

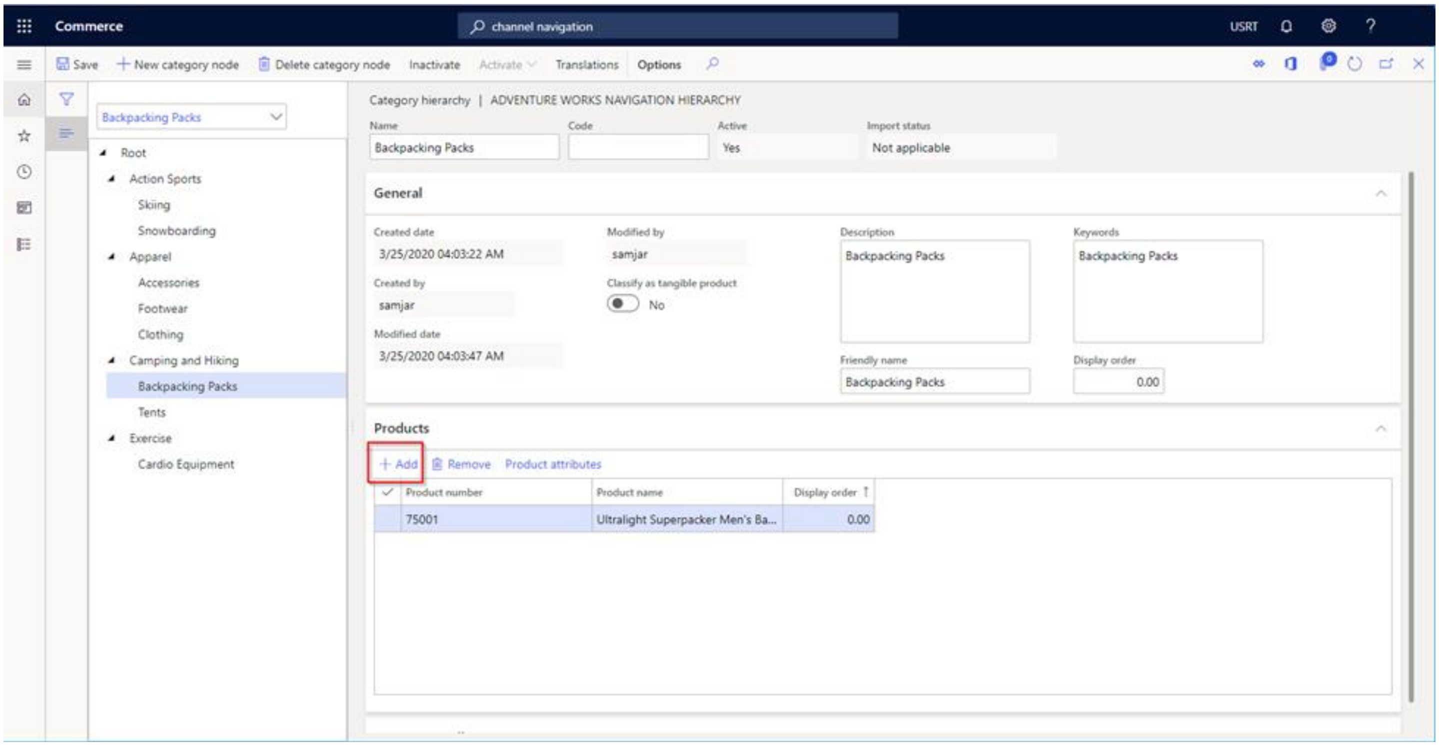
Category hierarchy | ADVENTURE WORKS NAVIGATION HIERARCHY

Name: Root
Code:
Active: Yes
Import status: Not applicable

Created date: 3/25/2020 03:40:39 AM
Modified by: samjar
Description:
Keywords:
Created by: samjar
Classify as tangible product: No
Modified date: 3/25/2020 03:41:00 AM
Friendly name: Root
Display order: 0.00

- Select the new “**Root**” node and add new nodes for each category with the "**New category node**" button.
- We’ll use the below example set of nodes representing a sporting goods site:
  - **Action Sports**
    - **Skiing**
    - **Snowboarding**
  - **Apparel**
    - **Accessories**
    - **Clothing**
    - **Footwear**
  - **Camping and Hiking**
    - **Backpacking Packs**
    - **Tents**
  - **Exercise**
    - **Cardio Equipment**

Next, we'll add your new product to the appropriate node(s). For example, select the "Backpacking Packs" node and expand the "Products" section and click the "**Add**" button and add the products.



Category hierarchy | ADVENTURE WORKS NAVIGATION HIERARCHY

Name	Code	Active	Import status
Backpacking Packs		Yes	Not applicable

General

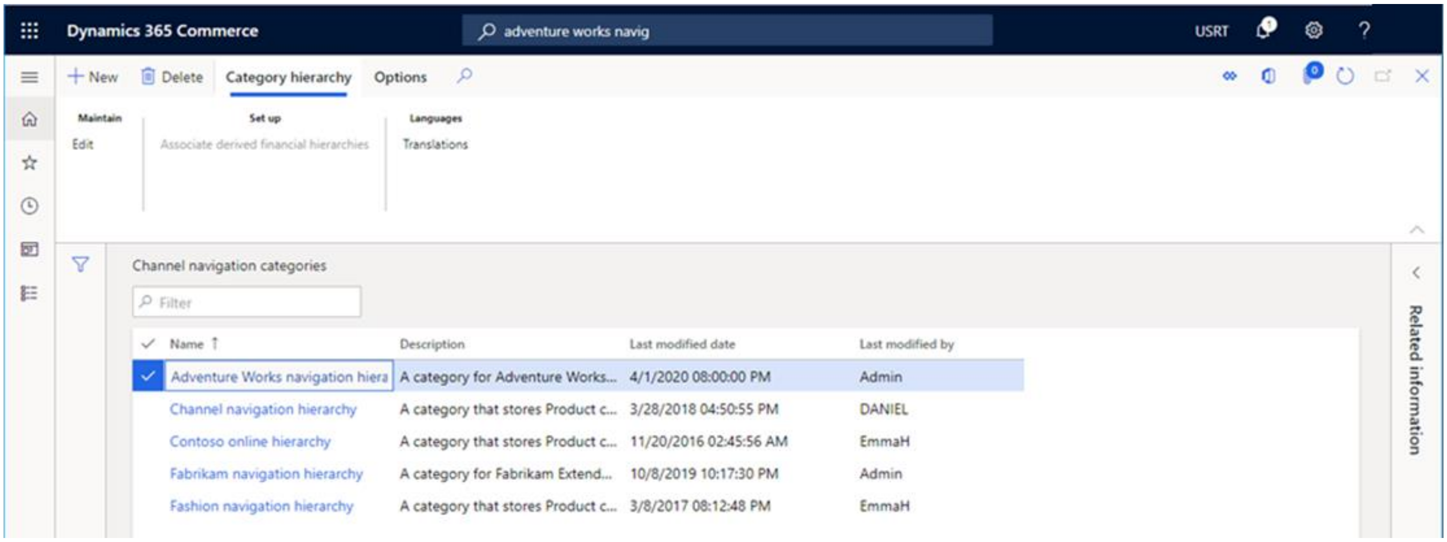
Created date: 3/25/2020 04:03:22 AM
Modified by: samjar
Description: Backpacking Packs
Keywords: Backpacking Packs
Created by: samjar
Classify as tangible product: No
Modified date: 3/25/2020 04:03:47 AM
Friendly name: Backpacking Packs
Display order: 0.00

Products

Product number	Product name	Display order ↑
75001	Ultralight Superpacker Men's Ba...	0.00

Next, we will add the product attribute groups to the node.

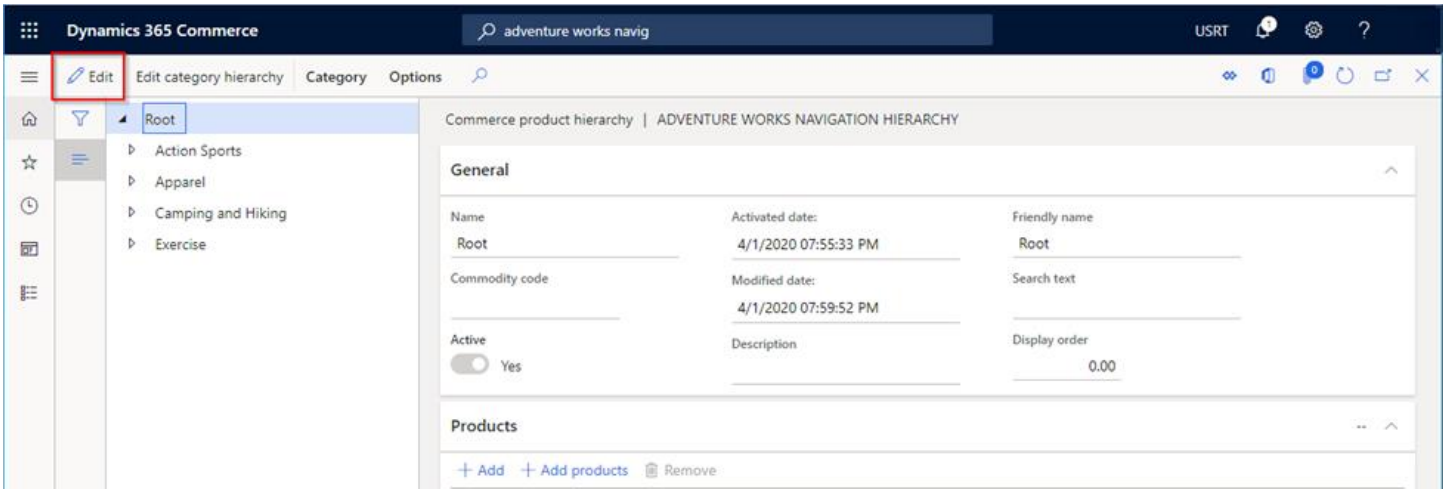
Close this page to go back to get back to **channel navigation categories** page as shown below.



Channel navigation categories

Name ↑	Description	Last modified date	Last modified by
Adventure Works navigation hiera	A category for Adventure Works...	4/1/2020 08:00:00 PM	Admin
Channel navigation hierarchy	A category that stores Product c...	3/28/2018 04:50:55 PM	DANIEL
Contoso online hierarchy	A category that stores Product c...	11/20/2016 02:45:56 AM	EmmaH
Fabrikam navigation hierarchy	A category for Fabrikam Extend...	10/8/2019 10:17:30 PM	Admin
Fashion navigation hierarchy	A category that stores Product c...	3/8/2017 08:12:48 PM	EmmaH

- Select the new navigation category we just created and click the **edit** button.

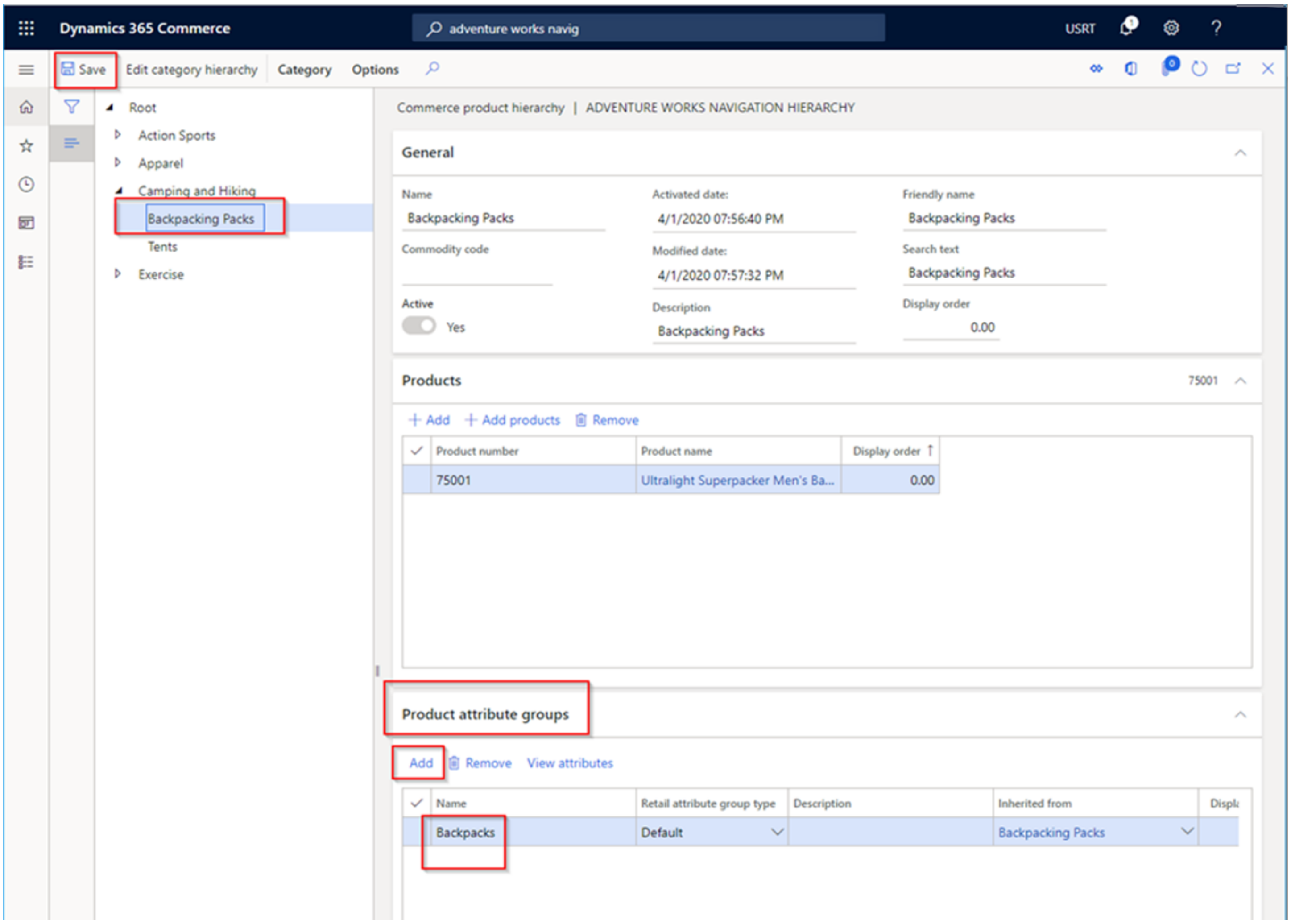


Commerce product hierarchy | ADVENTURE WORKS NAVIGATION HIERARCHY

General

Name: Root
Activated date: 4/1/2020 07:55:33 PM
Friendly name: Root
Commodity code:
Modified date: 4/1/2020 07:59:52 PM
Search text:
Active: Yes
Description:
Display order: 0.00

- Select the new node we previously added a product to, “Backpacking Packs” in previous example.
- Expand the “**Product attribute groups**” section and click the “**add**” button to add the “Backpacks” attribute group created earlier.
- Click “**Save**” when complete. (You may see a warning; you can Ignore this warning notifying that you may want to add the attributes elsewhere.)



Commerce product hierarchy | ADVENTURE WORKS NAVIGATION HIERARCHY

General

Name: Backpacking Packs

Activated date: 4/1/2020 07:56:40 PM

Friendly name: Backpacking Packs

Commodity code:

Modified date: 4/1/2020 07:57:32 PM

Search text: Backpacking Packs

Active: Yes

Description: Backpacking Packs

Display order: 0.00

Products 75001

Product number	Product name	Display order
75001	Ultralight Superpacker Men's Ba...	0.00

Product attribute groups

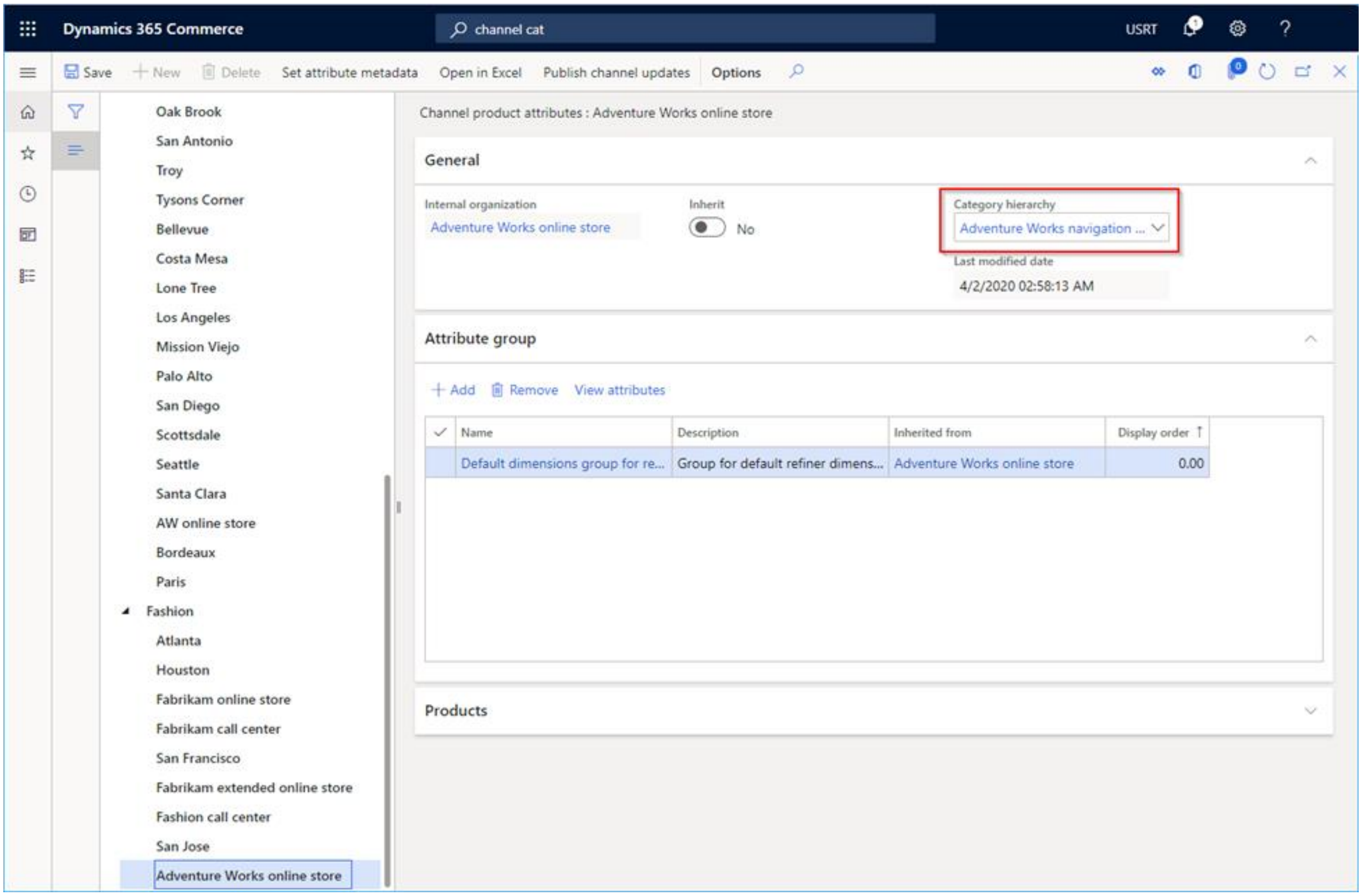
Name	Retail attribute group type	Description	Inherited from	Displ
Backpacks	Default		Backpacking Packs	

CONFIGURE CHANNEL TO USE THE NEW CHANNEL HIERARCHY

Now that we have created a channel navigation hierarchy, we will learn to associate it to the new channel we created.

Search for the "**Channel categories and product attributes**" page.

- Select the new online channel in the list that you want to configure and select the "**Edit**" button
- Change the category hierarchy to the new created in previous section



Channel product attributes : Adventure Works online store

Internal organization: Adventure Works online store

Inherit: No

Category hierarchy: Adventure Works navigation ...

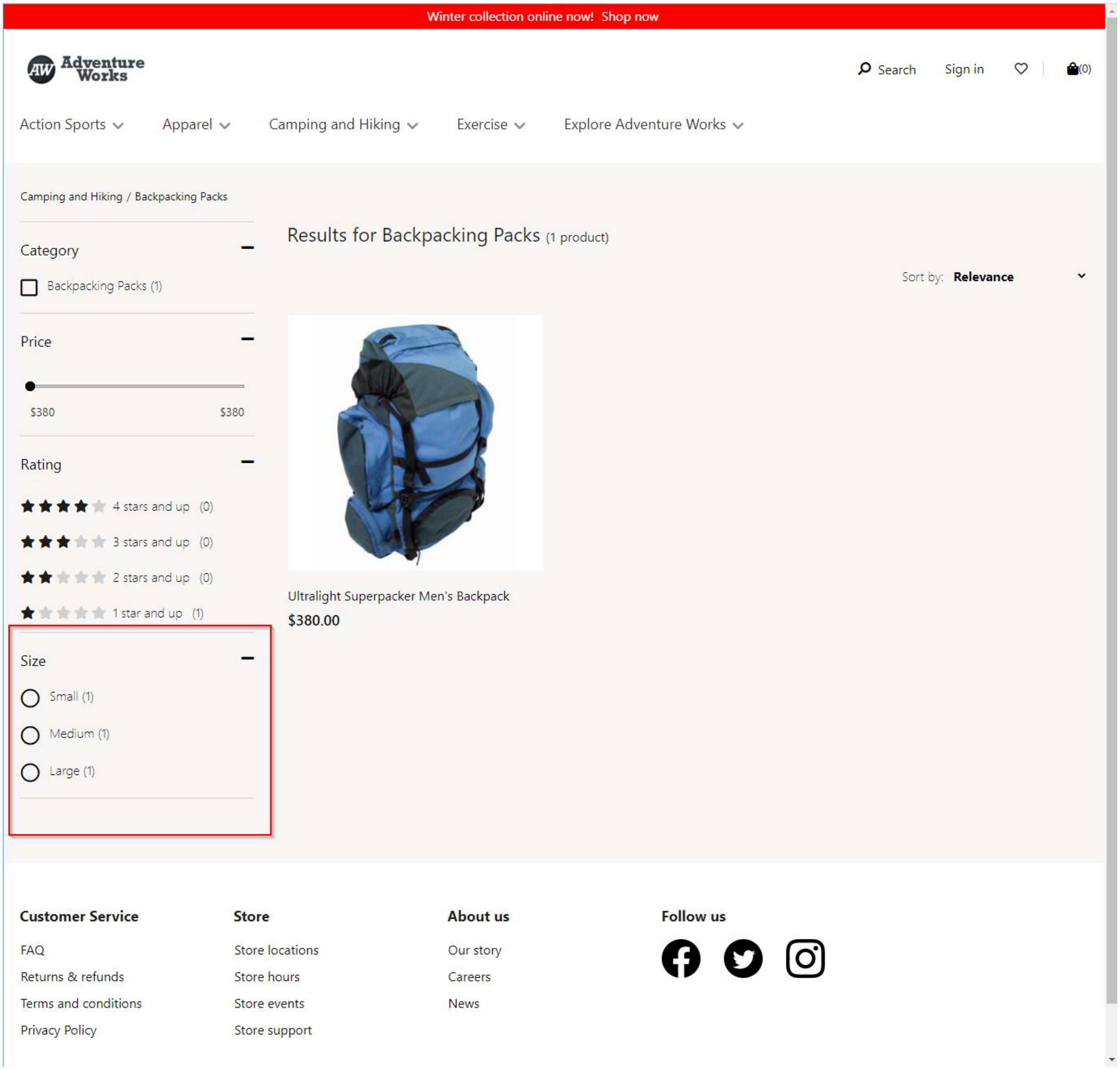
Last modified date: 4/2/2020 02:58:13 AM

Name	Description	Inherited from	Display order
Default dimensions group for re...	Group for default refiner dimens...	Adventure Works online store	0.00

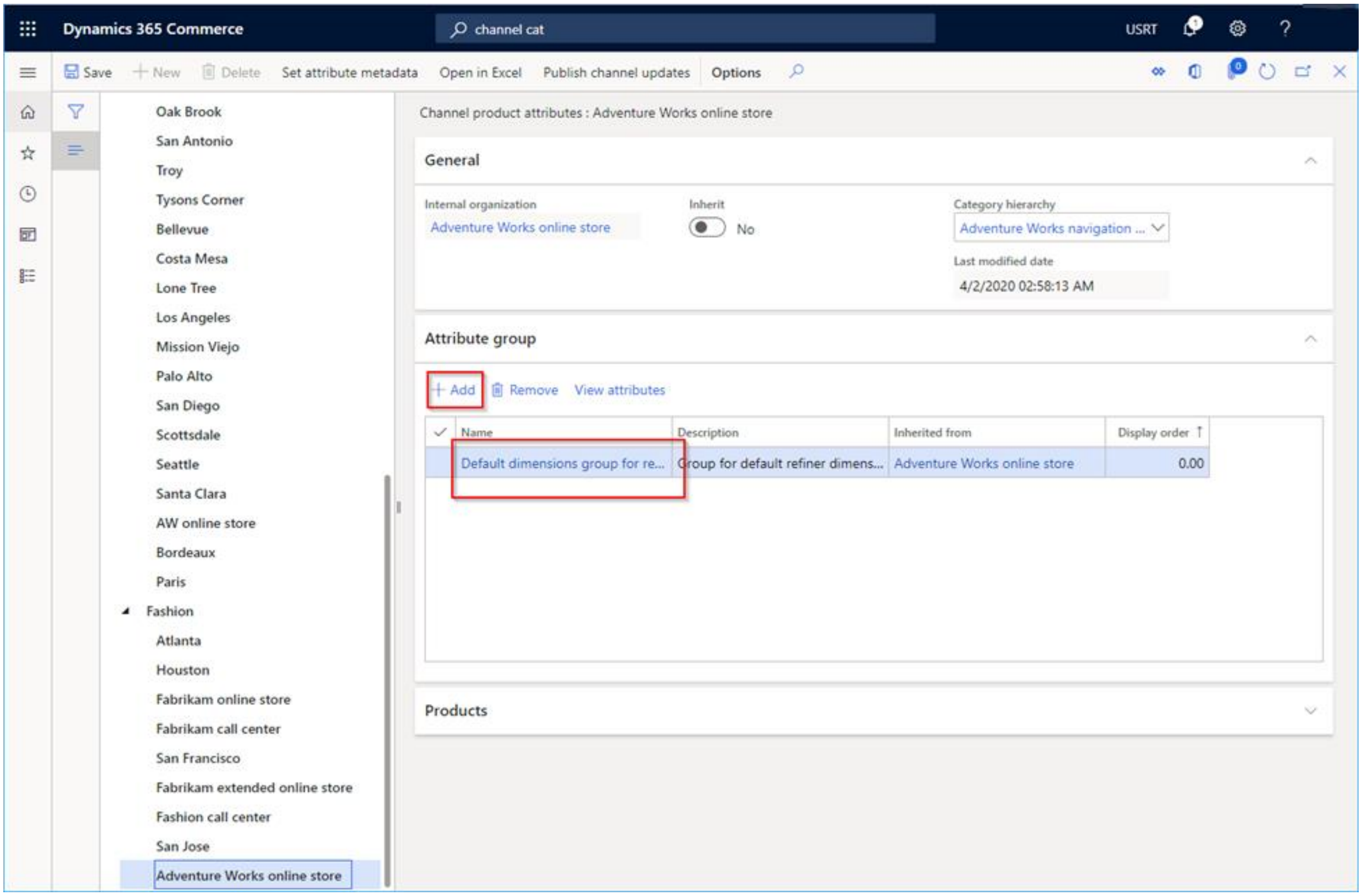
CONFIGURING CHANNEL PRODUCT ATTRIBUTES

We now want to ensure we have the appropriate attribute refiners for each category on our e-Commerce site. We will start by removing the default one that is added to a new channel and replace with a pre-defined one that is designed to show the product variants (size/color/style). Our example backpack product only defined size, so we will not see the color/style but if your other products have those variants they will show up.

By doing the below steps we will enable the standard refiners to show up in the category pages as shown below on our e-commerce site.



While still in the “**Channel categories and product attributes**” page from the previous section remove the “**SharepointProvisionedProductAttributeGroup**” and instead add the “**Default dimensions group for refiners**”.



Channel product attributes : Adventure Works online store

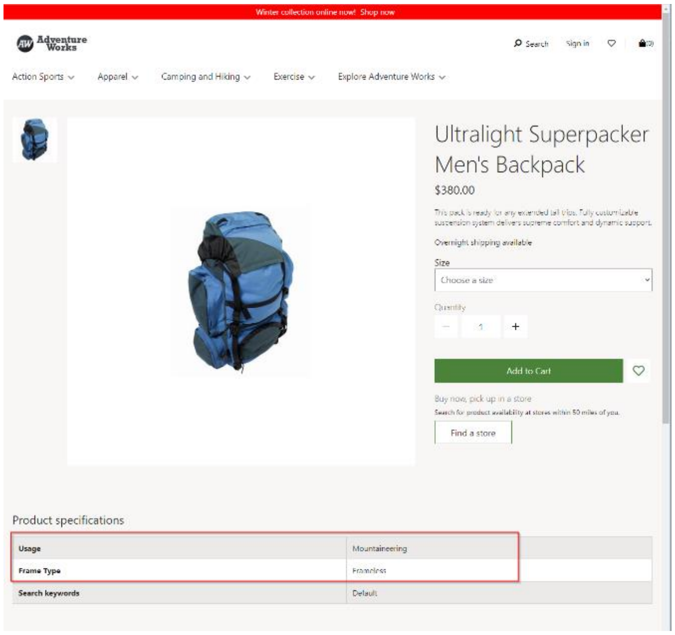
General

Internal organization: Adventure Works online store
Inherit: No
Category hierarchy: Adventure Works navigation ...
Last modified date: 4/2/2020 02:58:13 AM

Attribute group

Name	Description	Inherited from	Display order
Default dimensions group for re...	Group for default refiner dimens...	Adventure Works online store	0.00

Now we will enable the new attributes to also be refinable and this will ensure the attributes show up on the product details page as shown below.



Winter collection online now! Shop now

Adventure Works

Search Sign in

Action Sports Apparel Camping and Hiking Exercise Explore Adventure Works

Ultralight Superpacker Men's Backpack

$380.00

This pack is ready for any extended fall trips. Fully customizable suspension system delivers supreme comfort and dynamic support.

Overnight shipping available

Size: Choose a size

Quantity: 1

Add to Cart

Buy now, pick up in a store

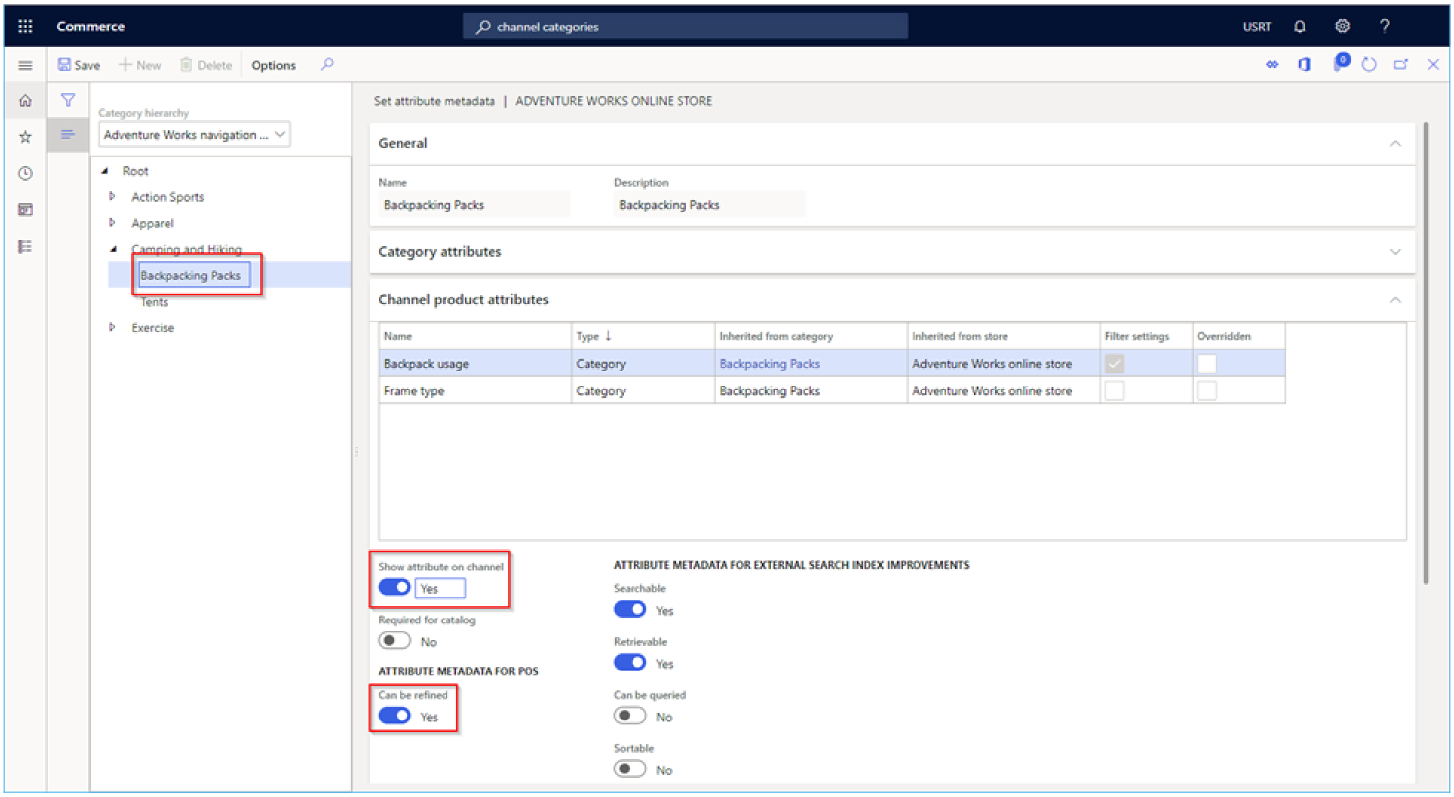
Search for product availability at stores within 50 miles of you.

Find a store

Product specifications

Usage	Mountaineering
Frame Type	Frameless
Search keywords	Default

- Select the “Set attribute metadata” at the top
- Select a node you want to modify (for example “**Backpacking Packs**” under “**Camping and Hiking**”)
- You should now see the previously created attributes, for each “**Channel product attributes**” (example: Backpack usage and Frame Type) set “**Show attribute on channel**” to “**Yes**” and make sure the “**Can be refined**” is set to “**Yes**” as well.
- The same can be done to all other categories to enable refiners.
- Select “**Save**” and close



Set attribute metadata | ADVENTURE WORKS ONLINE STORE

General

Name: Backpacking Packs

Description: Backpacking Packs

Category attributes

Channel product attributes

Name	Type ↓	Inherited from category	Inherited from store	Filter settings	Overridden
Backpack usage	Category	Backpacking Packs	Adventure Works online store	☑	☐
Frame type	Category	Backpacking Packs	Adventure Works online store	☐	☐

Show attribute on channel: Yes

Required for catalog: No

ATTRIBUTE METADATA FOR POS

Can be refined: Yes

ATTRIBUTE METADATA FOR EXTERNAL SEARCH INDEX IMPROVEMENTS

Searchable: Yes

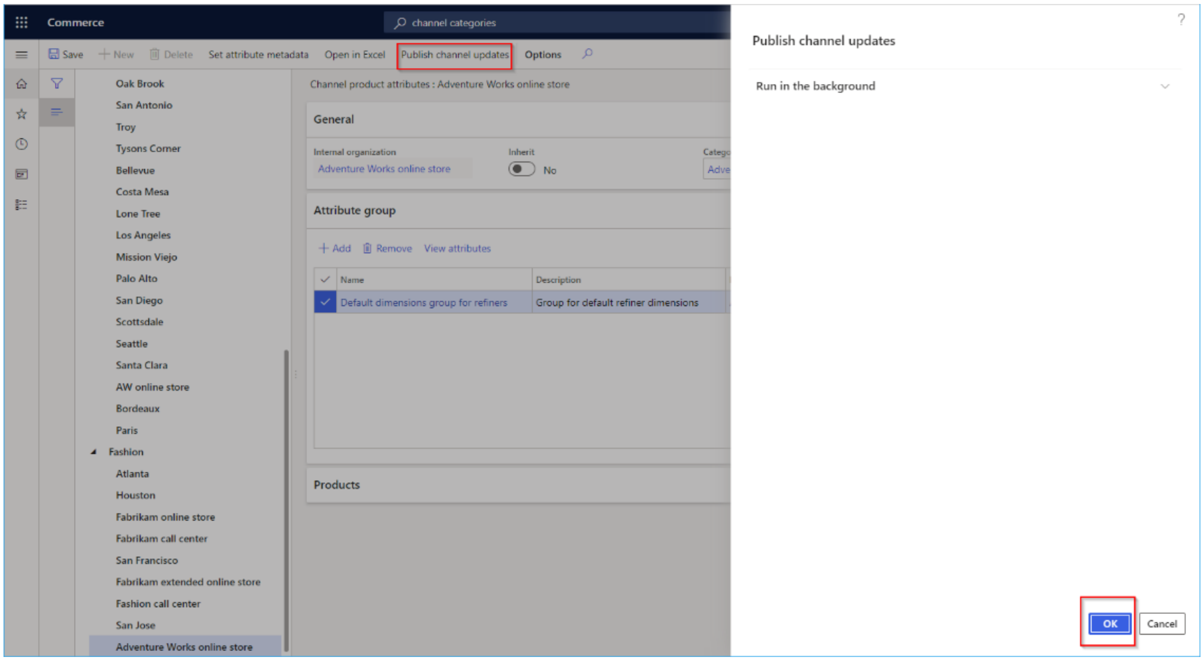
Retrievable: Yes

Can be queried: No

Sortable: No

We can now publish the changes.

- Press "**Publish channel updates**" followed by “**OK**”



Channel product attributes : Adventure Works online store

Internal organization: Adventure Works online store

Inherit: No

Name	Description
Default dimensions group for refiners	Group for default refiner dimensions

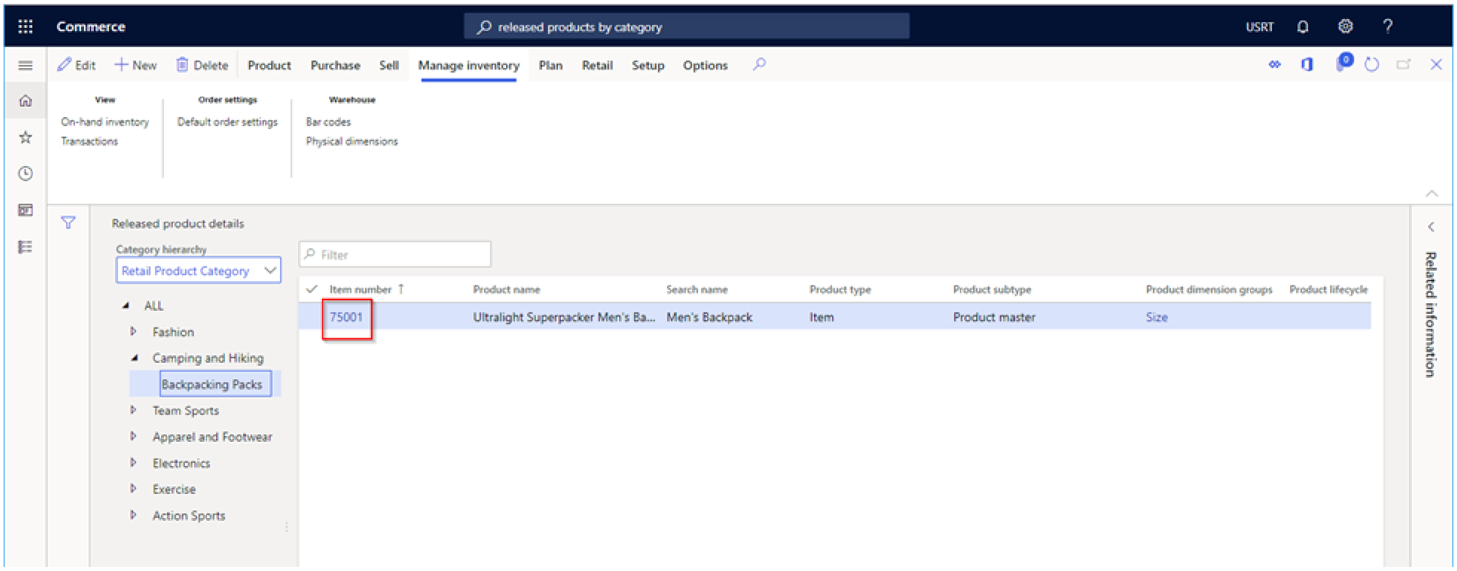
Publish channel updates

Run in the background

Configure Attributes on new product(s)

Now that we have created attributes on the product and node, we need to set the specific value for the new products we've added. The below example shows how to do this for the example products added earlier.

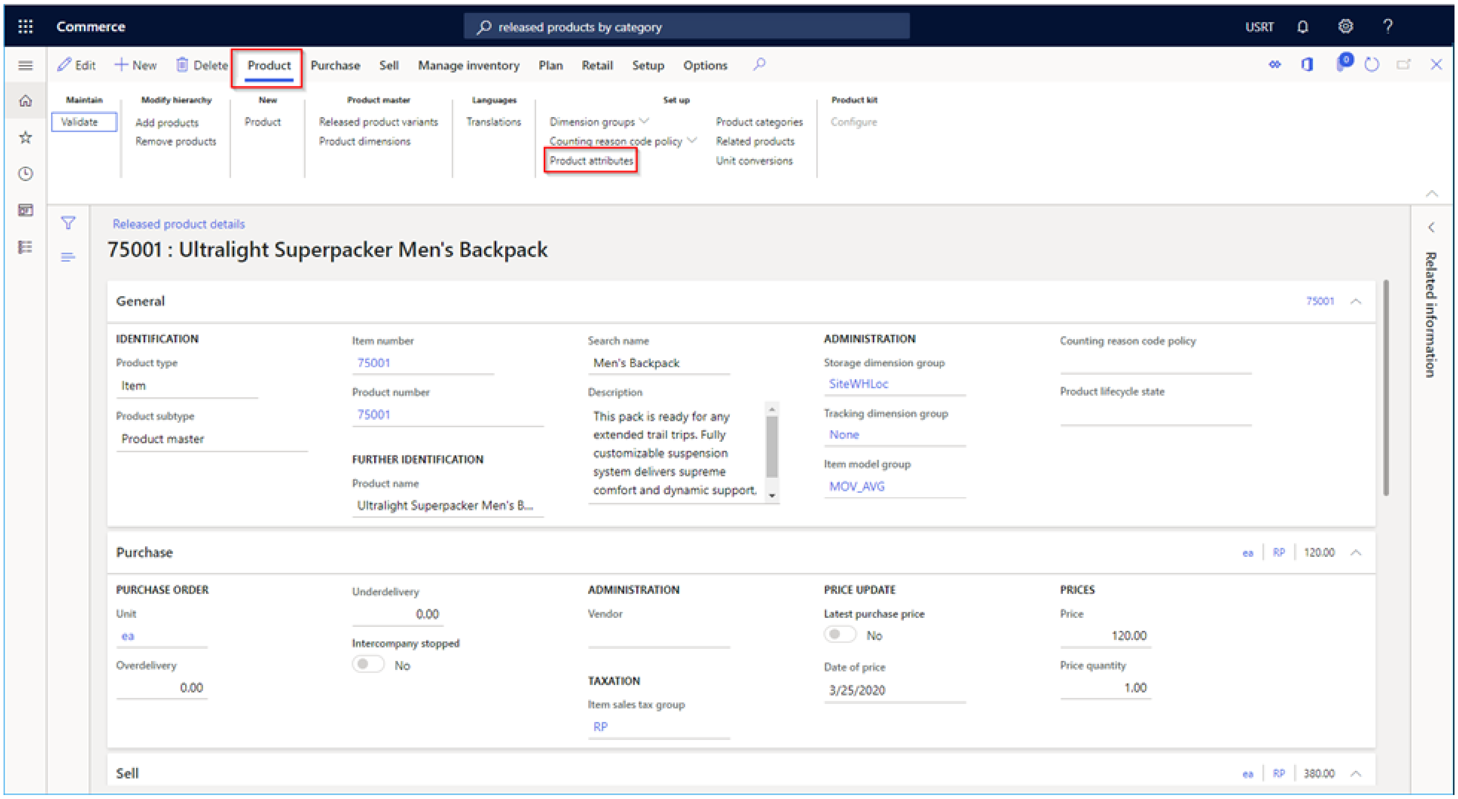
- Type in "**Released products by category**" in the search box and navigate to that page
- Select the new product link "**75001**" under "**Camping and Hiking**" -> "**Backpacking Packs**"



Category hierarchy: Retail Product Category

Item number	Product name	Search name	Product type	Product subtype	Product dimension groups	Product lifecycle
75001	Ultralight Superpacker Men's Ba...	Men's Backpack	Item	Product master	Size	

- Select the "**Product"** tab at the top
- Select "**Product attributes**"



Released product details

75001 : Ultralight Superpacker Men's Backpack

General 75001

IDENTIFICATION

Product type: Item

Product subtype: Product master

Item number: 75001

Product number: 75001

FURTHER IDENTIFICATION

Product name: Ultralight Superpacker Men's B...

Search name: Men's Backpack

Description: This pack is ready for any extended trail trips. Fully customizable suspension system delivers supreme comfort and dynamic support.

ADMINISTRATION

Storage dimension group: SiteWHLoc

Tracking dimension group: None

Item model group: MOV_AVG

Counting reason code policy

Product lifecycle state

Purchase ea | RP | 120.00

PURCHASE ORDER

Unit: ea

Overdelivery: 0.00

Underdelivery: 0.00

Intercompany stopped: No

ADMINISTRATION

Vendor

TAXATION

Item sales tax group: RP

PRICE UPDATE

Latest purchase price: No

Date of price: 3/25/2020

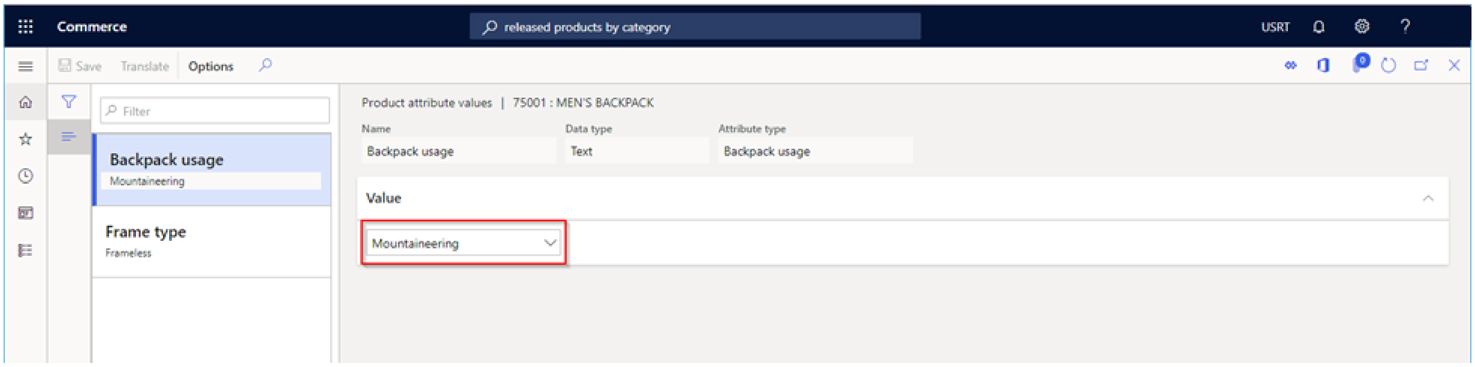
PRICES

Price: 120.00

Price quantity: 1.00

Sell ea | RP | 380.00

- Select “**Mountaineering**” for “**Backpack Usage**”
- Select “**Frameless**” for “**Frame Type**”
- Close page
- Do the same for all other new products



Product attribute values | 75001 : MEN'S BACKPACK

Backpack usage
Mountaineering

Frame type
Frameless

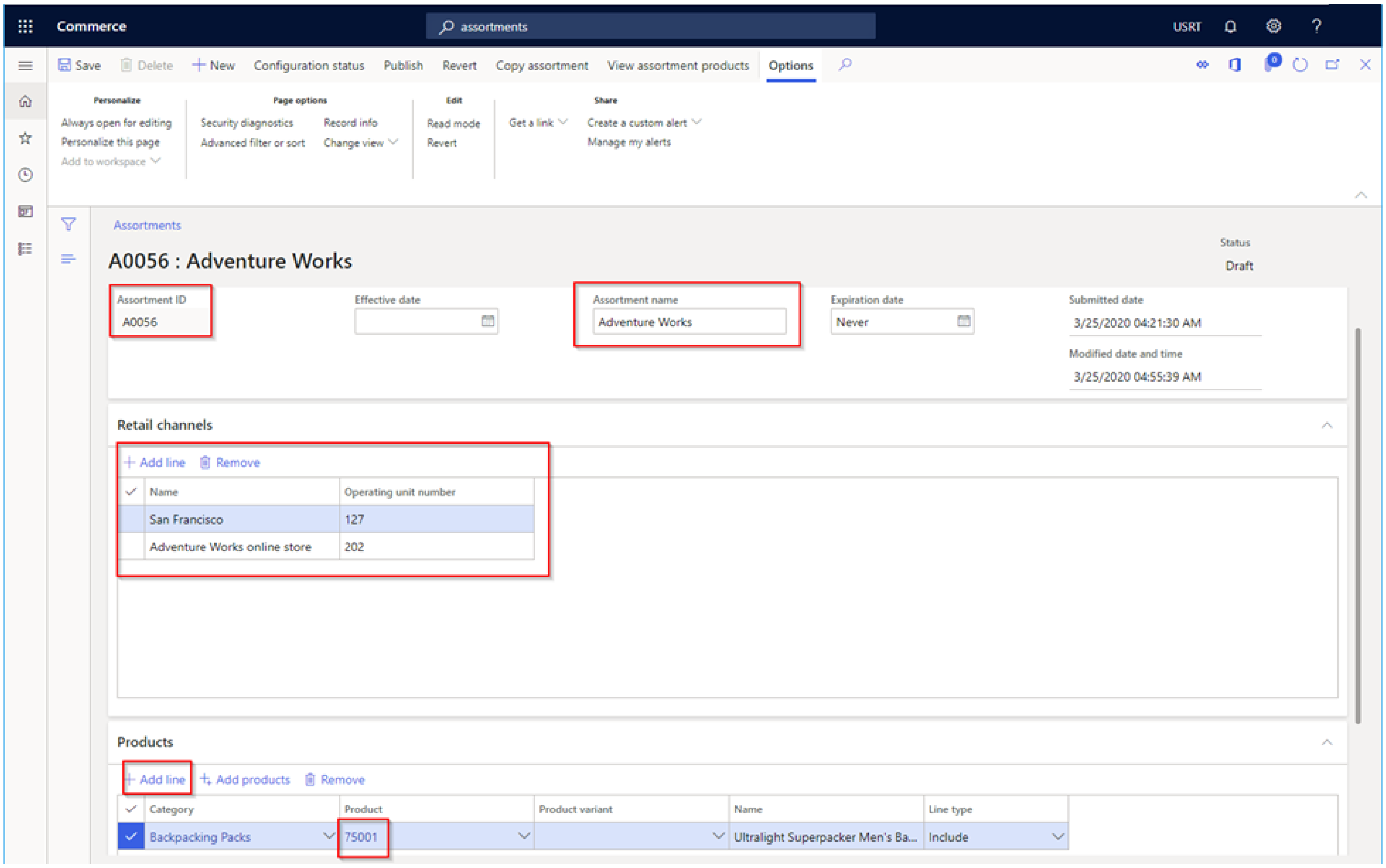
Name: Backpack usage
Data type: Text
Attribute type: Backpack usage

Value
Mountaineering

CREATE A NEW ASSORTMENT

The final step before products will be available for sale in a channel, they need to be “assorted” to the channel. Follow the steps below to create a new assortment for our new online store and add assort products to it.

- Search for "**Assortments**" and navigate to page
- Select the "**New**" button
  - Set **Assortment ID** = "**A0007**" (or accept auto number)
  - Set **Assortment Name** = "**Adventure Works**" (or any name)
- In "**Retail channels**" section add the new online channel created earlier and add the “San Francisco” for BOPIS scenario.
- Under “**Products**” section select “**Add line**”
- Add the new backpack item **75001** (or any product) (alternatively a category of items can be added here if desired)
- Click the “**Save**” button and then the “**Publish**” button



A0056 : Adventure Works

Status: Draft

Assortment ID	Effective date	Assortment name	Expiration date	Submitted date
A0056		Adventure Works	Never	3/25/2020 04:21:30 AM

Modified date and time: 3/25/2020 04:55:39 AM

Retail channels

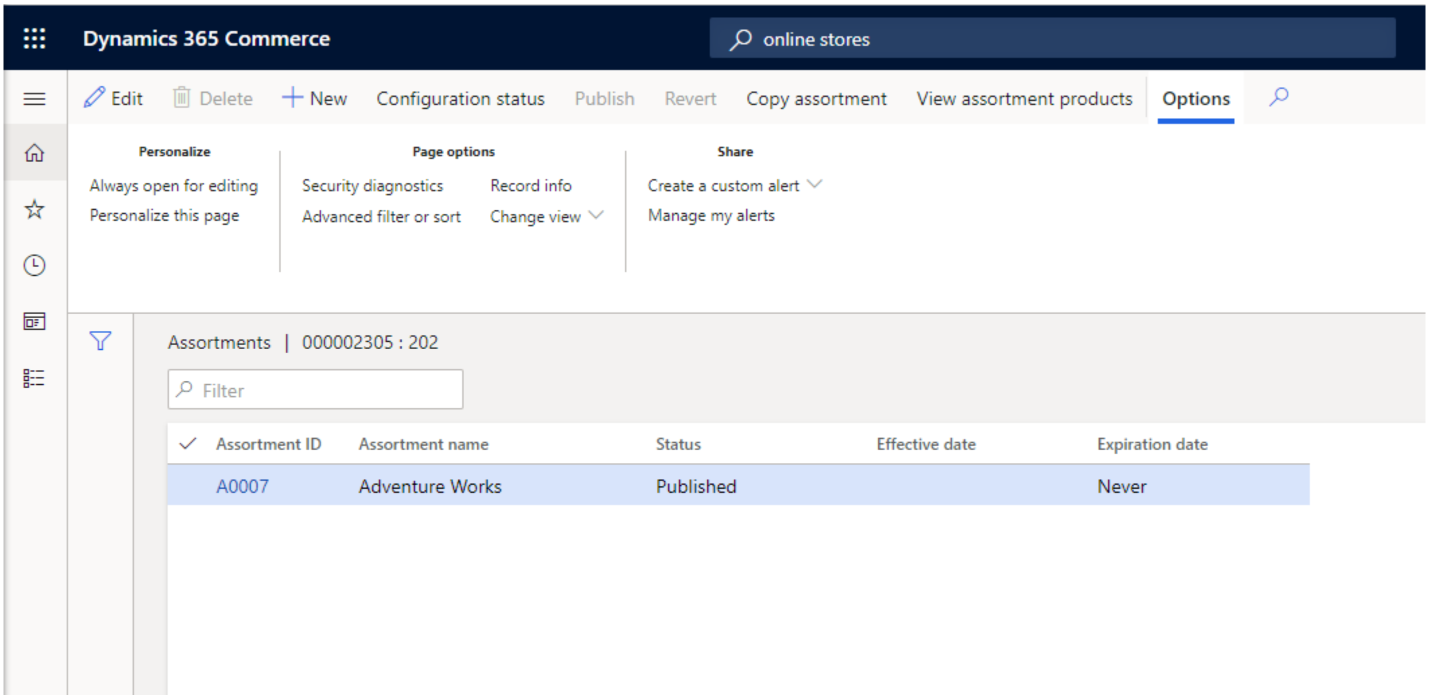
Name	Operating unit number
San Francisco	127
Adventure Works online store	202

Products

Category	Product	Product variant	Name	Line type
Backpacking Packs	75001		Ultralight Superpacker Men's Ba...	Include

Publishing will automatically process the assortments.

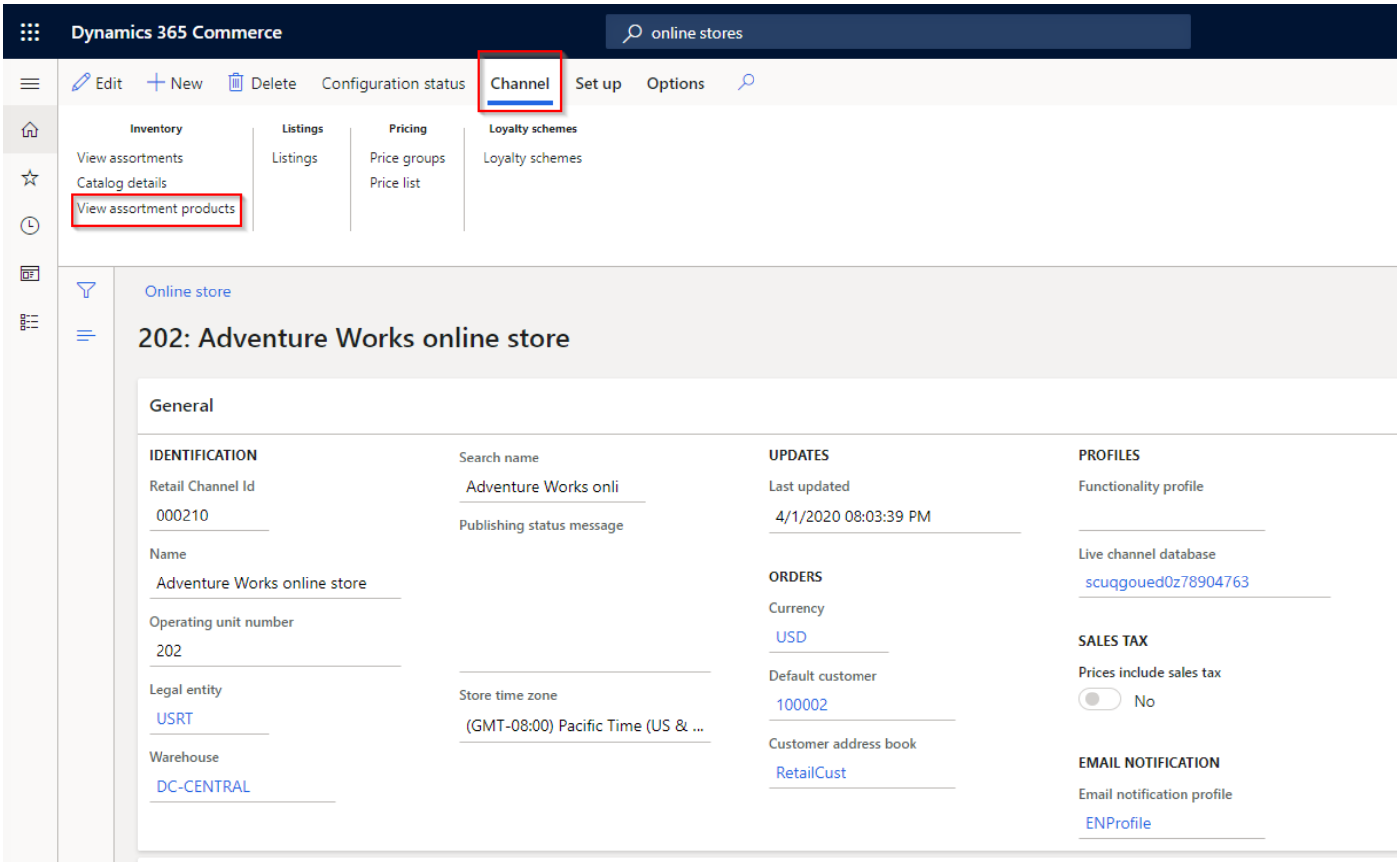
To see the assortments in our channel, select the channel from “Online stores” page and select “View assortments” and you should only see 1 assortment as shown below.



Assortments | 000002305 : 202

✓	Assortment ID	Assortment name	Status	Effective date	Expiration date
	A0007	Adventure Works	Published		Never

To see the assorted products on the channel, select “View assortment products” from the “Channel” tab on our channel.



202: Adventure Works online store

General

IDENTIFICATION

Retail Channel Id
000210

Name
Adventure Works online store

Operating unit number
202

Legal entity
USRT

Warehouse
DC-CENTRAL

Search name
Adventure Works onli

Publishing status message

Store time zone
(GMT-08:00) Pacific Time (US & ...

UPDATES

Last updated
4/1/2020 08:03:39 PM

ORDERS

Currency
USD

Default customer
100002

Customer address book
RetailCust

PROFILES

Functionality profile

Live channel database
scuqgoued0z78904763

SALES TAX

Prices include sales tax
No

EMAIL NOTIFICATION

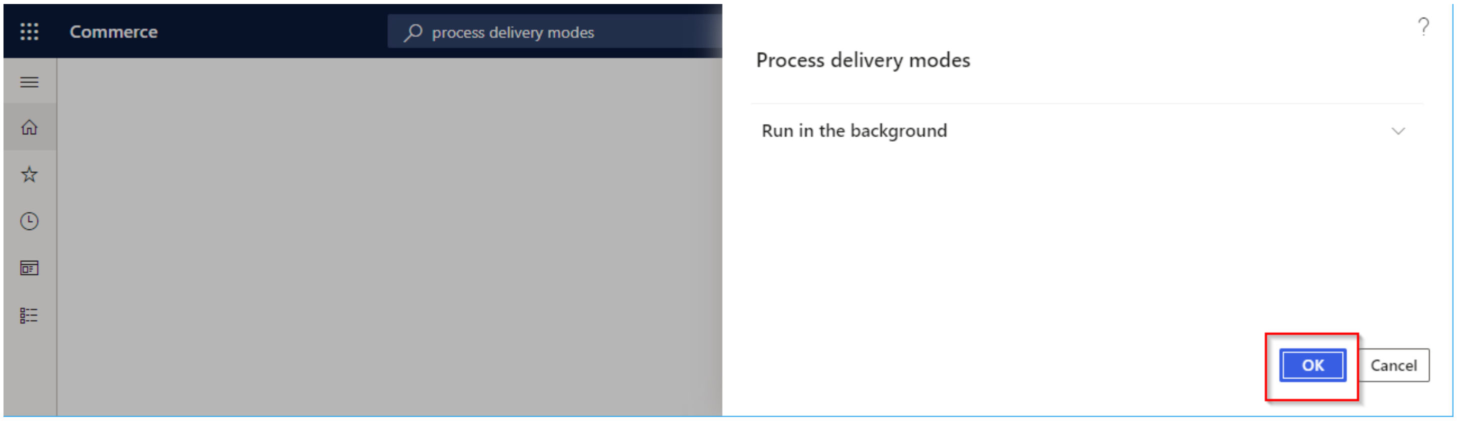
Email notification profile
ENProfile

SYNC DATA

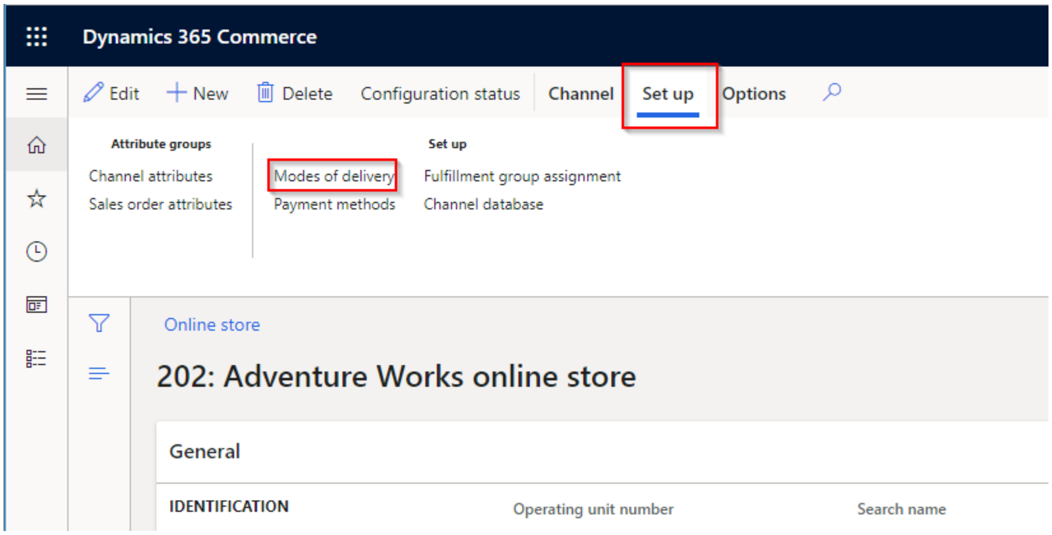
We now need to run some jobs to pick up all our changes, running the below **’9999’** job will ensure we catch everything.

Before we sync all jobs, let us process delivery modes to ensure they all get picked up:

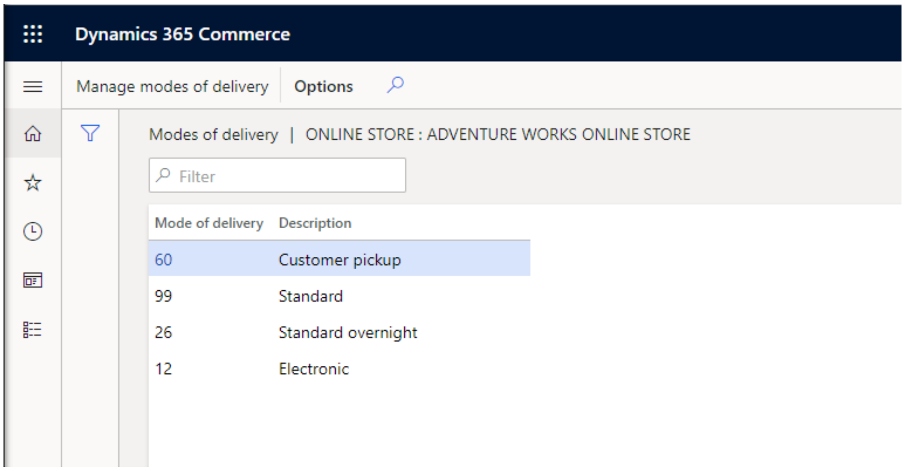
- Search for “**Process delivery modes**” and select “**OK**” to run.



To see if the new delivery modes are configured correctly, go back to “**Online stores**” and select the new store. Select the “**Set up**” tab at the top and then “**modes of delivery**”.



You should see a similar list to below image.

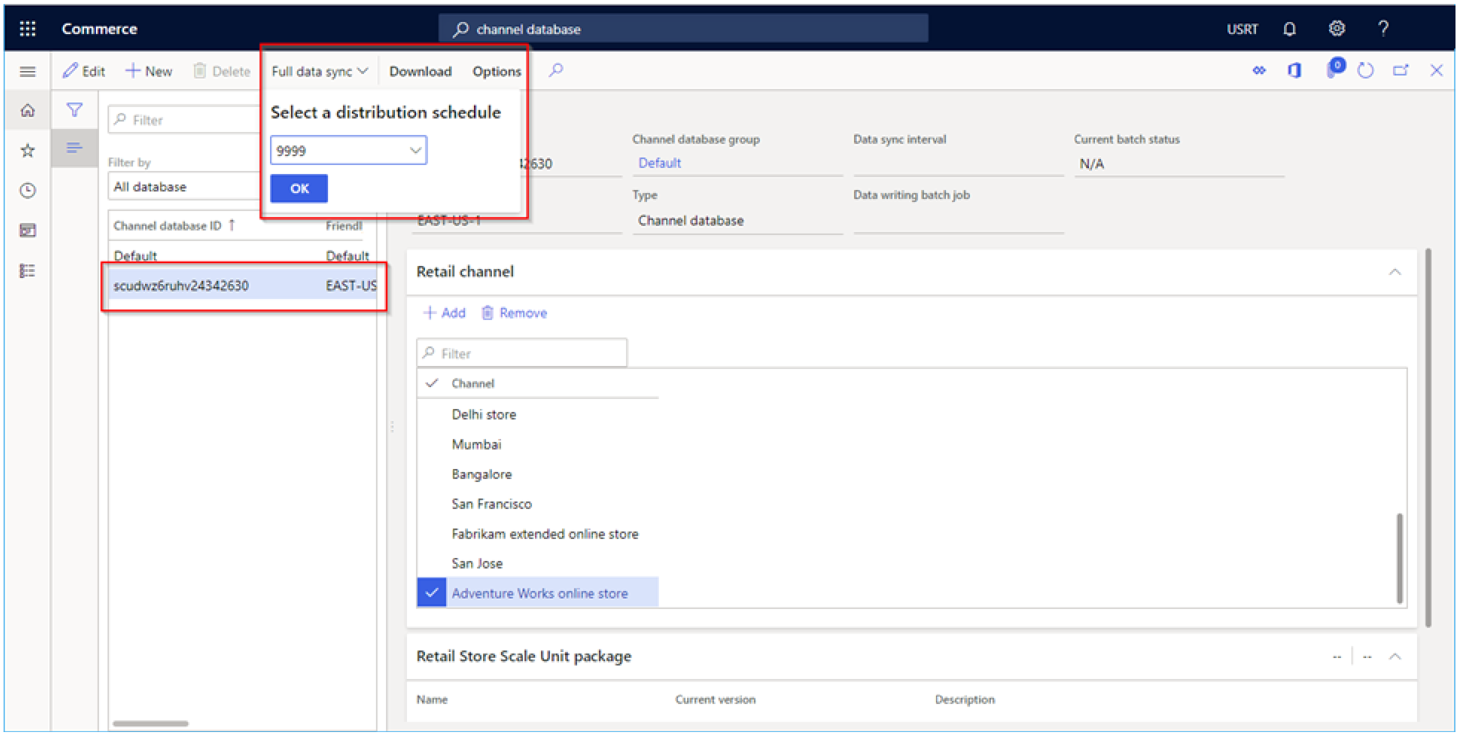


Modes of delivery | ONLINE STORE : ADVENTURE WORKS ONLINE STORE

Mode of delivery	Description
60	Customer pickup
99	Standard
26	Standard overnight
12	Electronic

To sync all jobs to ensure all the data gets sync'd to the scale unit - search for the "**Channel Database**" page.

- **Ensure** the RSCU channel database is selected and select "**Full data sync**" drop down and type in "**9999**" to run all jobs. Press "**OK**" on the next screen



Select a distribution schedule

9999

OK

Channel database ID	Friendl
Default	Default
scudwz6ruhv24342630	EAST-US

Channel database group Default	Data sync interval	Current batch status N/A
Type Channel database	Data writing batch job	

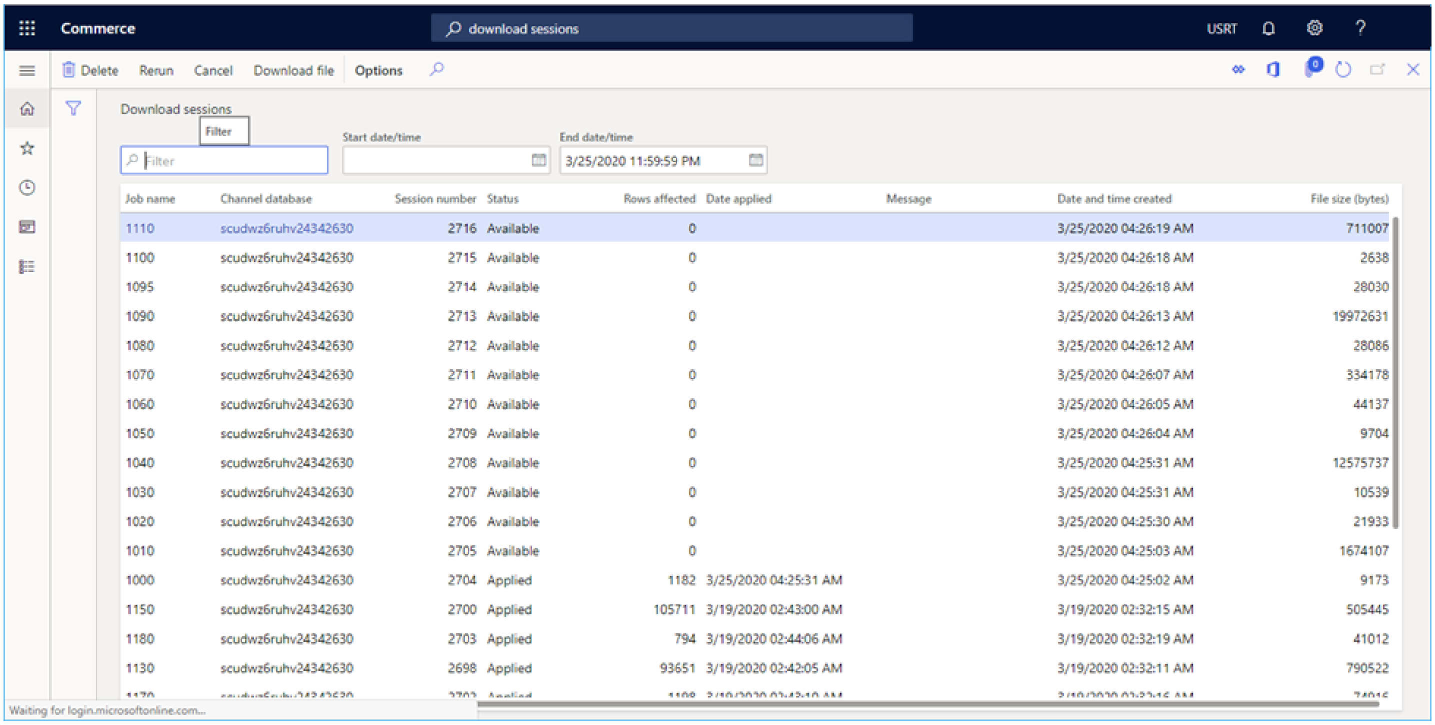
Retail channel

Channel
Delhi store
Mumbai
Bangalore
San Francisco
Fabrikam extended online store
San Jose
✓ Adventure Works online store

Retail Store Scale Unit package

Name	Current version	Description

- This could take 10-15 minutes to complete (coffee break time!). It could take up to another 15 minutes for the Azure search indexing before products will be viewable on the e-Commerce site.
- When the dialog goes away type "**download sessions**" in the search box and refresh until all Status items say "Applied"



Download sessions

Filter:

Start date/time:

End date/time: 3/25/2020 11:59:59 PM

Job name	Channel database	Session number	Status	Rows affected	Date applied	Message	Date and time created	File size (bytes)
1110	scudwz6ruhv24342630	2716	Available	0			3/25/2020 04:26:19 AM	711007
1100	scudwz6ruhv24342630	2715	Available	0			3/25/2020 04:26:18 AM	2638
1095	scudwz6ruhv24342630	2714	Available	0			3/25/2020 04:26:18 AM	28030
1090	scudwz6ruhv24342630	2713	Available	0			3/25/2020 04:26:13 AM	19972631
1080	scudwz6ruhv24342630	2712	Available	0			3/25/2020 04:26:12 AM	28086
1070	scudwz6ruhv24342630	2711	Available	0			3/25/2020 04:26:07 AM	334178
1060	scudwz6ruhv24342630	2710	Available	0			3/25/2020 04:26:05 AM	44137
1050	scudwz6ruhv24342630	2709	Available	0			3/25/2020 04:26:04 AM	9704
1040	scudwz6ruhv24342630	2708	Available	0			3/25/2020 04:25:31 AM	12575737
1030	scudwz6ruhv24342630	2707	Available	0			3/25/2020 04:25:31 AM	10539
1020	scudwz6ruhv24342630	2706	Available	0			3/25/2020 04:25:30 AM	21933
1010	scudwz6ruhv24342630	2705	Available	0			3/25/2020 04:25:03 AM	1674107
1000	scudwz6ruhv24342630	2704	Applied	1182	3/25/2020 04:25:31 AM		3/25/2020 04:25:02 AM	9173
1150	scudwz6ruhv24342630	2700	Applied	105711	3/19/2020 02:43:00 AM		3/19/2020 02:32:15 AM	505445
1180	scudwz6ruhv24342630	2703	Applied	794	3/19/2020 02:44:06 AM		3/19/2020 02:32:19 AM	41012
1130	scudwz6ruhv24342630	2698	Applied	93651	3/19/2020 02:42:05 AM		3/19/2020 02:32:11 AM	790522
1170	scudwz6ruhv24342630	2702	Applied	1198	3/19/2020 02:43:10 AM		3/19/2020 02:32:16 AM	74016

This completes the Headquarters setup. We now have a new channel and a new channel hierarchy with products in it.

UNIT 2 CREATE AND MANAGE AN E-COMMERCE SITE

CREATE AN ONLINE SITE

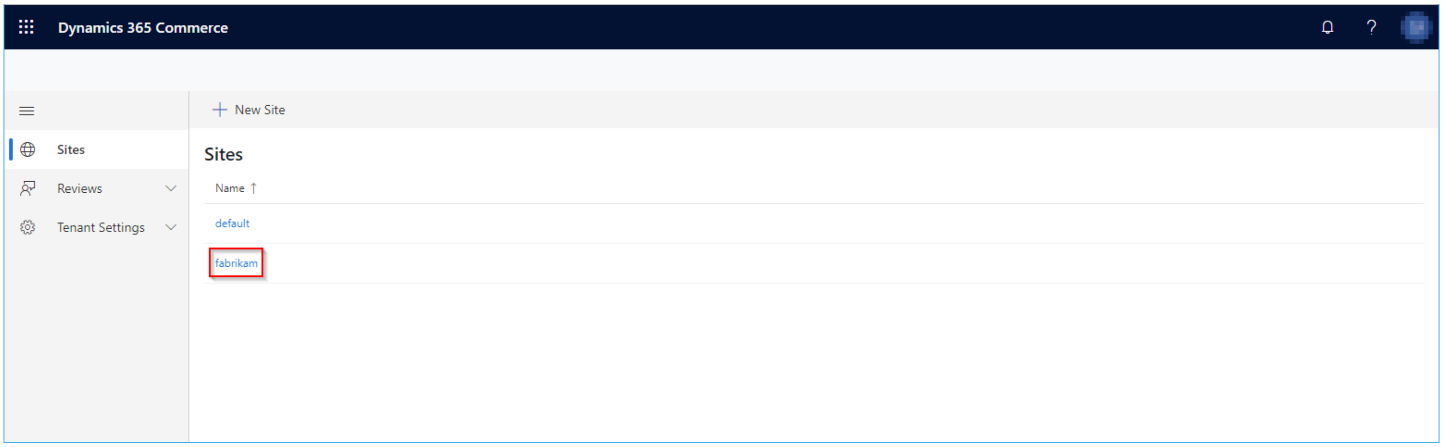
In this section we will create a new e-Commerce site with the Site Builder tool (see top of document on how to load this tool from within LCS). When creating the site, we will be choosing an online store channel, the language, and the Active Directory security group to use for site administrators.

There are two options when creating a new site, the most ideal option is the first run experience which will create a default set of pages for your site with Fabrikam branding and sample marketing images. The second option is to create a new site, which will not have any pages created for your site. Follow the below steps for the first run experience if you are using a newly deployed e-Commerce site or see Appendix A/B to create a new site and the corresponding fragments/templates and pages.

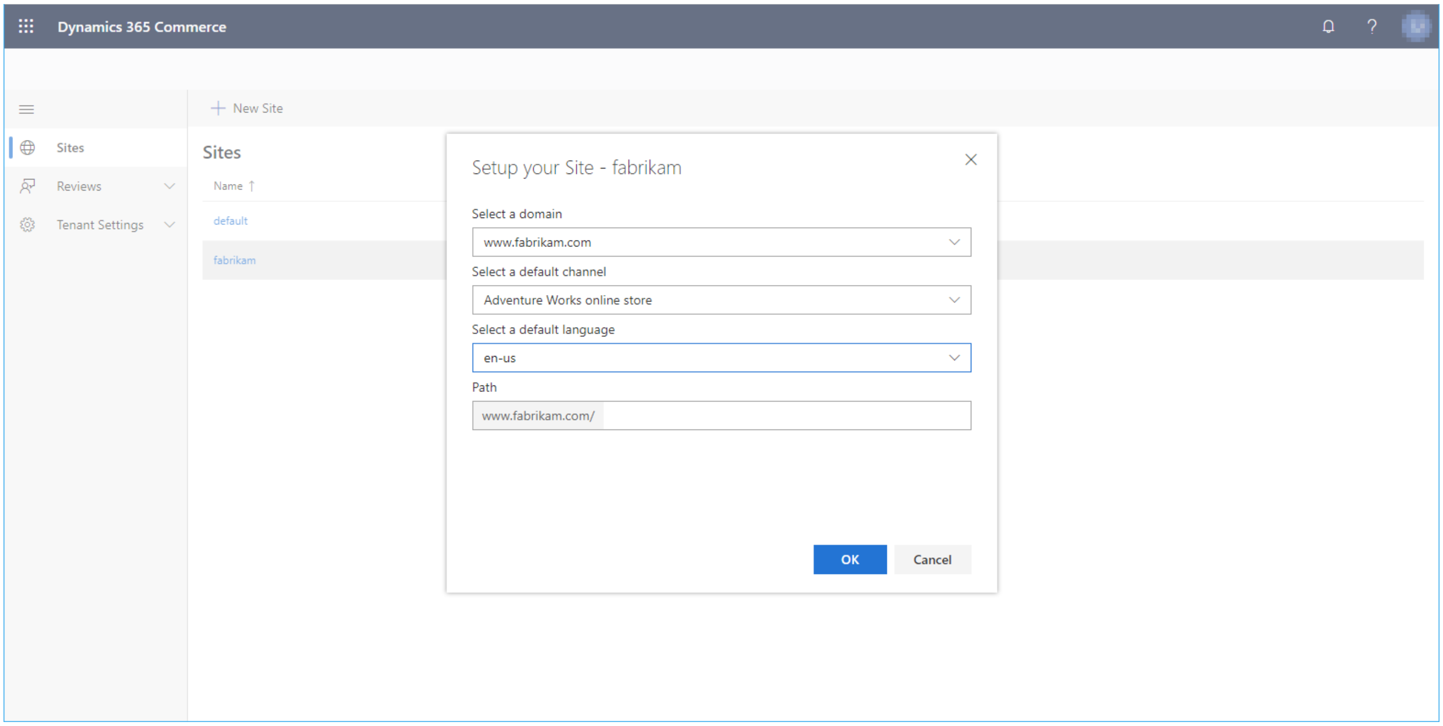
SET UP YOUR E-COMMERCE SITE

Follow the next set of steps to set up your e-Commerce site.

Launch the site builder tool and you should see a "Sites" selection page. Select the **Fabrikam** site to open the site setup dialog box (see note below if you do not see the dialog box).



NOTE: If you are using an environment that already has the Fabrikam site setup against the "Fabrikam extended online store" then when you select the Fabrikam site, instead of getting the below "Setup your Site" dialog box it will take you to site to a list of site pages. This indicates that the first run experience has already happened, in this case you may need to either create a new site (see appendix A/B) or change the Fabrikam site to use your new channel (see next section to make this change). If you are in this situation, **we recommend you change the channel on the site for this workshop** instead of creating a new site since you'll have to create the pages/templates and fragments manually if you create a new site.



Setup your Site - fabrikam

Select a domain: www.fabrikam.com

Select a default channel: Adventure Works online store

Select a default language: en-us

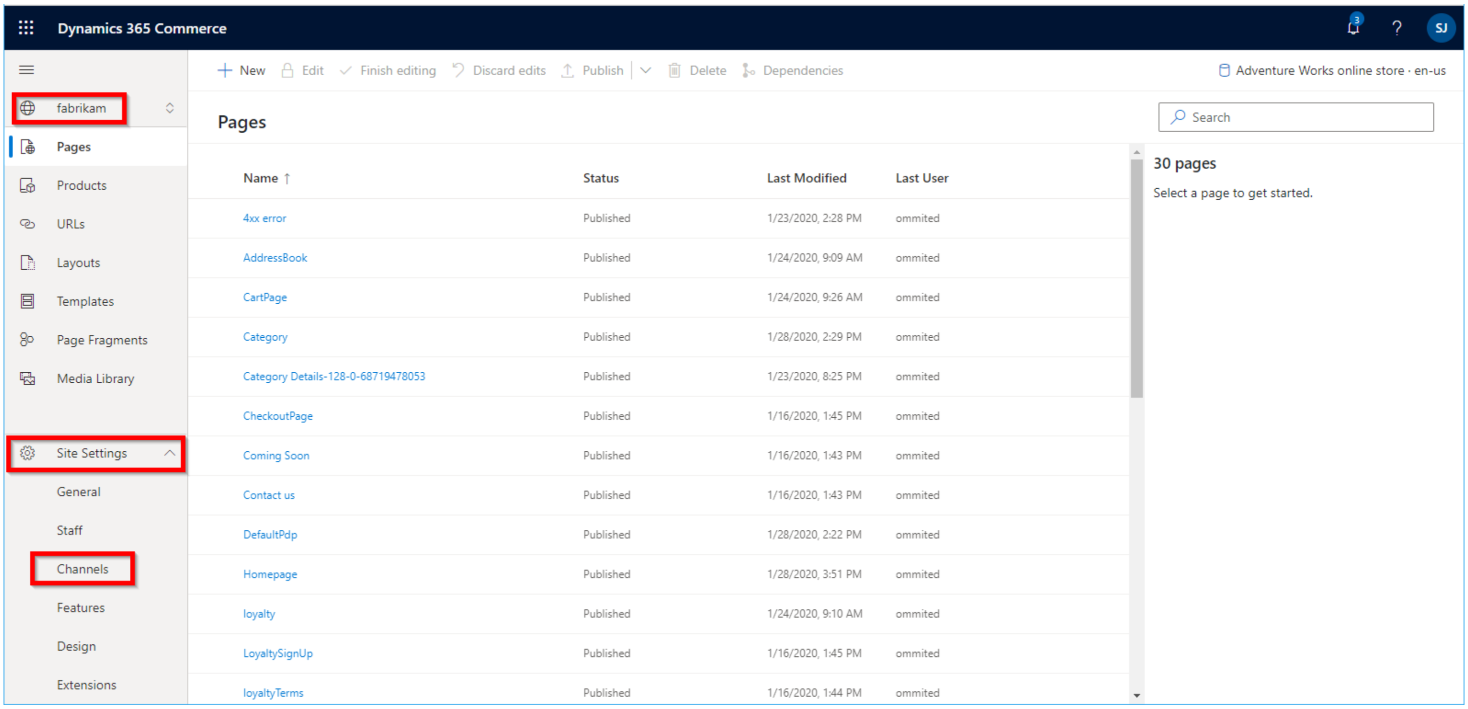
Path: www.fabrikam.com/

- Select the domain from the drop down, there should only be one option that was entered in when the e-Commerce site was deployed through LCS.
- Select the new online channel created earlier in this guide, which should appear in the drop down.
- Select **en-us** as the default language. NOTE: this is important to use en-us for this step since the default pages are only provided in English, additional languages can be added later.
- Leave the value of the **Path** field as it is.
- Select **OK**. The list of pages on the site appears.

CHANGING SITE TO A DIFFERENT ONLINE CHANNEL

If the above new "Setup your site" dialog did not come it, it means the site has already been initialized, we just need to change it to point to the new channel created earlier. Follow the below steps.

You should now be in the selected "Fabrikam" site inside of the site builder tool and be on the "Pages" tab as shown below.



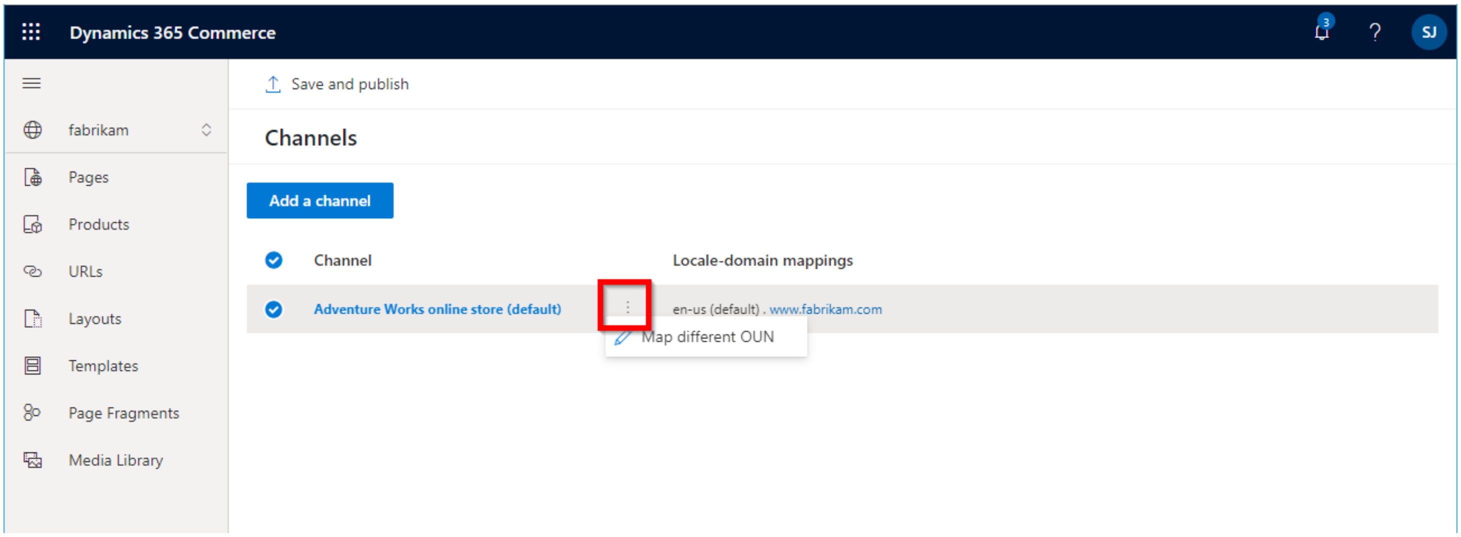
Pages

Name ↑	Status	Last Modified	Last User
4xx error	Published	1/23/2020, 2:28 PM	ommited
AddressBook	Published	1/24/2020, 9:09 AM	ommited
CartPage	Published	1/24/2020, 9:26 AM	ommited
Category	Published	1/28/2020, 2:29 PM	ommited
Category Details-128-0-68719478053	Published	1/23/2020, 8:25 PM	ommited
CheckoutPage	Published	1/16/2020, 1:45 PM	ommited
Coming Soon	Published	1/16/2020, 1:43 PM	ommited
Contact us	Published	1/16/2020, 1:43 PM	ommited
DefaultPdp	Published	1/28/2020, 2:22 PM	ommited
Homepage	Published	1/28/2020, 3:51 PM	ommited
loyalty	Published	1/24/2020, 9:10 AM	ommited
LoyaltySignUp	Published	1/16/2020, 1:45 PM	ommited
loyaltyTerms	Published	1/16/2020, 1:44 PM	ommited

30 pages

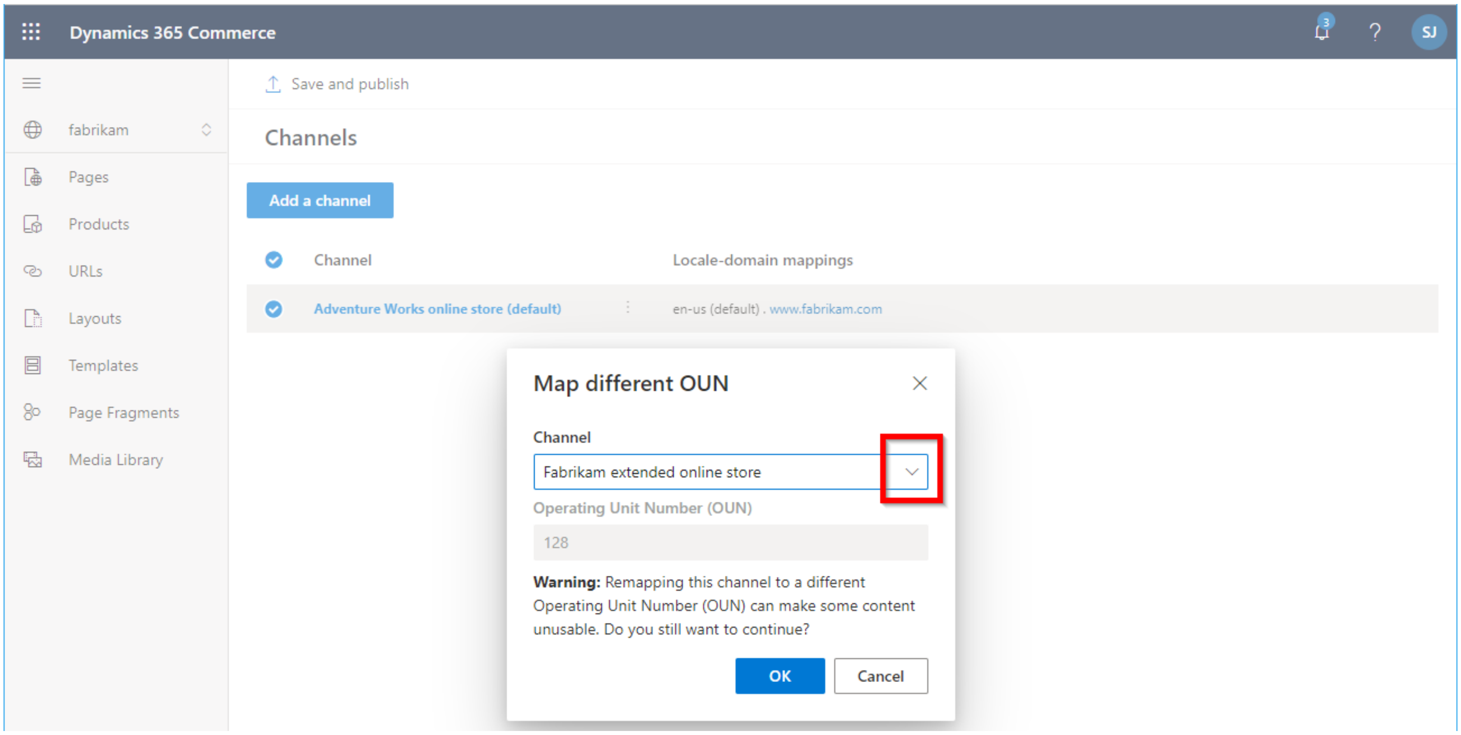
Select a page to get started.

At the bottom left, expand the "Site Settings" section and select the "Channels" tab. You should now see a list of channels for this site, this is the same area you can configure multiple locales for a site. Select the vertical "…" and then "Map different OUN" in the pop-up menu.



Channel	Locale-domain mappings
Adventure Works online store (default)	en-us (default) . www.fabrikam.com

You can now select the appropriate channel from the Channel drop down as shown below followed by pressing the OK button. You will then need to the select the “Save and publish” button at the top for the setting to take effect.



Channel	Locale-domain mappings
Adventure Works online store (default)	en-us (default) . www.fabrikam.com

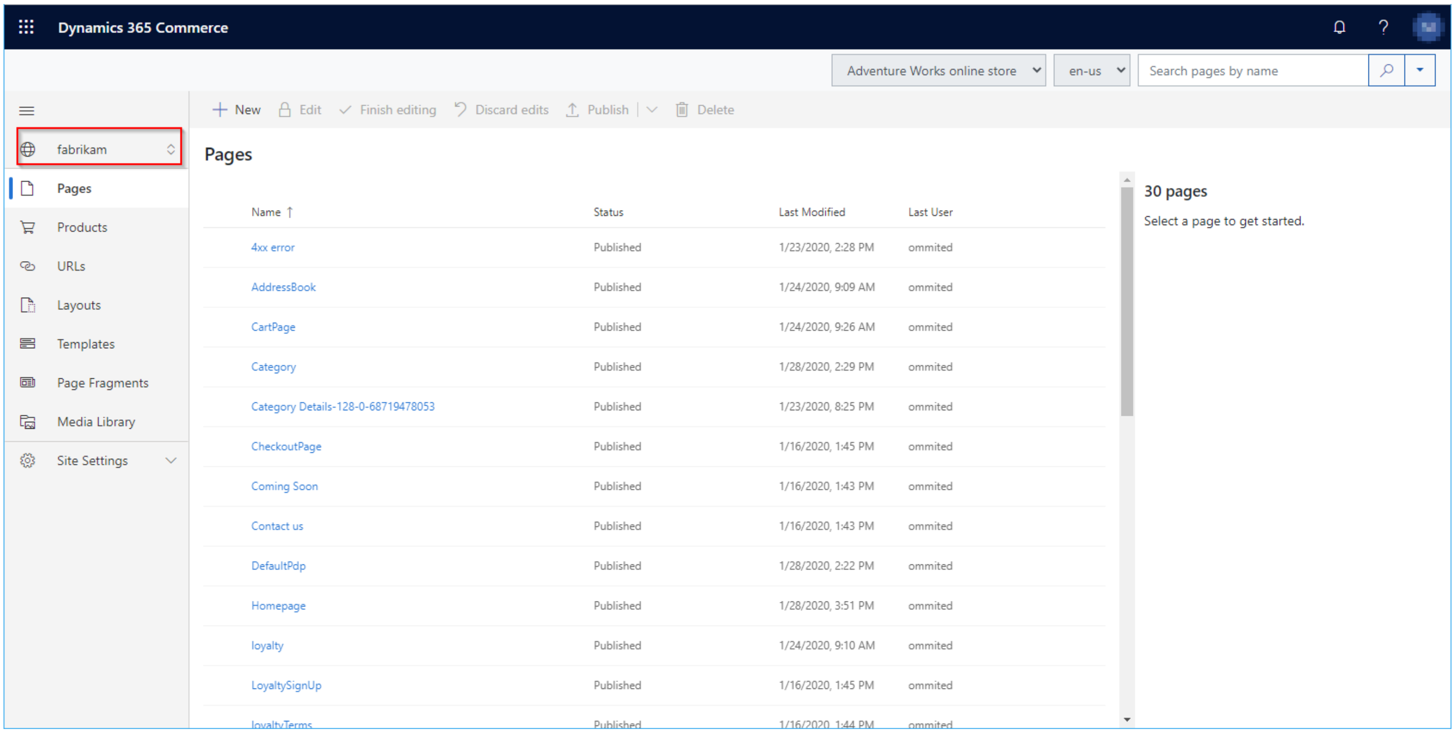
Map different OUN

Channel: Fabrikam extended online store

Operating Unit Number (OUN): 128

Warning: Remapping this channel to a different Operating Unit Number (OUN) can make some content unusable. Do you still want to continue?

At this point you should now have a full featured e-Commerce site, the new site called “Fabrikam” will be auto selected and you should now see a list of e-Commerce pages.



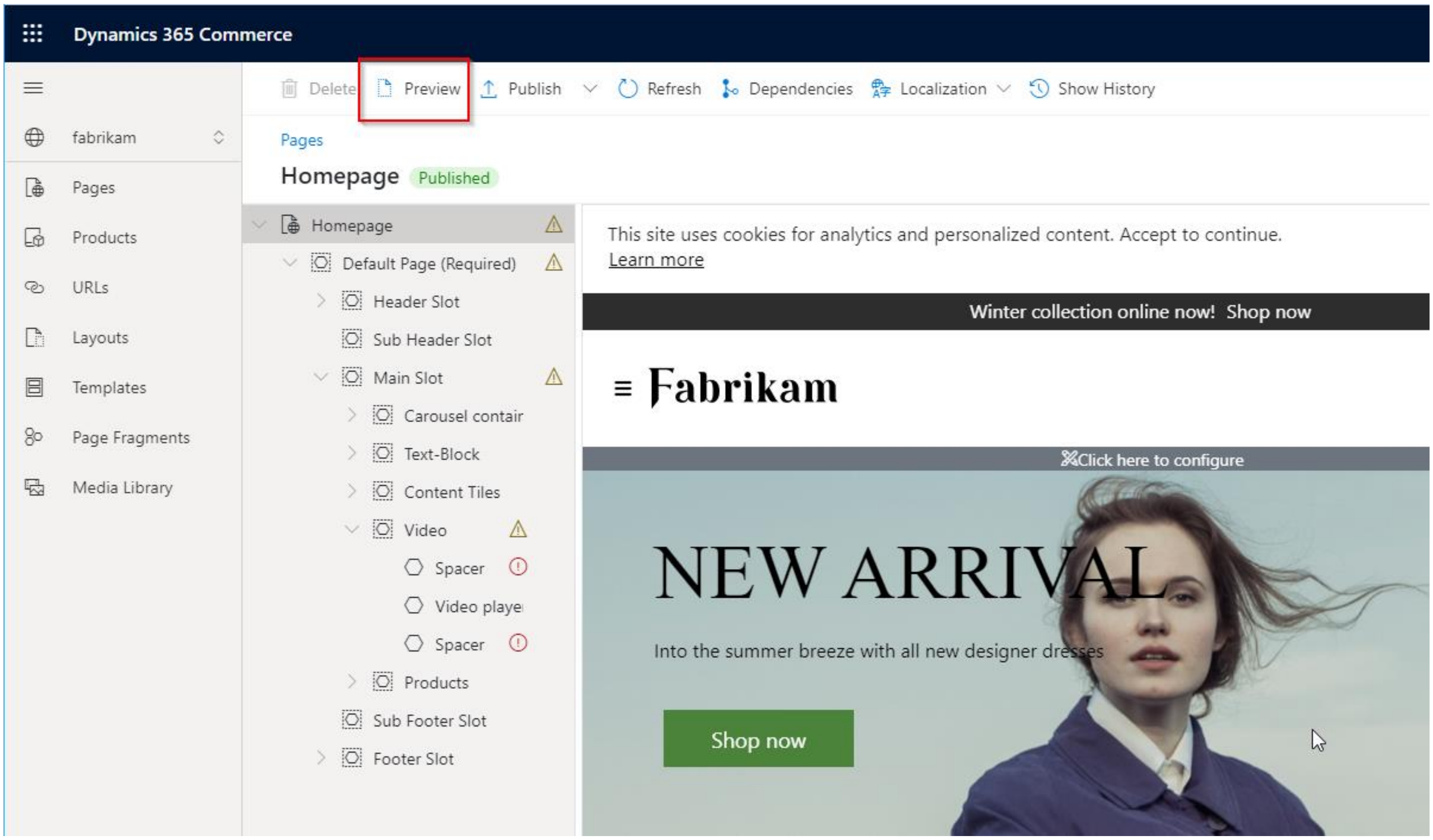
Pages

Name ↑	Status	Last Modified	Last User
4xx error	Published	1/23/2020, 2:28 PM	ommited
AddressBook	Published	1/24/2020, 9:09 AM	ommited
CartPage	Published	1/24/2020, 9:26 AM	ommited
Category	Published	1/28/2020, 2:29 PM	ommited
Category Details-128-0-68719478053	Published	1/23/2020, 8:25 PM	ommited
CheckoutPage	Published	1/16/2020, 1:45 PM	ommited
Coming Soon	Published	1/16/2020, 1:43 PM	ommited
Contact us	Published	1/16/2020, 1:43 PM	ommited
DefaultPdp	Published	1/28/2020, 2:22 PM	ommited
Homepage	Published	1/28/2020, 3:51 PM	ommited
loyalty	Published	1/24/2020, 9:10 AM	ommited
LoyaltySignUp	Published	1/16/2020, 1:45 PM	ommited
loyaltyTerms	Published	1/16/2020, 1:44 PM	ommited

30 pages

Select a page to get started.

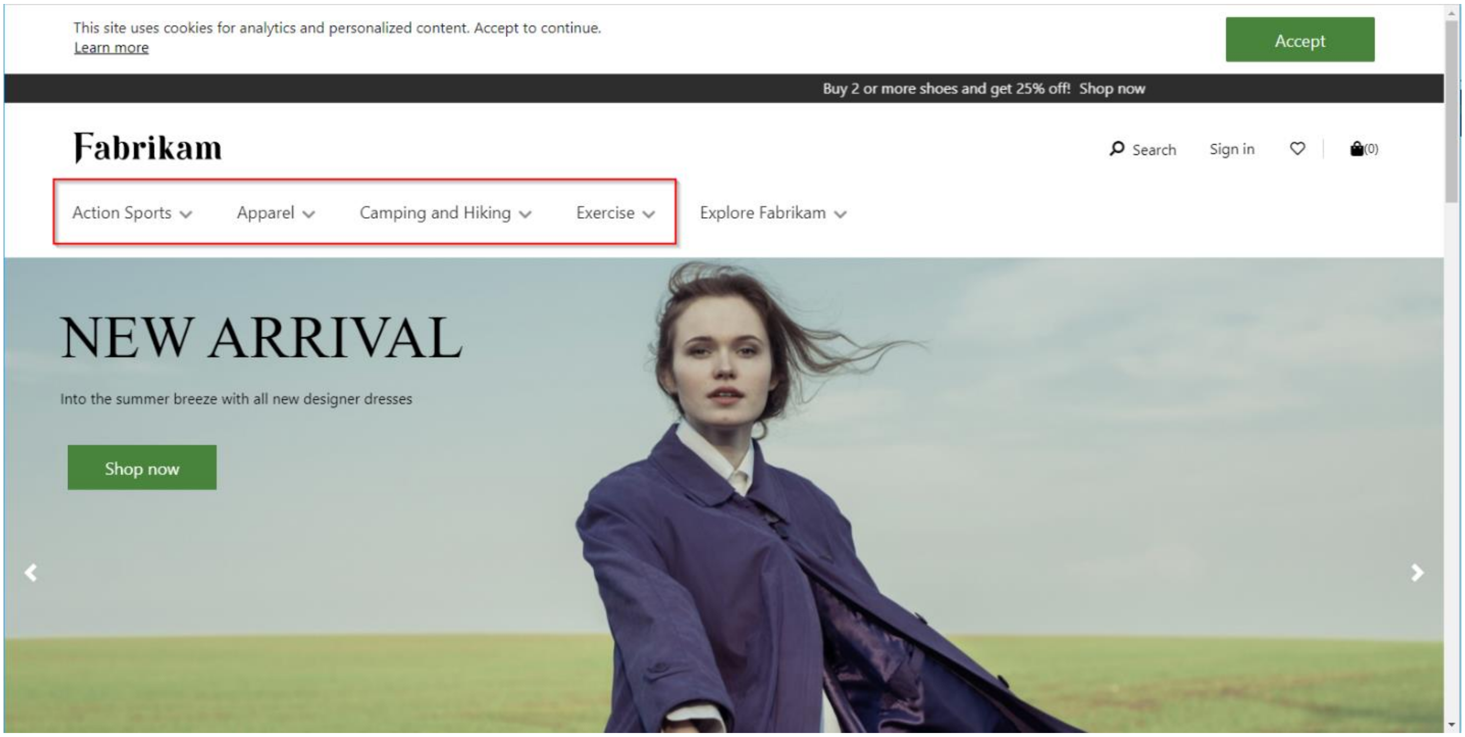
To test out the web site, select the “**Homepage**” from above list and select the Preview button as shown below.



The web site should now be a "Fabrikam" branded site with the channel navigation you created earlier in this document and clicking through to categories you should be able to see the products you created.

Take note of the URL it should look similar to : https://commerceinaday-9-eval.commerce.dynamics.com/?domain=www.fabrikam.com&preview=inprogress

If you remove the "**?domain=www.fabrikam.com&preview=inprogress"** from the url you can bookmark this for the site home page (otherwise you can get to the homepage from LCS).



This site uses cookies for analytics and personalized content. Accept to continue.

Buy 2 or more shoes and get 25% off! Shop now

NEW ARRIVAL

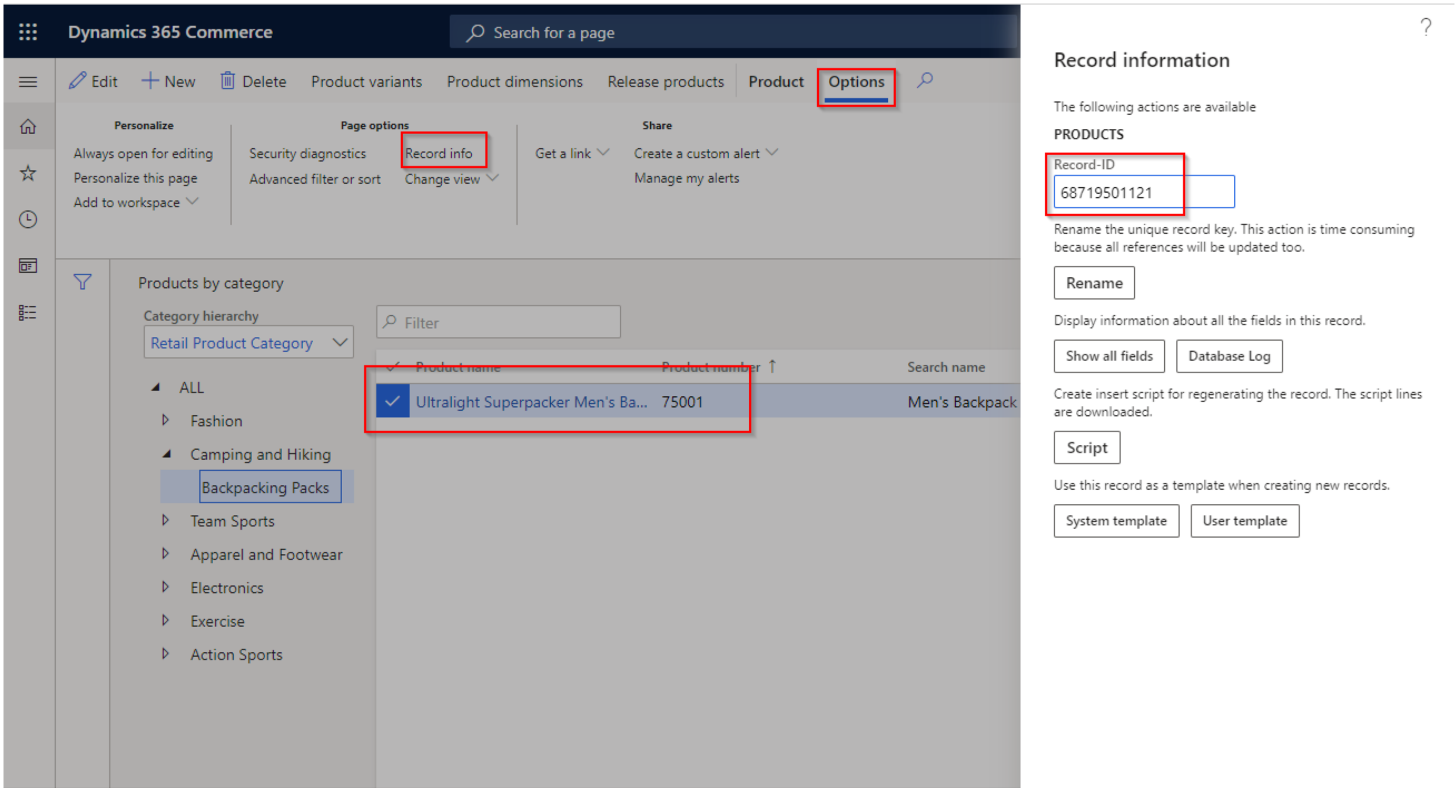
Into the summer breeze with all new designer dresses

Note: As you navigate into product categories, if no products show up, it may just take some time for the search indexing to finish up, try again in 10-20 minutes. If products still do not show up after some time seek some support.

Also note that new product images will not show up until you upload them to the "Media Library" which we'll do in the next section.

If your product is still not showing up in the category (could be due to the search indexing taking longer than expected), another way to test if your product is working is to directly link to the product details page.

From within Headquarters search for "**products by category**" and select the product created earlier. Select "Options" and "Record Info and copy the "Record-ID" for the product.



The following actions are available

Record-ID: 68719501121

Rename the unique record key. This action is time consuming because all references will be updated too.

Display information about all the fields in this record.

Create insert script for regenerating the record. The script lines are downloaded.

Use this record as a template when creating new records.

You can then construct a URL based on this record id. Take your site URL and add the Record-ID.p (the ".p" represents product). For example if the site URL is: https://commerceinaday-9-eval.commerce.dynamics.com/

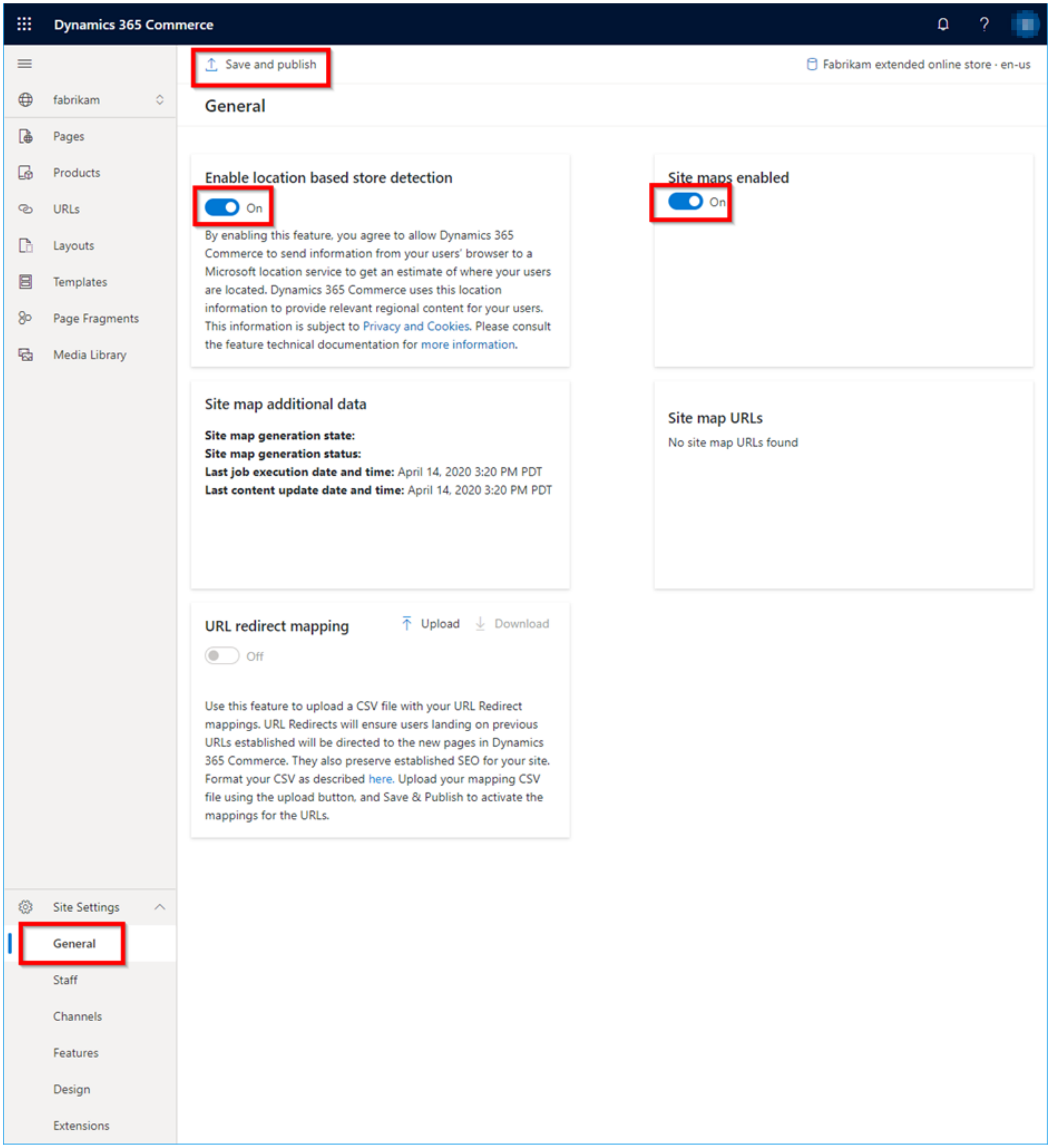
Then change it to : https://commerceinaday-9-eval.commerce.dynamics.com/68719501121.p

E-COMMERCE SITE MANAGEMENT

We will now discuss some options that are available in configuring an e-Commerce site. Most of this will be left at the default values for this workshop.

Open the “**Site Settings**” on the bottom left and select the “**General**” tab. From here you can modify features as you wish, below shows the location-based store detection and site maps being enabled. Enabling site maps will enable the automated sitemap generation service, which will automatically keep track of page, category, product and URL additions/deletions/updates and maintain an up-to-date sitemap document for search engine indexing.

Select “**Save and Publish**” to make the settings live.



General

Enable location based store detection

On

By enabling this feature, you agree to allow Dynamics 365 Commerce to send information from your users' browser to a Microsoft location service to get an estimate of where your users are located. Dynamics 365 Commerce uses this location information to provide relevant regional content for your users. This information is subject to Privacy and Cookies. Please consult the feature technical documentation for more information.

Site maps enabled

On

Site map additional data

Site map generation state:
Site map generation status:
Last job execution date and time: April 14, 2020 3:20 PM PDT
Last content update date and time: April 14, 2020 3:20 PM PDT

Site map URLs

No site map URLs found

URL redirect mapping

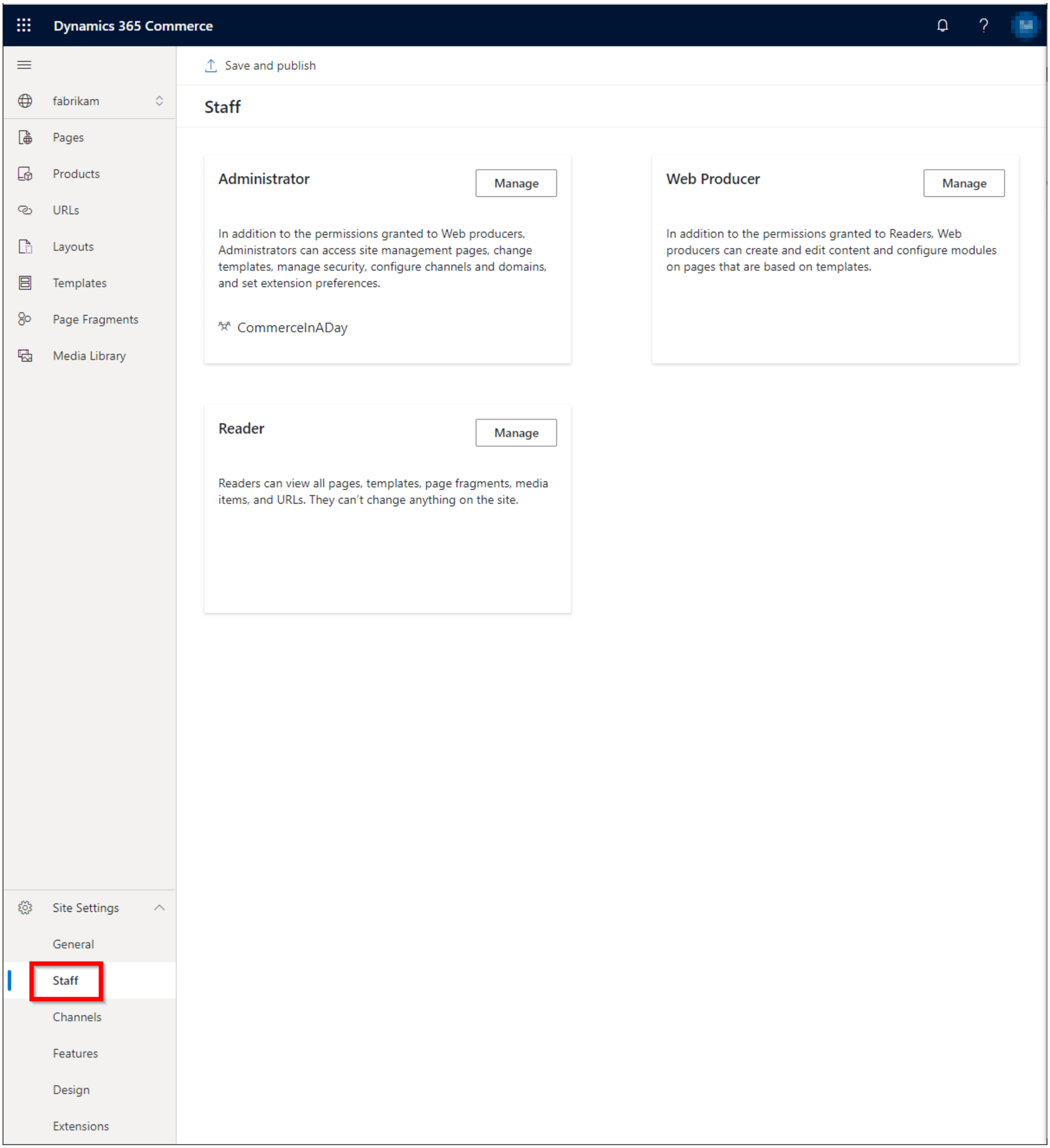
Off

Use this feature to upload a CSV file with your URL Redirect mappings. URL Redirects will ensure users landing on previous URLs established will be directed to the new pages in Dynamics 365 Commerce. They also preserve established SEO for your site. Format your CSV as described here. Upload your mapping CSV file using the upload button, and Save & Publish to activate the mappings for the URLs.

STAFF CONFIGURATION

From within the security configuration tab you can configure various AAD groups to various roles including “**Administrator**”, “**Web Producer**” and “**Reader**”. By default, you will see the AD group added to the Administrator section used when the online site was created.

Selecting manage will allow you to add or remove AAD Groups.



Staff

Administrator

In addition to the permissions granted to Web producers, Administrators can access site management pages, change templates, manage security, configure channels and domains, and set extension preferences.

CommerceInADay

Web Producer

In addition to the permissions granted to Readers, Web producers can create and edit content and configure modules on pages that are based on templates.

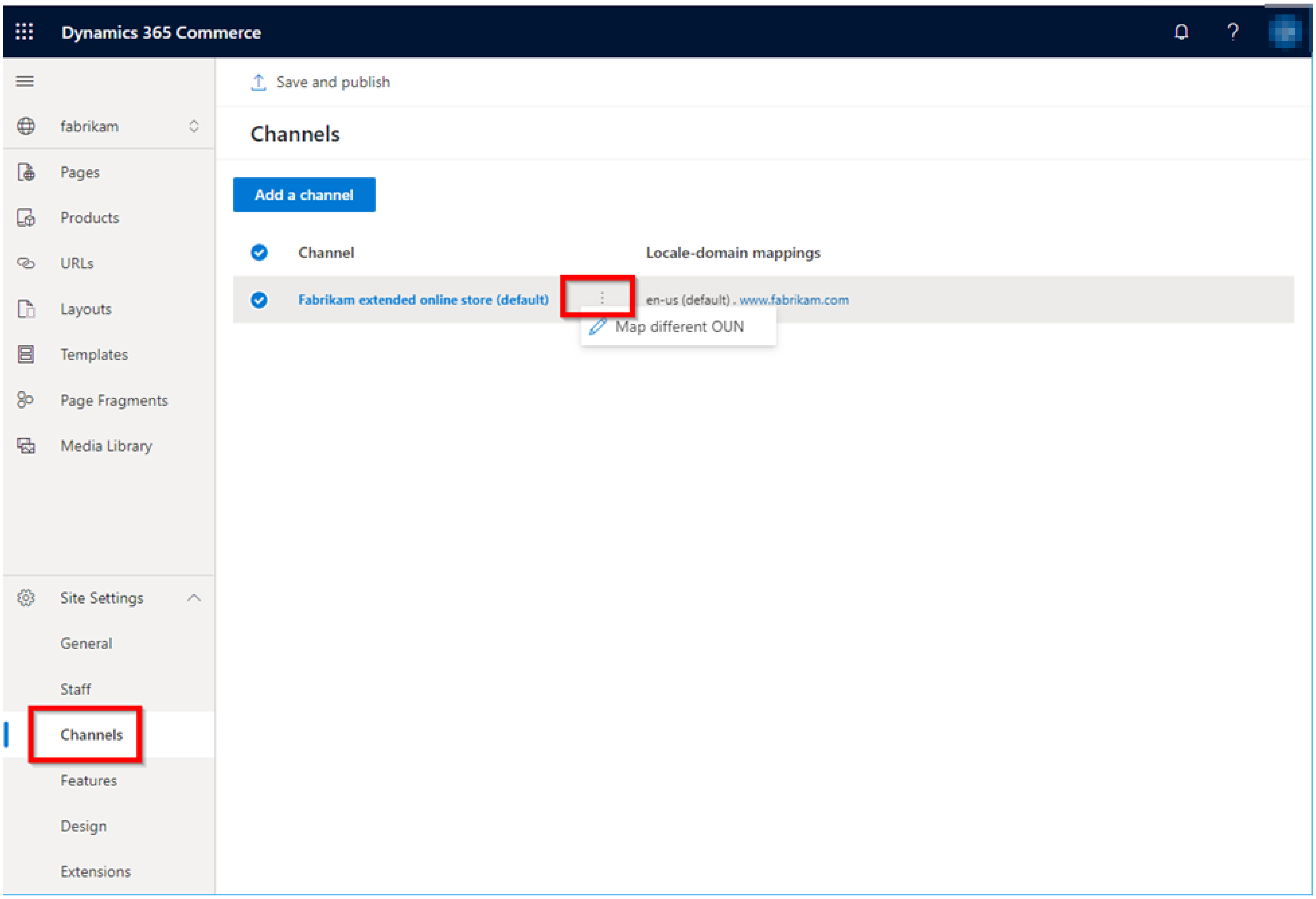
Reader

Readers can view all pages, templates, page fragments, media items, and URLs. They can't change anything on the site.

CHANNEL CONFIGURATION

From within the “Channels” setting tab, you can map additional channels into the single site. This is how you can add additional localized channels into a single site. For example, you can add an additional “**Adventure Works online store**” for another locale if configured.

You can also select the three ellipses to change the online channel used for the site.



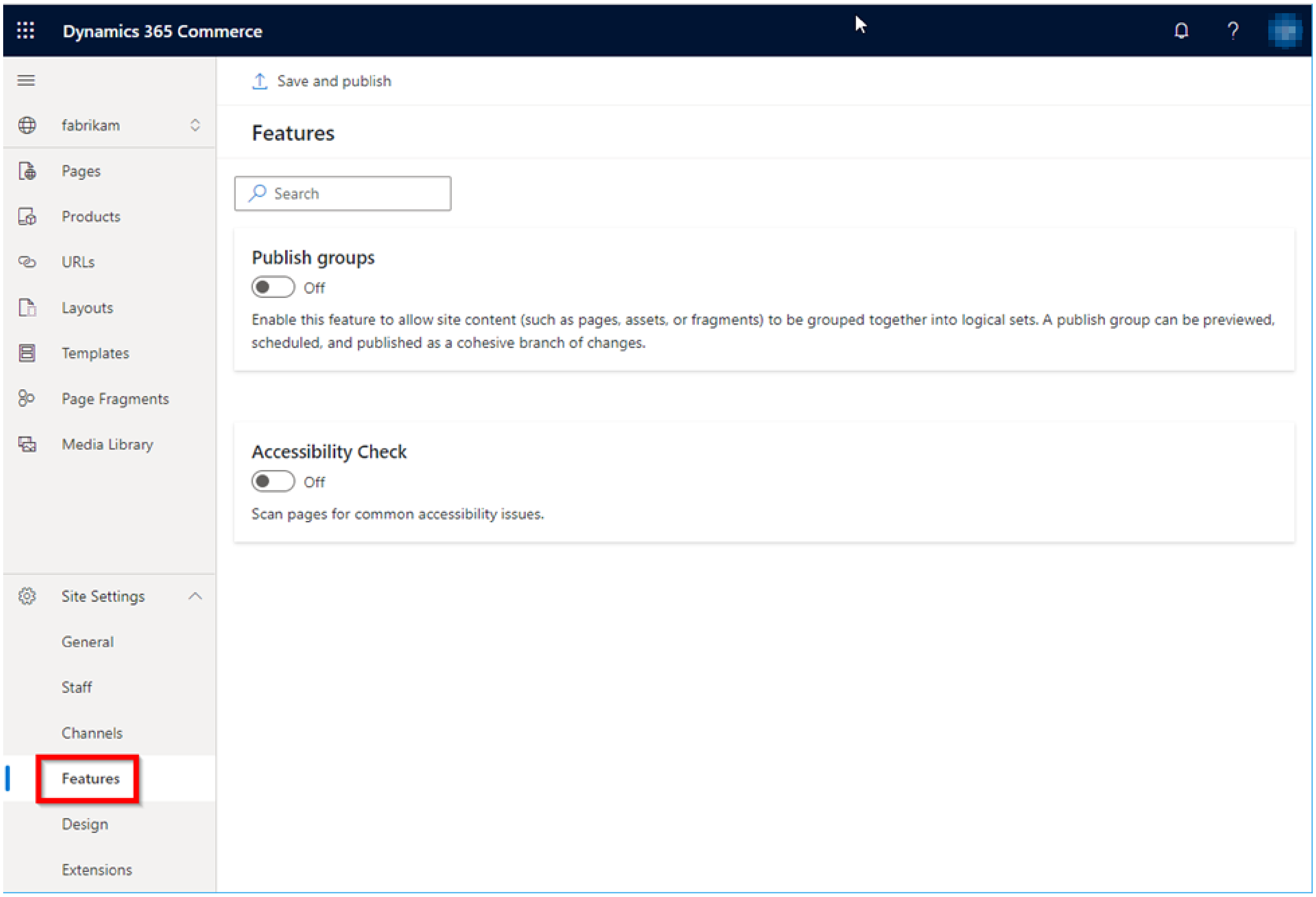
Channels

Channel	Locale-domain mappings
Fabrikam extended online store (default)	en-us (default) . www.fabrikam.com

FEATURES SETTINGS

The **Features** Settings section provides some additional feature options that can be enabled included “Publish groups” and “Accessibility Check”.

Publish groups allow a site author to modify several pages, templates and fragments together as a group and when complete publish them live together.



Features

Publish groups

Off

Enable this feature to allow site content (such as pages, assets, or fragments) to be grouped together into logical sets. A publish group can be previewed, scheduled, and published as a cohesive branch of changes.

Accessibility Check

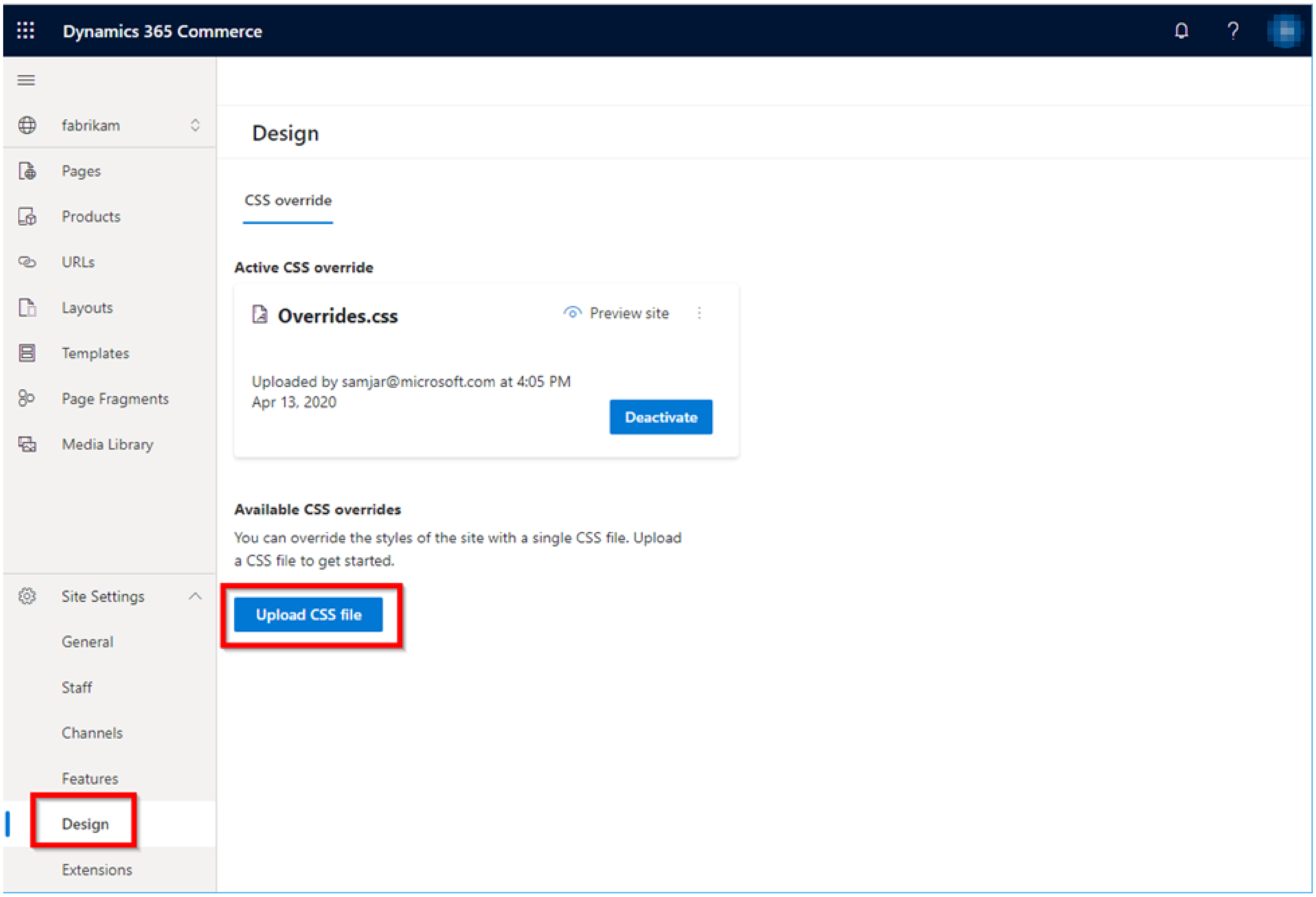
Off

Scan pages for common accessibility issues.

DESIGN CONFIGURATION

The “Design” settings section is used to upload CSS theme overrides to a site. Only 1 theme override file can be uploaded at a time. You can also preview the site with the changes before the override file is activated. Once activated, it can also be de-activated at any time.

Theme overrides allow for quick site CSS overrides that will be applied to the entire site. Generally, CSS changes should be done within the theme code, however this requires the use of the online extensibility SDK (see unit 3) to create and modify a site theme then uploading and deploying the extensibility package. Overrides provide a way to do simple CSS overrides, great for testing changes on the site or building demos. More details are provided later in Unit 2.



Design

CSS override

Active CSS override

Overrides.css

Uploaded by samjar@microsoft.com at 4:05 PM Apr 13, 2020

Available CSS overrides

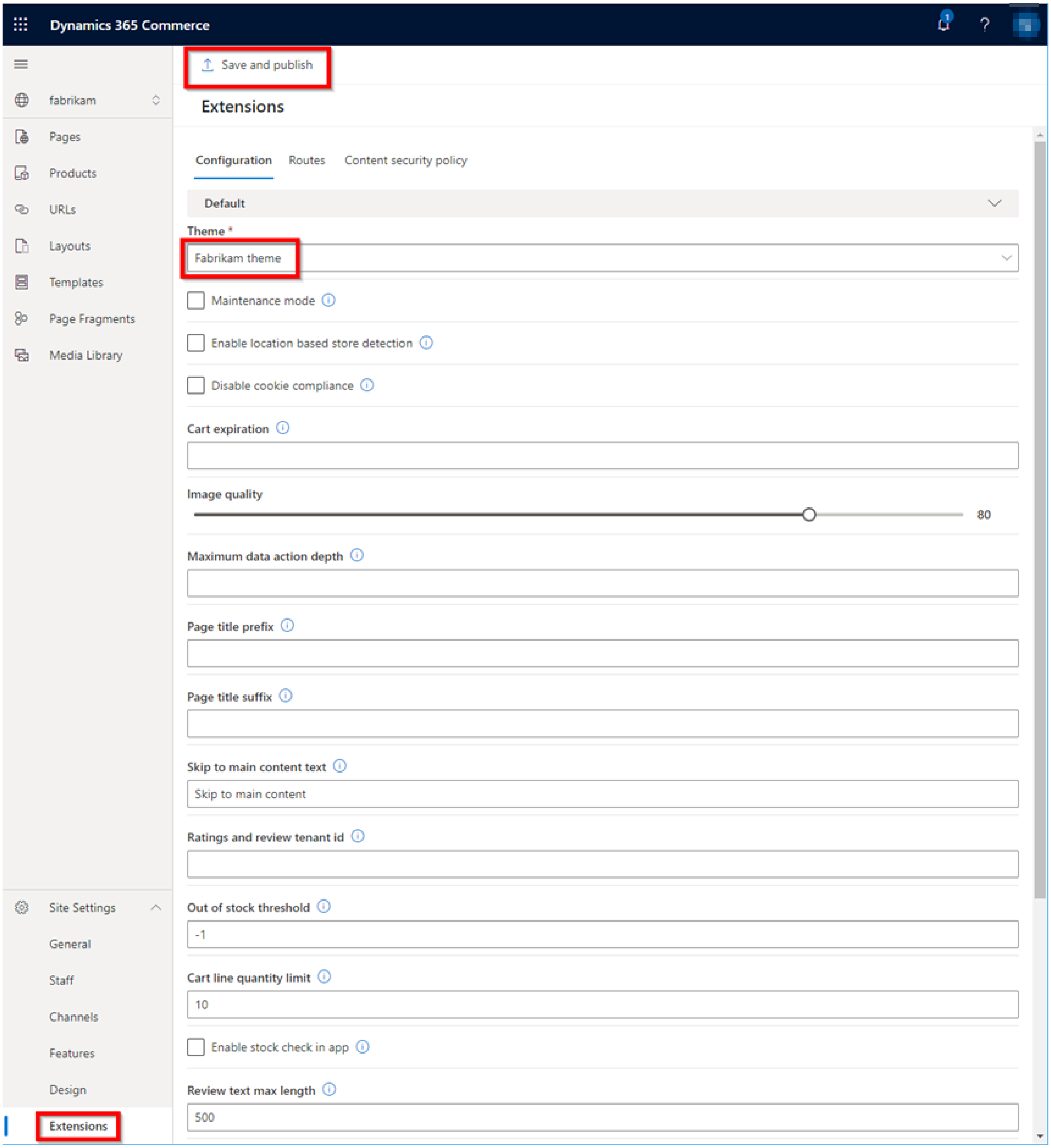
You can override the styles of the site with a single CSS file. Upload a CSS file to get started.

EXTENSIONS CONFIGURATION

The extensions configuration tab exposes various global module settings including Route and Content security policy configuration. These settings are provided as part of either the starter kit or extension modules created by partners.

If a site theme is not selected (if you created a new site), then select the “Farbikam theme” to leverage the starter kit theme for the new site. Unit 3 will teach how to create additional themes.

When complete select the “Save and publish” button



Extensions

Default

Theme *
Fabrikam theme

□ Maintenance mode

□ Enable location based store detection

□ Disable cookie compliance

Cart expiration

Image quality
80

Maximum data action depth

Page title prefix

Page title suffix

Skip to main content text
Skip to main content

Ratings and review tenant id

Out of stock threshold
-1

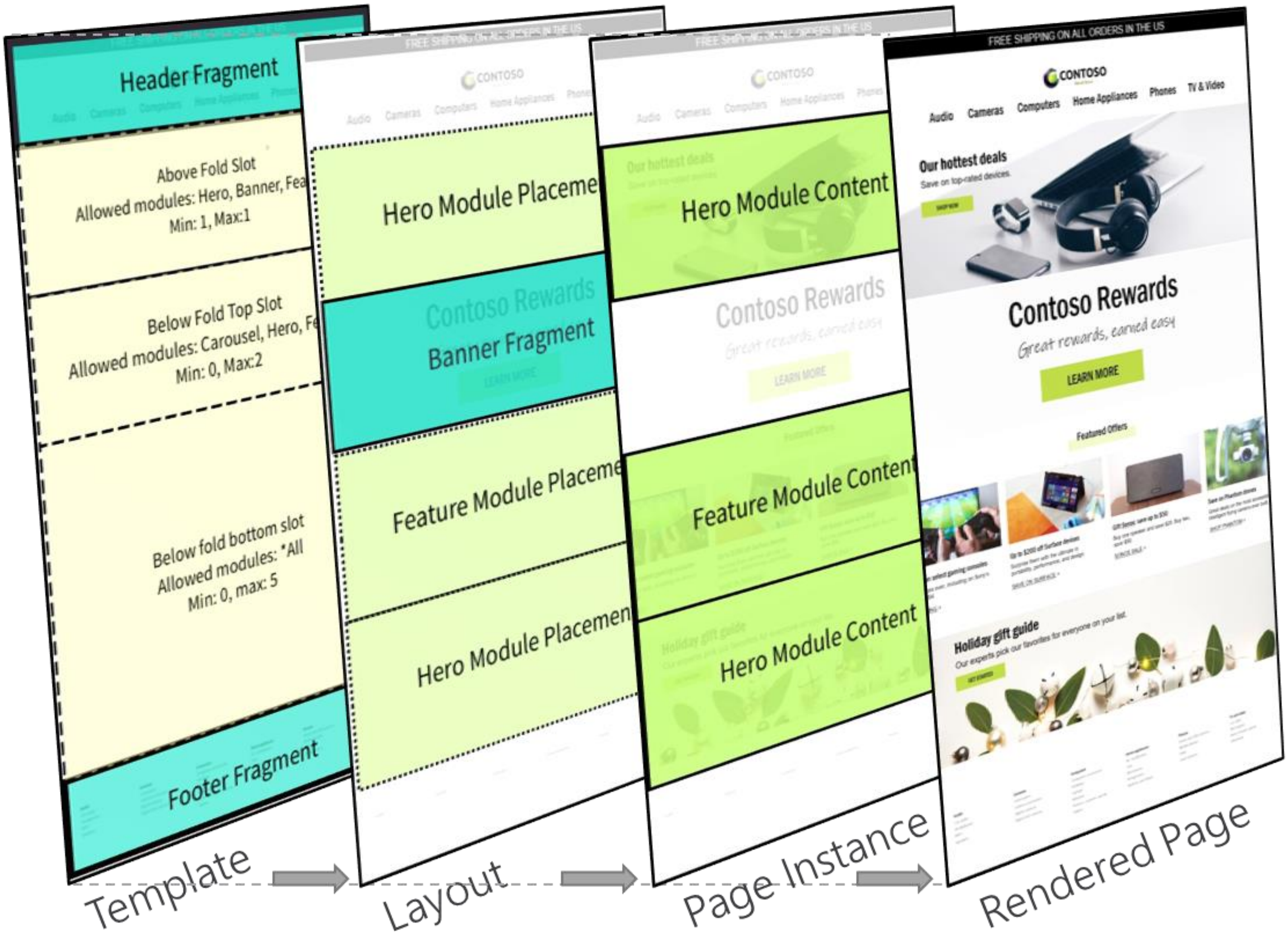
Cart line quantity limit
10

□ Enable stock check in app

Review text max length
500

COMMERCE PAGE MODEL

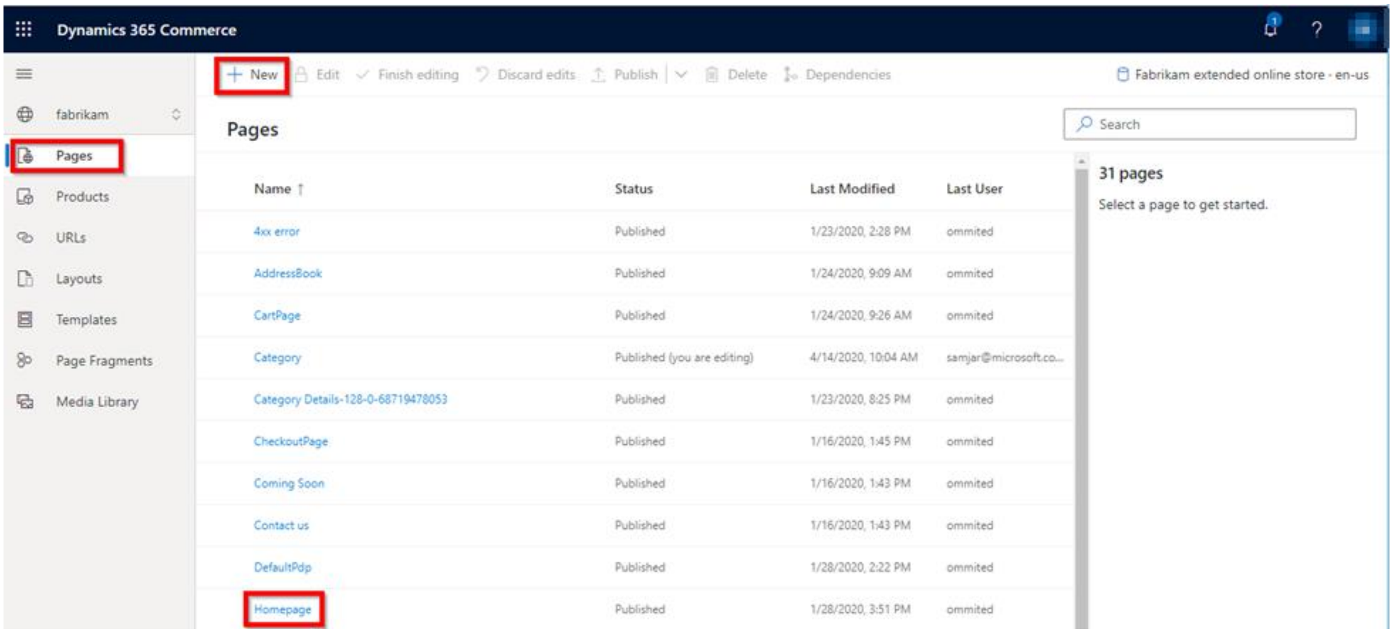
The below diagram shows the relationship between templates -> layouts -> pages and finally the rendered page.



Details for each of the main site builder tabs can be found below.

PAGES

After selecting a site, the "Pages" tab is generally the default tab that gets selected. This tab contains a list of all the pages that have been created for a site.



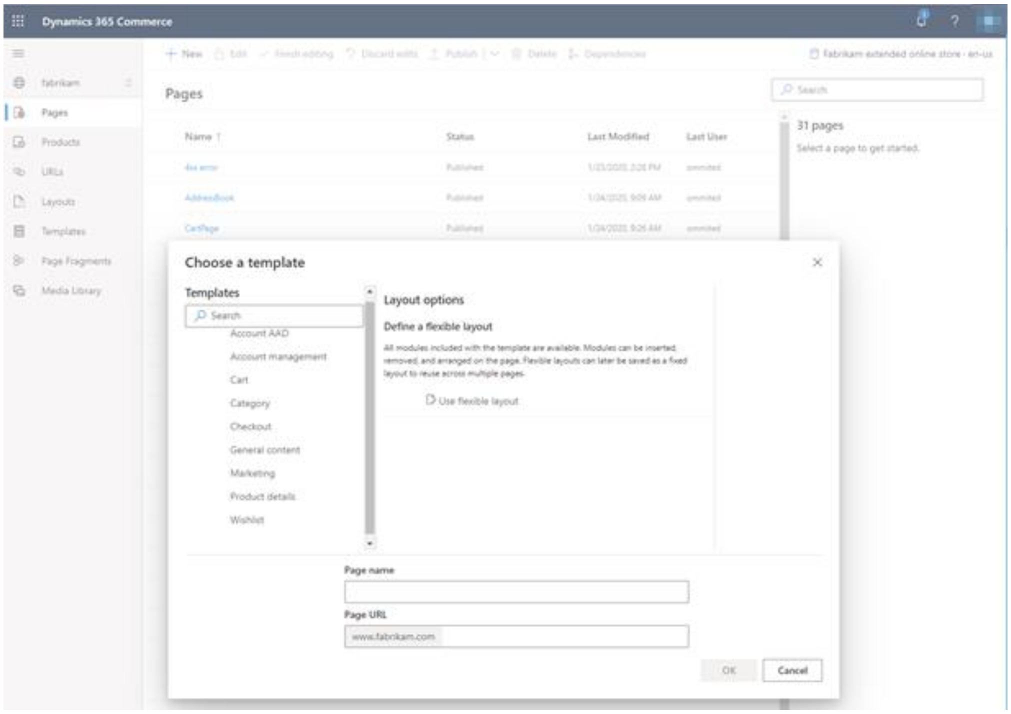
Pages

Name ↑	Status	Last Modified	Last User
4xx error	Published	1/23/2020, 2:28 PM	ommited
AddressBook	Published	1/24/2020, 9:09 AM	ommited
CartPage	Published	1/24/2020, 9:26 AM	ommited
Category	Published (you are editing)	4/14/2020, 10:04 AM	samjar@microsoft.co...
Category Details-128-0-68719478053	Published	1/23/2020, 8:25 PM	ommited
CheckoutPage	Published	1/16/2020, 1:45 PM	ommited
Coming Soon	Published	1/16/2020, 1:43 PM	ommited
Contact us	Published	1/16/2020, 1:43 PM	ommited
DefaultPdp	Published	1/28/2020, 2:22 PM	ommited
Homepage	Published	1/28/2020, 3:51 PM	ommited

31 pages

Select a page to get started.

Selecting "**New**" will bring up a dialog box where you can provide the name of the page. Every page must be based off a template or a template layout.



Choose a template

Templates

Layout options

Define a flexible layout

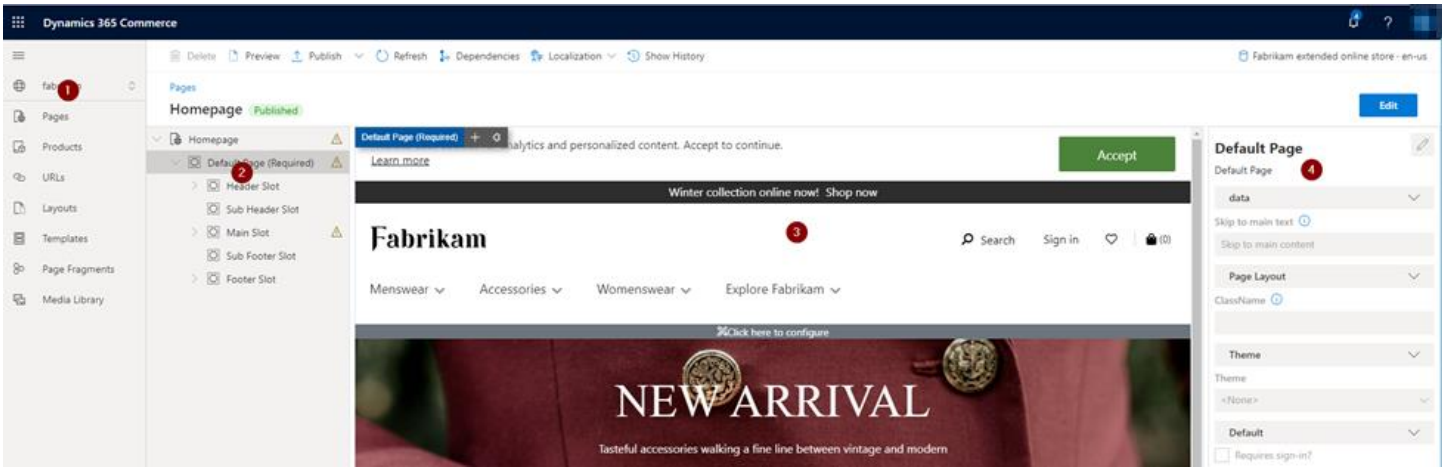
All modules included with the template are available. Modules can be inserted, removed, and arranged on the page. Flexible layouts can later be saved as a fixed layout to reuse across multiple pages.

Page name

Page URL: www.fabrikam.com

Selecting a page such as “Homepage”, will bring you into the page editor.

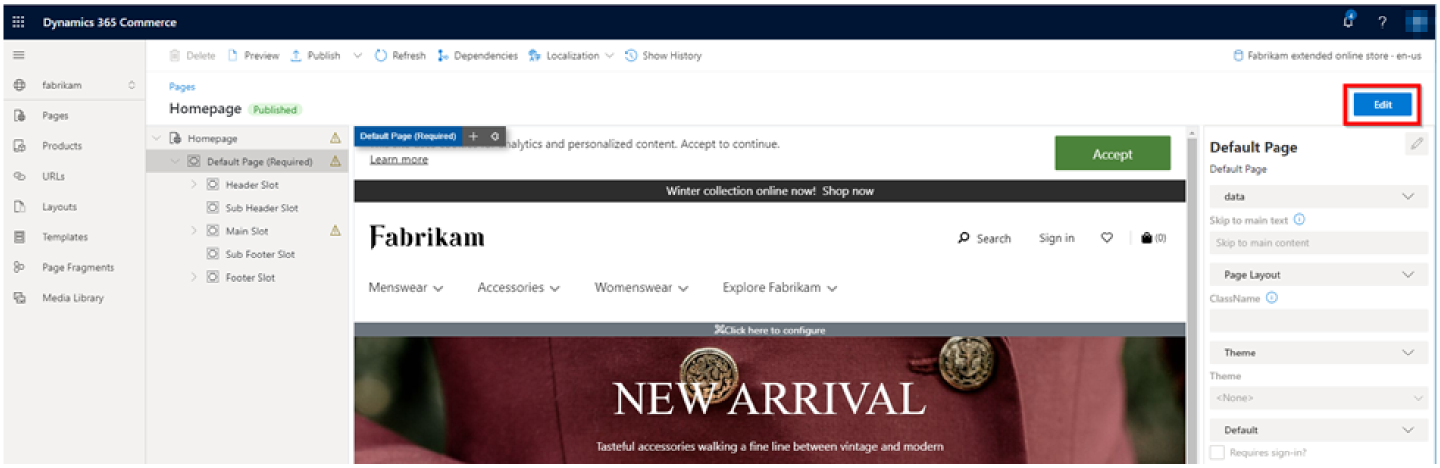
PAGE EDITOR



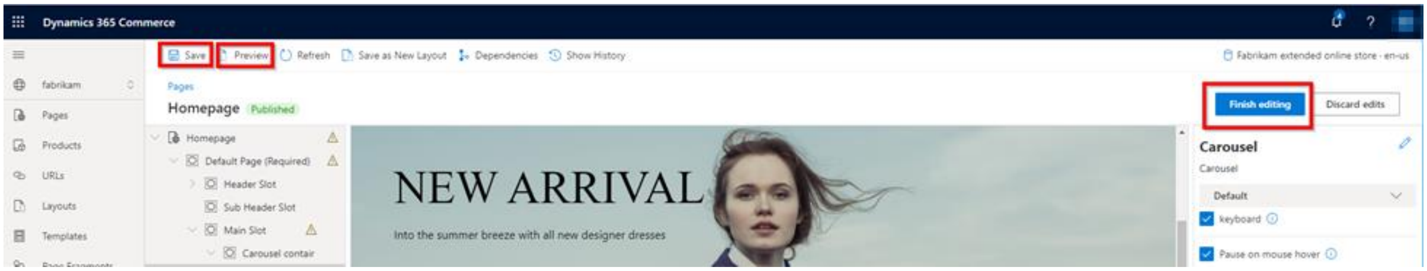
In the image above:

- Section 1 is the content management tabs; you can press the top hamburger menu to expand/collapse.
- Section 2 is the tree view of the page, this is generally a list of modules that make up the page, some modules also have layout ‘slots’ that additional modules can be placed in.
- Section 3 contains the page preview. Modules can be selected either in this section or in tree in section 2.
- Section 4 contains the selected modules configuration properties. Each module type on a page has a different set of configurations.

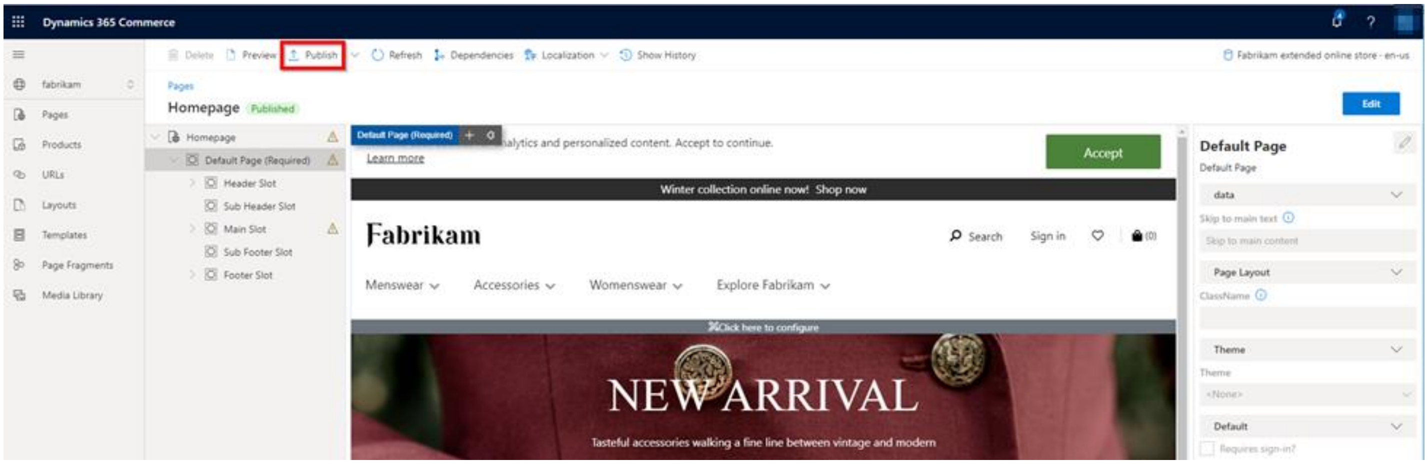
To edit a page select the edit button.



Changes can now be made. When finished with edits, the page can saved or previewed. You can then select the “Finish editing”.

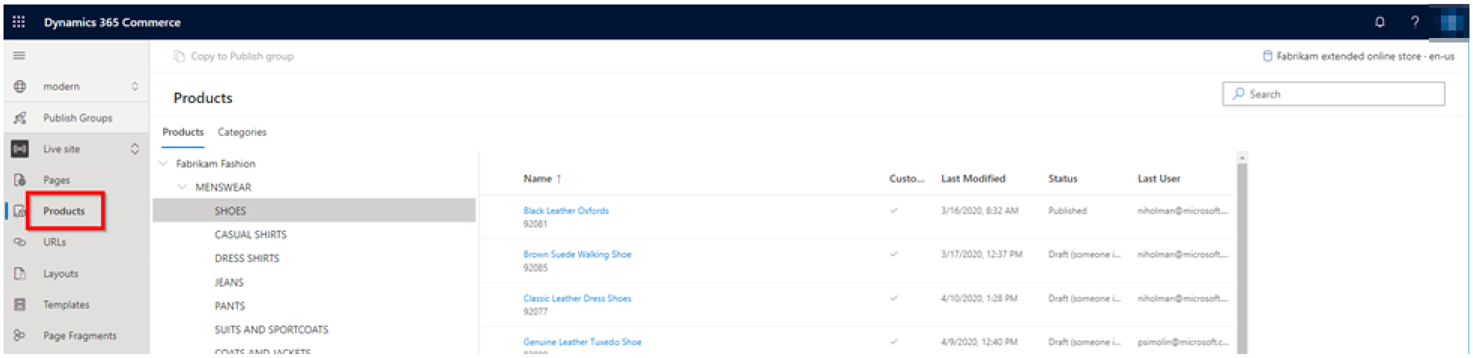


The changes will not be live until you select “Publish”.



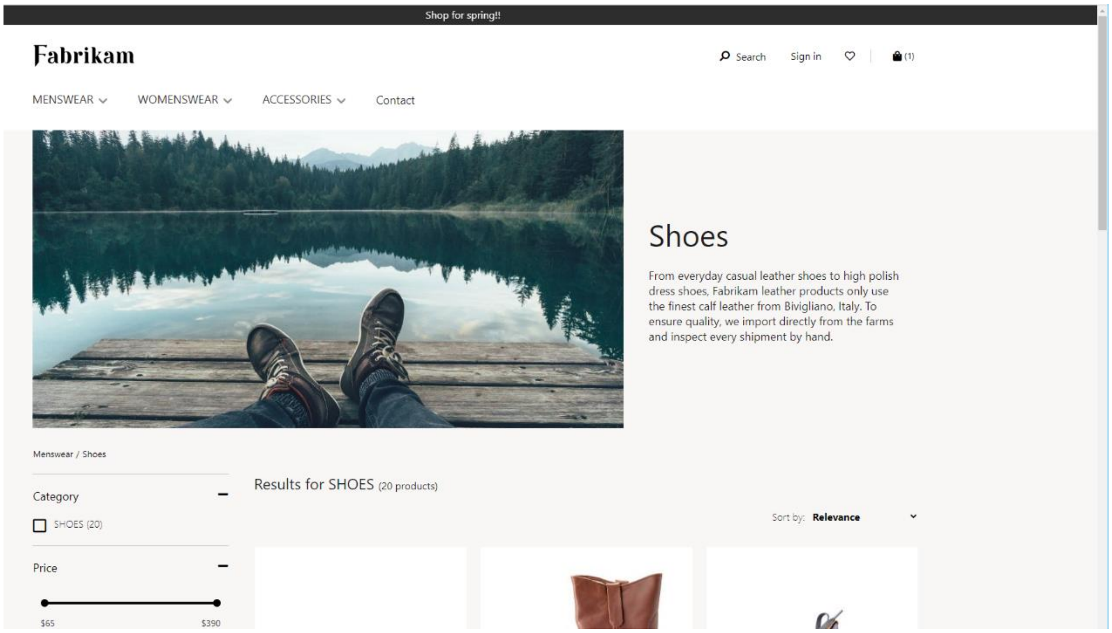
PRODUCTS

The products section allows you to view the channel products and categories and create customized pages for any product or category, this is great when promoting a product to add additional storytelling to the product or category.



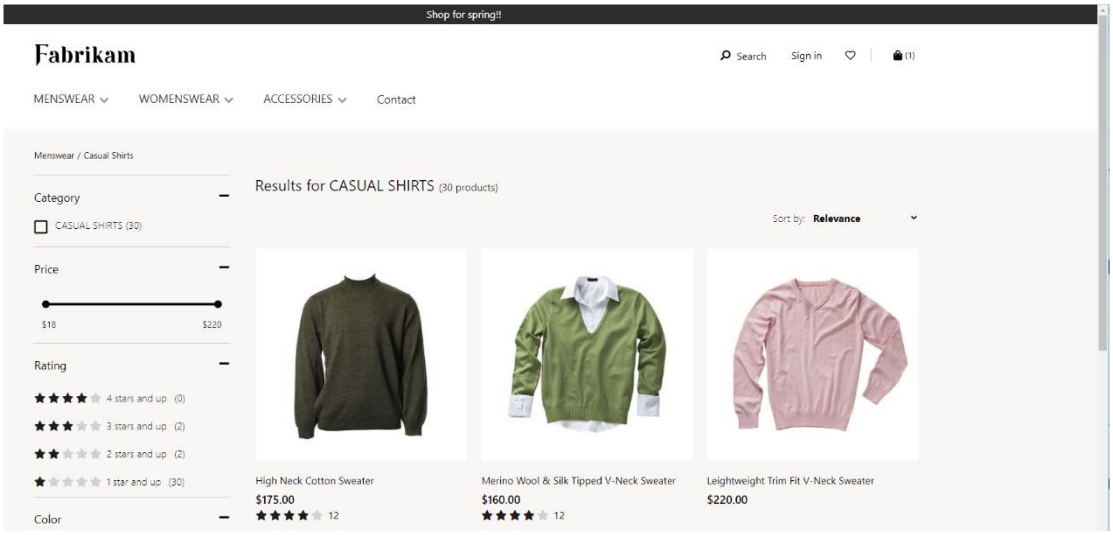
Name ↑	Custo...	Last Modified	Status	Last User
Black Leather Oxfords 92081	✓	3/16/2020, 8:32 AM	Published	niholman@microsoft...
Brown Suede Walking Shoe 92085	✓	3/17/2020, 12:37 PM	Draft (someone i...	niholman@microsoft...
Classic Leather Dress Shoes 92077	✓	4/10/2020, 1:28 PM	Draft (someone i...	niholman@microsoft...
Genuine Leather Tuxedo Shoe	✓	4/9/2020, 12:40 PM	Draft (someone i...	psimolin@microsoft.c...

For example, notice how the shoe category below has an additional image and text at the top, this category



From everyday casual leather shoes to high polish dress shoes, Fabrikam leather products only use the finest calf leather from Bivigliano, Italy. To ensure quality, we import directly from the farms and inspect every shipment by hand.

The below shows a standard category that has not been customized:



Shop for spring!!

Fabrikam

Search Sign in (1)

MENSWEAR WOMENSWEAR ACCESSORIES Contact

Menswear / Casual Shirts

Results for CASUAL SHIRTS (30 products)

Sort by: **Relevance**

Category

□ CASUAL SHIRTS (30)

Price

$18 $220

Rating

★★★★☆ 4 stars and up (0)

★★★☆☆ 3 stars and up (2)

★★☆☆☆ 2 stars and up (2)

★☆☆☆☆ 1 star and up (30)

Color

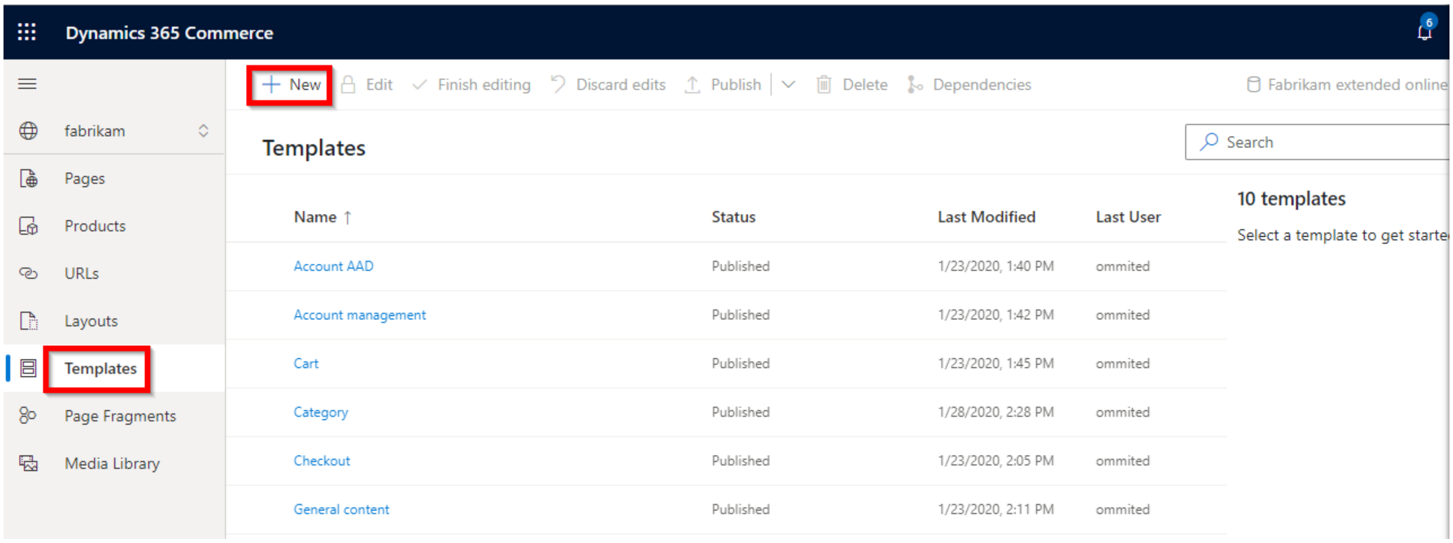
High Neck Cotton Sweater
$175.00
★★★★☆ 12

Merino Wool & Silk Tipped V-Neck Sweater
$160.00
★★★★☆ 12

Leightweight Trim Fit V-Neck Sweater
$220.00

TEMPLATES

Template contain the basic structure of a page and can be configured to tightly control the contents of a page if desired. For example, a template can choose to allow any module to appear any number of times on a page or can enforce rules such as specific must appear on a page and also limit the number of times a module can appear. Templates can also set default values for module configuration fields and even lock them so that pages that derive from them cannot change the values.



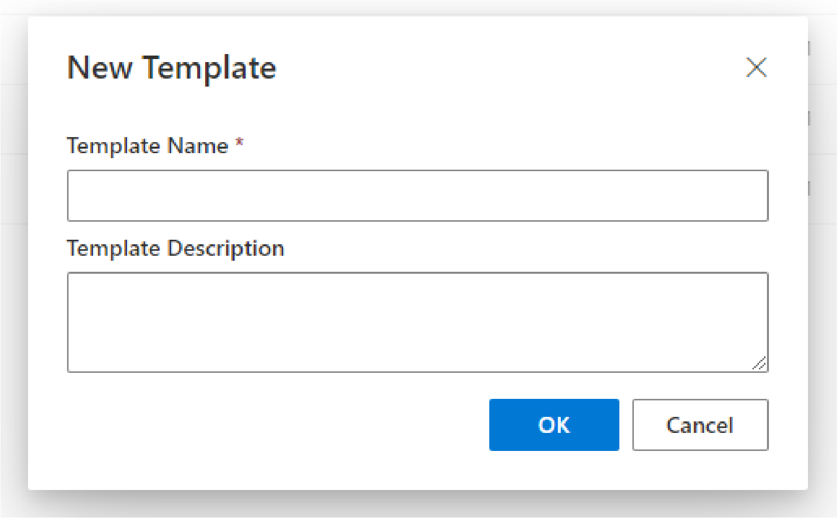
Templates

Name ↑	Status	Last Modified	Last User
Account AAD	Published	1/23/2020, 1:40 PM	ommited
Account management	Published	1/23/2020, 1:42 PM	ommited
Cart	Published	1/23/2020, 1:45 PM	ommited
Category	Published	1/28/2020, 2:28 PM	ommited
Checkout	Published	1/23/2020, 2:05 PM	ommited
General content	Published	1/23/2020, 2:11 PM	ommited

10 templates

Select a template to get starte

Select the “New” link will allow you to create a new template. You only need to provide a name and description.



New Template

Template Name *

Template Description

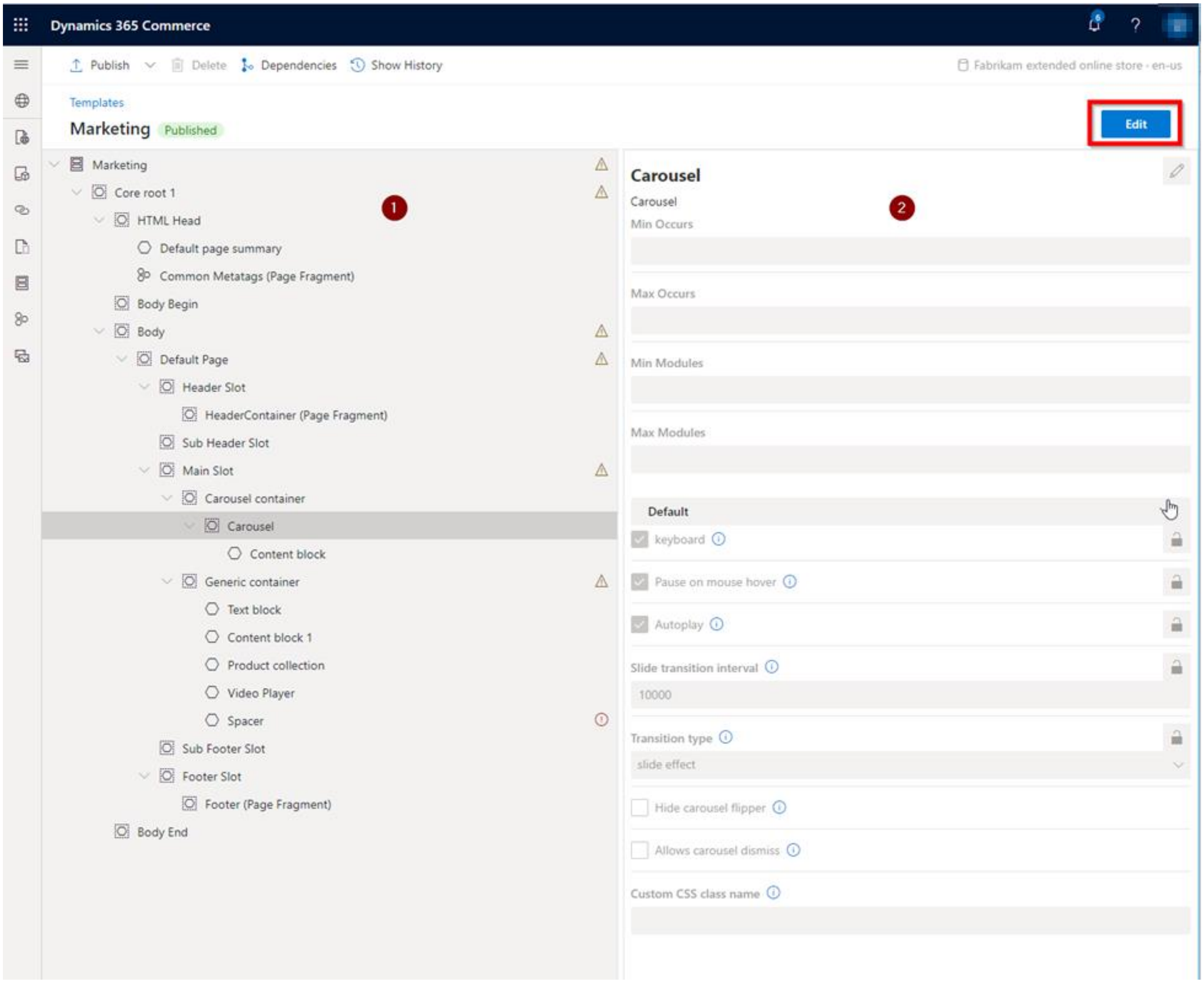
Both creating a new template or selecting an existing template will bring you to the template editor.

TEMPLATE EDITOR

The below example shows the "Marketing" template. Section 1 contains the tree view of the page and section 2 shows the configuration properties for the selected module.

To make changes to a page, like the page editor, select the "Edit" button. Once complete you can select the "Finish editing" button and "Publish" when complete.

The is no preview for a template, since the template is just a set of rules that a page inherits. For example, in the below screen shot notice, the Carousel module is selected. The Min and Max are not set in the configuration, this means that a page that uses this template can optionally add one or more Carousel modules to the page.



- Marketing
  - Core root 1
    - HTML Head
      - Default page summary
      - Common Metatags (Page Fragment)
    - Body Begin
    - Body
      - Default Page
        - Header Slot
          - HeaderContainer (Page Fragment)
        - Sub Header Slot
        - Main Slot
          - Carousel container
            - Carousel
              - Content block
          - Generic container
            - Text block
            - Content block 1
            - Product collection
            - Video Player
            - Spacer
        - Sub Footer Slot
        - Footer Slot
          - Footer (Page Fragment)
    - Body End

Carousel

Carousel

Min Occurs

Max Occurs

Min Modules

Max Modules

Default

☑ keyboard

☑ Pause on mouse hover

☑ Autoplay

Slide transition interval: 10000

Transition type: slide effect

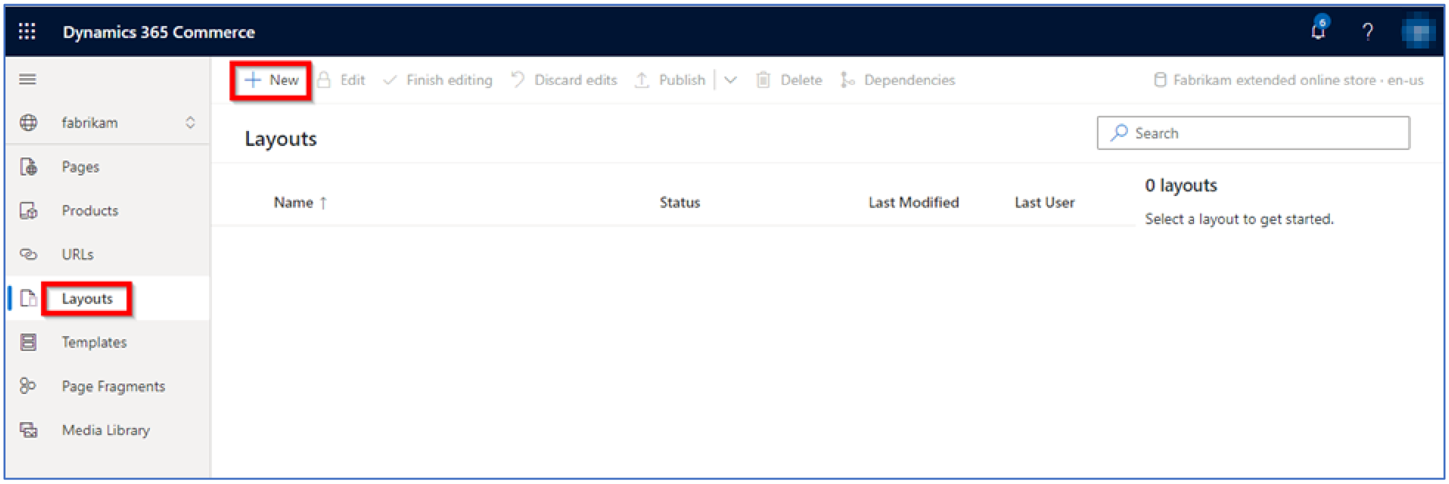
☐ Hide carousel flipper

☐ Allows carousel dismiss

Custom CSS class name

LAYOUTS

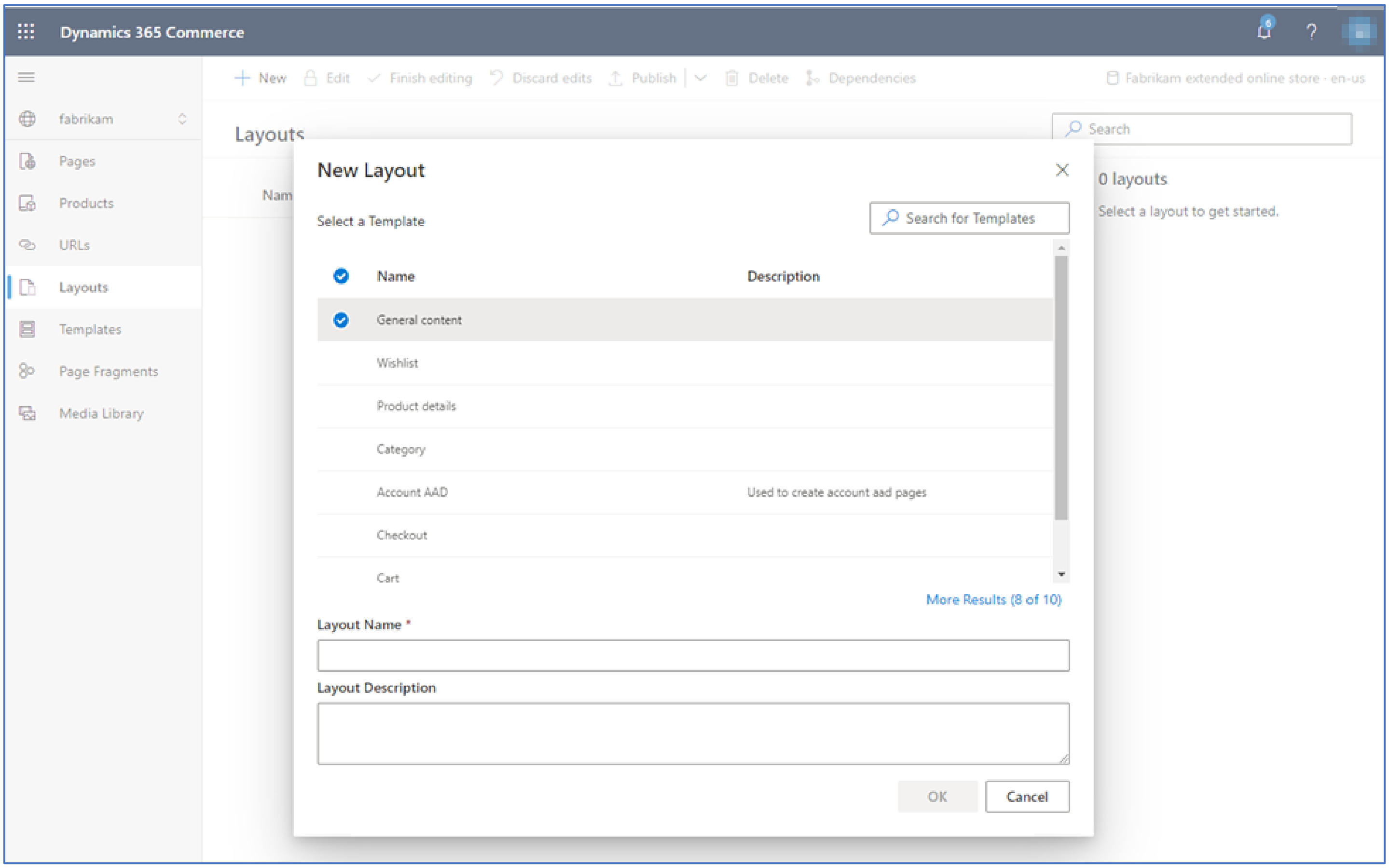
When pages are created, they must be derived from either a template or a layout. Layouts must be derived from a single template. For example, you could create a product details page template that is generic offering a wide level of page optimizations. You may then create several different layouts for the different product types, perhaps clothing will get additional size charts on the pages, and sporting goods will get different manufacturing information. The modules for the two layouts could be layout out completely differently thus the end pages for either would look quite different even though they both still adhere to the same template.



0 layouts

Select a layout to get started.

Selecting “New” will bring up the “New Layout” dialog box as shown below. A base template must be chosen for the layout along with a name and optional description.



New Layout

Select a Template

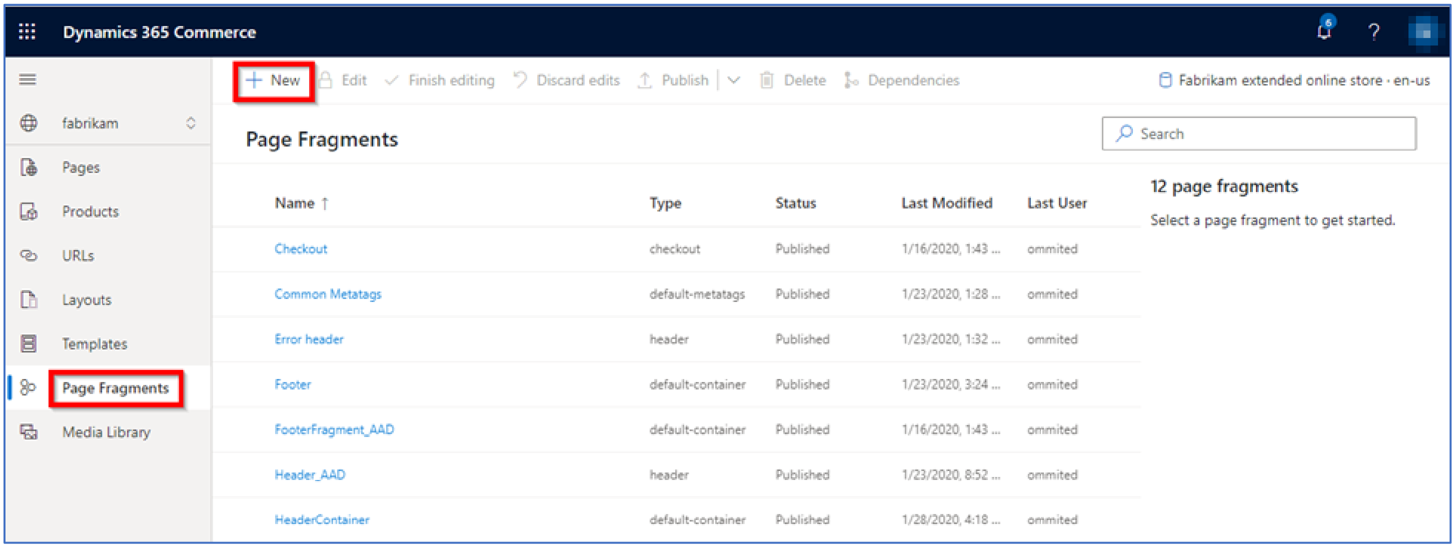
Name	Description
General content	
Wishlist	
Product details	
Category	
Account AAD	Used to create account aad pages
Checkout	
Cart	

Layout Name *

Layout Description

PAGE FRAGMENTS

Page fragments allow the creation and configuration of a module or a group of modules that can then be added to a page or template. Fragments are great for creating items such as the header or footer that will show up in most of the site pages. They can then be configured in the fragment editor and once saved and published will be updated on all pages.



Page Fragments

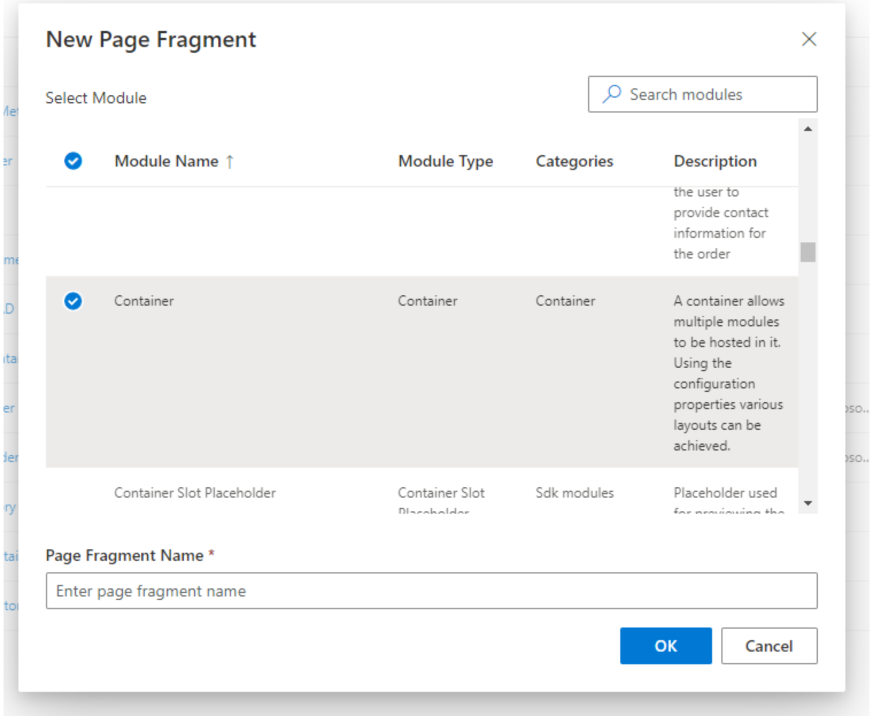
Name ↑	Type	Status	Last Modified	Last User
Checkout	checkout	Published	1/16/2020, 1:43 ...	ommited
Common Metatags	default-metatags	Published	1/23/2020, 1:28 ...	ommited
Error header	header	Published	1/23/2020, 1:32 ...	ommited
Footer	default-container	Published	1/23/2020, 3:24 ...	ommited
FooterFragment_AAD	default-container	Published	1/16/2020, 1:43 ...	ommited
Header_AAD	header	Published	1/23/2020, 8:52 ...	ommited
HeaderContainer	default-container	Published	1/28/2020, 4:18 ...	ommited

12 page fragments

Select a page fragment to get started.

Groups of modules on a page can also be saved as a fragment, this is great if you created and configured modules on a page and decide you'd like to also include it on another page, when this is done, the fragment can only be globally updated and the same configuration will appear on all pages.

Selecting "**New**" will bring up the "**New Page Fragment**" editor as shown below.



New Page Fragment

Select Module

	Module Name ↑	Module Type	Categories	Description
				the user to provide contact information for the order
✓	Container	Container	Container	A container allows multiple modules to be hosted in it. Using the configuration properties various layouts can be achieved.
	Container Slot Placeholder	Container Slot Placeholder	Sdk modules	Placeholder used for previewing the

Page Fragment Name *

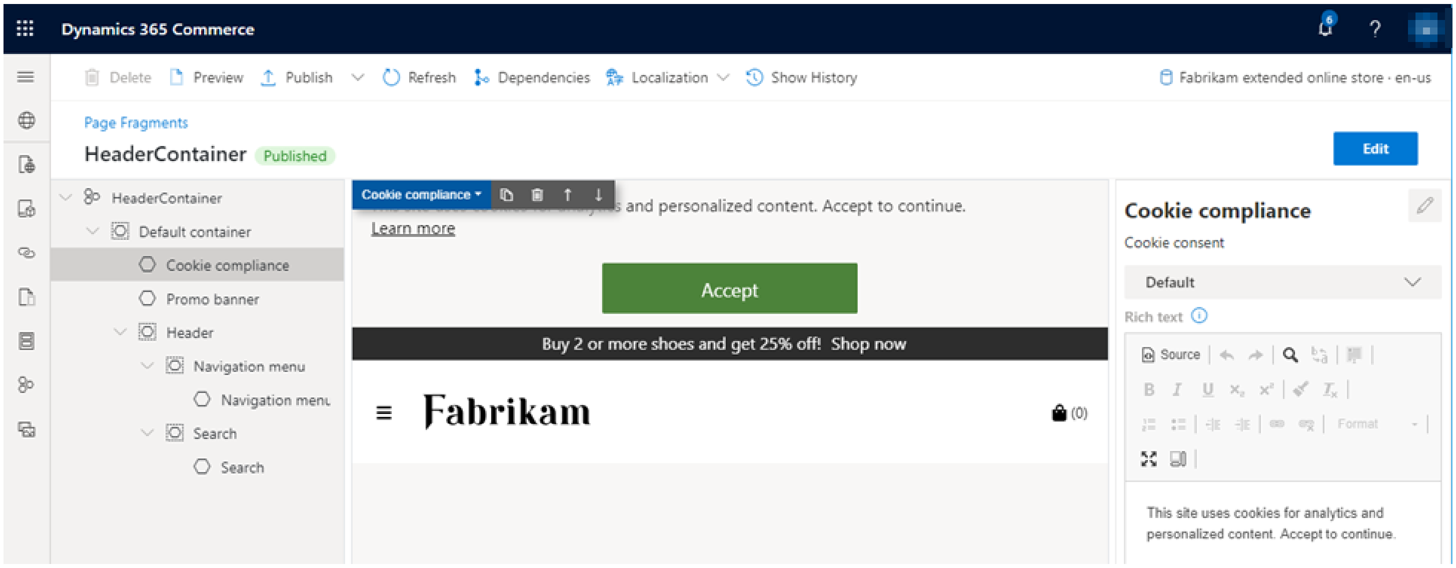
Enter page fragment name

You will need to select the root module of the fragment, the “Container” module is a good generic starting point for many fragments, then any modules can further be placed inside the container.

Selecting a module will bring up the fragment editor.

FRAGMENT EDITOR

The below screen shot shows a sample header fragment, notice how the fragment is just a set of modules and their configurations. The same “Edit”, “Save” and “Publish” methods can be found on this editor as in the Page, Template and Layouts editor.



…s and personalized content. Accept to continue.

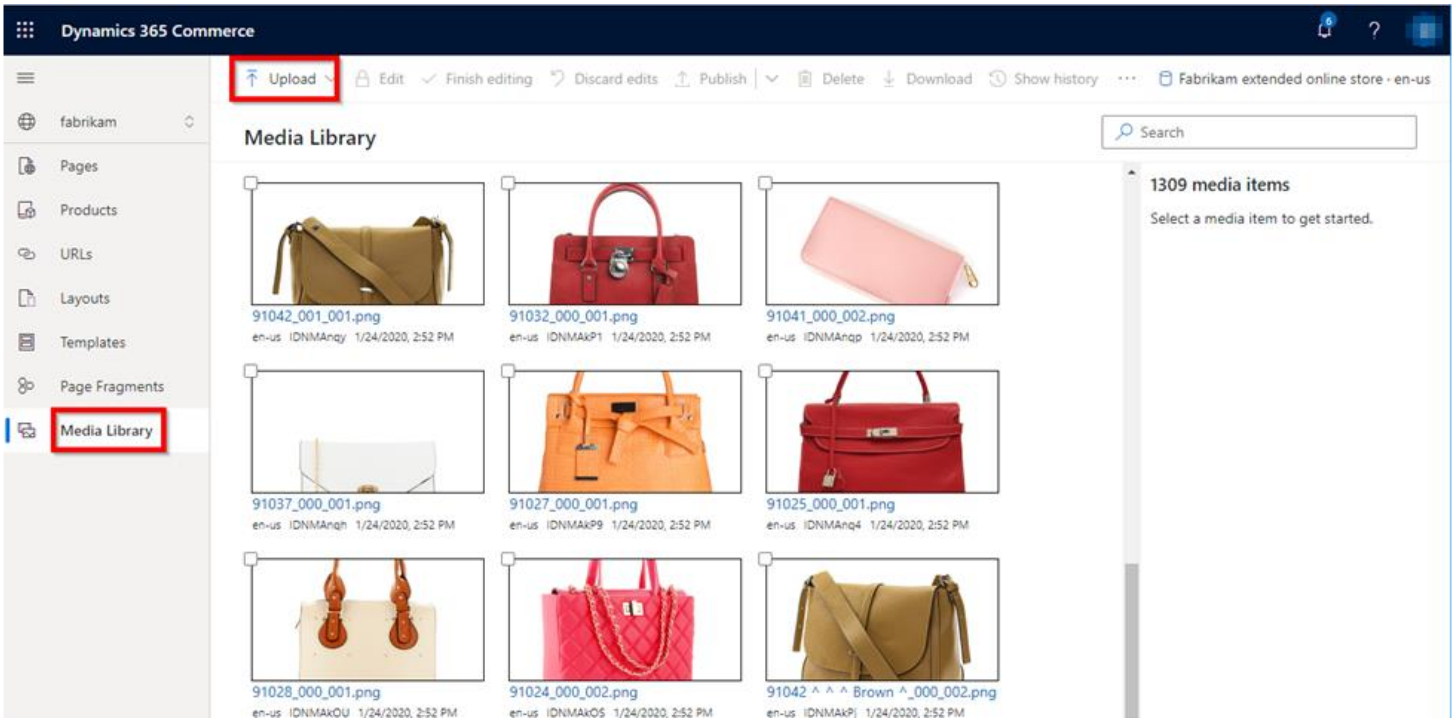
This site uses cookies for analytics and personalized content. Accept to continue.

MEDIA LIBRARY

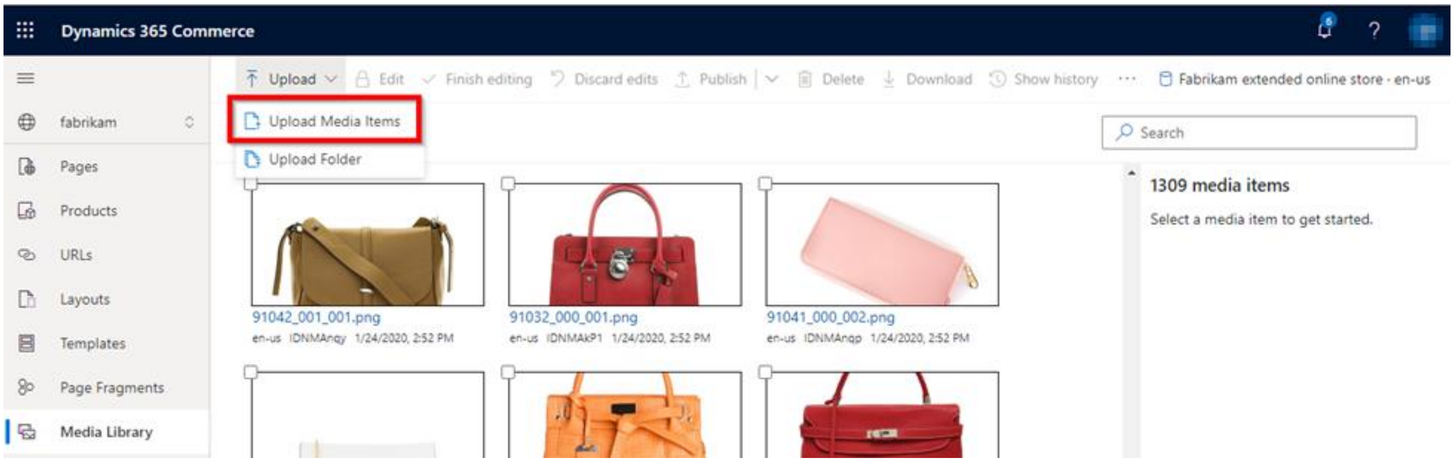
The media library is the store for all images used in the e-Commerce site, it can also be used to hold images for the point of sale system.

When you select the “Media Library” you should see a list of previously uploaded images. If you are using the Fabrikam site, you should see existing product and marketing images used with the demo data.

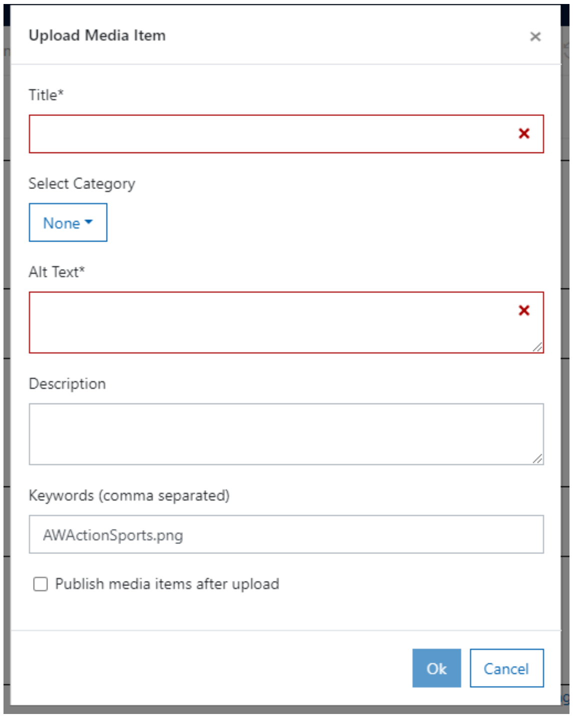
To upload a new image, select the “Upload” button.


Dynamics 365 Commerce
Upload
Edit
Finish editing
Discard edits
Publish
Delete
Download
Show history
Fabrikam extended online store - en-us
fabrikam
Pages
Products
URLs
Layouts
Templates
Page Fragments
Media Library
Search
1309 media items
Select a media item to get started.
91042_001_001.png
en-us IDNMAnqy 1/24/2020, 2:52 PM
91032_000_001.png
en-us IDNMAkP1 1/24/2020, 2:52 PM
91041_000_002.png
en-us IDNMAnqp 1/24/2020, 2:52 PM
91037_000_001.png
en-us IDNMAnqh 1/24/2020, 2:52 PM
91027_000_001.png
en-us IDNMAkP9 1/24/2020, 2:52 PM
91025_000_001.png
en-us IDNMAnq4 1/24/2020, 2:52 PM
91028_000_001.png
en-us IDNMAkOU 1/24/2020, 2:52 PM
91024_000_002.png
en-us IDNMAkOS 1/24/2020, 2:52 PM
91042 ^ ^ ^ Brown ^_000_002.png
en-us IDNMAkPi 1/24/2020, 2:52 PM

You’ll see two options, one to “Upload Media Items” and two to “Upload Folder” which will upload a whole folder of images at a time.



If you select “Upload Media Items” and select an image you will then be prompted for the image title and alt text and a category. The title and alt text are mandatory unless you select multiple images.



Upload Media Item

Title*

Select Category

None

Alt Text*

Description

Keywords (comma separated)

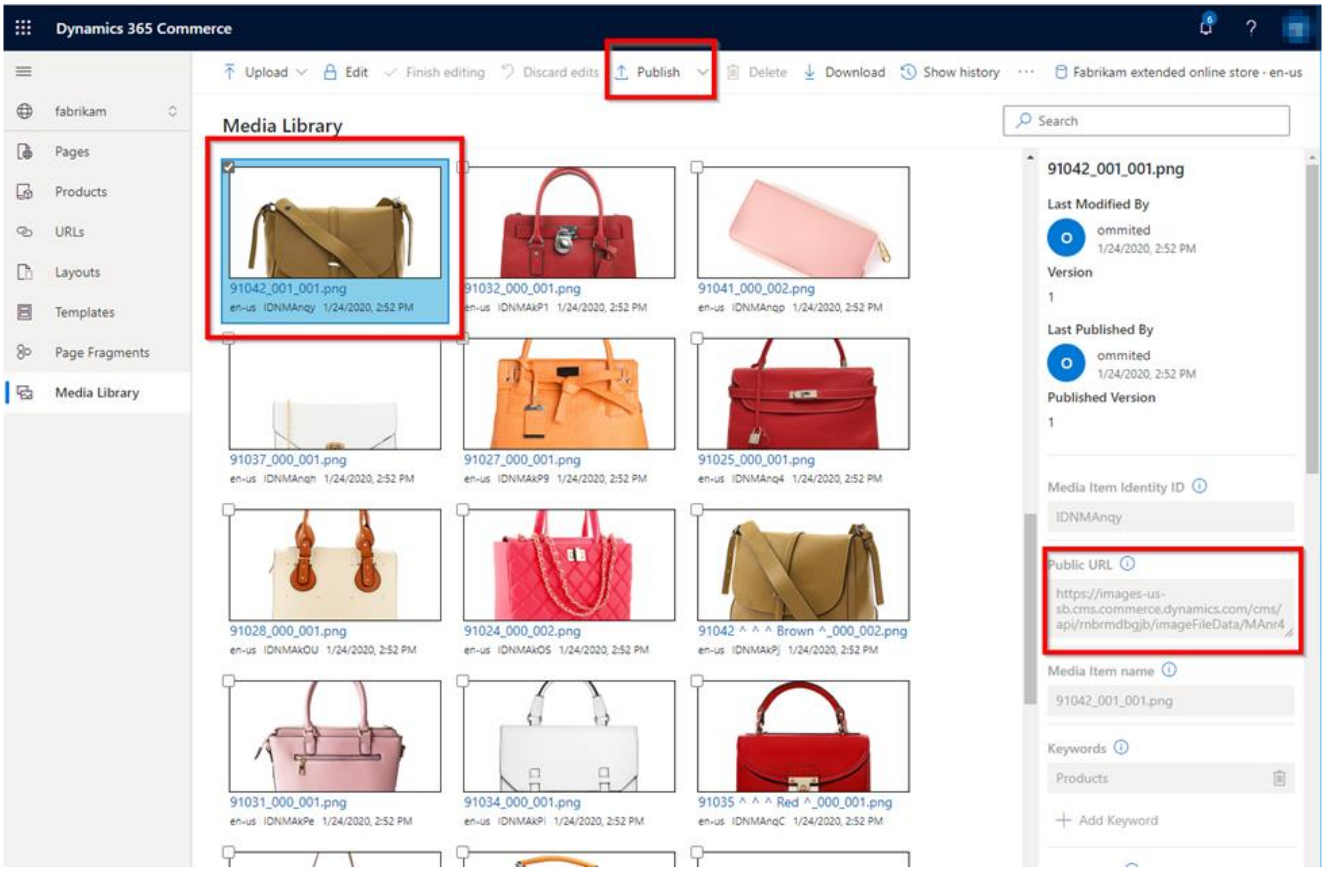
AWActionSports.png

□ Publish media items after upload

The category type is very important, if it is just marketing images leave it as “None”, if it’s a product image you must select type “Product”, there are also other types used by the point of sale for category images, employees, …

It is also handy to select “Publish media item after upload” otherwise it will not show up on the site. However, if missed you can select the image and publish as shown below.

https://images-us-sb.cms.commerce.dynamics.com/cms/api/qqdgtmsdp/imageFileData/search?fileName=


Dynamics 365 Commerce
Upload
Edit
Finish editing
Discard edits
Publish
Delete
Download
Show history
Fabrikam extended online store - en-us
fabrikam
Pages
Products
URLs
Layouts
Templates
Page Fragments
Media Library
Search
91042_001_001.png en-us IDNMAnqy 1/24/2020, 2:52 PM
91032_000_001.png en-us IDNMAkP1 1/24/2020, 2:52 PM
91041_000_002.png en-us IDNMAnqp 1/24/2020, 2:52 PM
91037_000_001.png en-us IDNMAnqh 1/24/2020, 2:52 PM
91027_000_001.png en-us IDNMAkP9 1/24/2020, 2:52 PM
91025_000_001.png en-us IDNMAnq4 1/24/2020, 2:52 PM
91028_000_001.png en-us IDNMAkOU 1/24/2020, 2:52 PM
91024_000_002.png en-us IDNMAkOS 1/24/2020, 2:52 PM
91042 ^ ^ ^ Brown ^_000_002.png en-us IDNMAkPj 1/24/2020, 2:52 PM
91031_000_001.png en-us IDNMAkPe 1/24/2020, 2:52 PM
91034_000_001.png en-us IDNMAkPl 1/24/2020, 2:52 PM
91035 ^ ^ ^ Red ^_000_001.png en-us IDNMAnqC 1/24/2020, 2:52 PM
Add Keyword

Media Library

91042_001_001.png

Last Modified By
ommited
1/24/2020, 2:52 PM

Version
1

Last Published By
ommited
1/24/2020, 2:52 PM

Published Version
1

Media Item Identity ID
IDNMAnqy

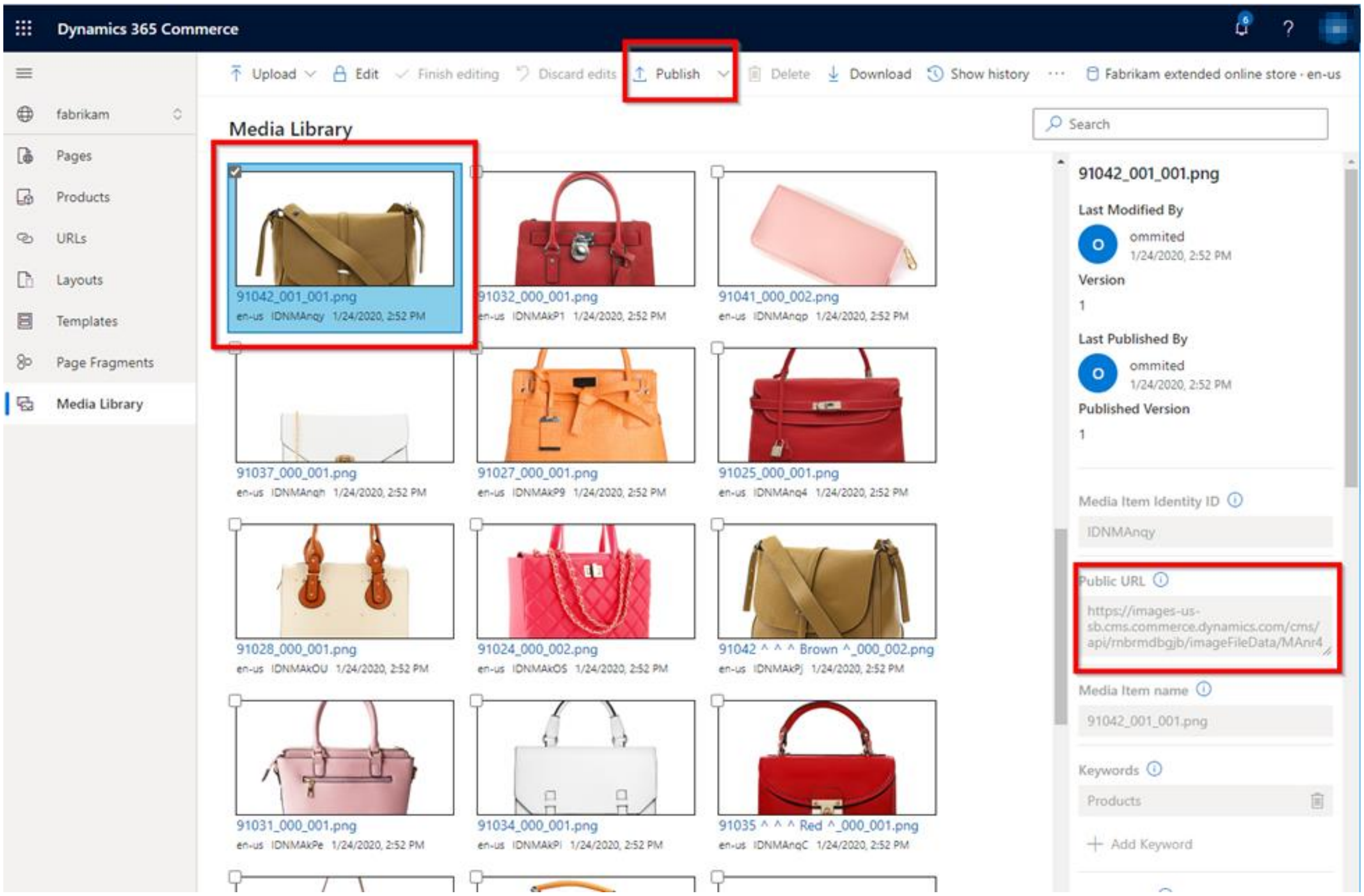
Public URL
https://images-us-sb.cms.commerce.dynamics.com/cms/api/rnbrmdbgjb/imageFileData/MAnr4

Media Item name
91042_001_001.png

Keywords
Products

Notice the “Public URL” can be used to see the exposed URL for the image.

To delete an image, you must first Unpublish which is available in the “Publish” dropdown as highlighted in the image below.


Dynamics 365 Commerce
fabrikam
Pages
Products
URLs
Layouts
Templates
Page Fragments
Media Library
Upload
Edit
Finish editing
Discard edits
Publish
Delete
Download
Show history
Fabrikam extended online store · en-us
Media Library
Search
91042_001_001.png
en-us IDNMAnqy 1/24/2020, 2:52 PM
91032_000_001.png
en-us IDNMAkP1 1/24/2020, 2:52 PM
91041_000_002.png
en-us IDNMAnqp 1/24/2020, 2:52 PM
91037_000_001.png
en-us IDNMAngh 1/24/2020, 2:52 PM
91027_000_001.png
en-us IDNMAkP9 1/24/2020, 2:52 PM
91025_000_001.png
en-us IDNMAnq4 1/24/2020, 2:52 PM
91028_000_001.png
en-us IDNMAkOU 1/24/2020, 2:52 PM
91024_000_002.png
en-us IDNMAkOS 1/24/2020, 2:52 PM
91042 ^ ^ ^ Brown ^_000_002.png
en-us IDNMAkPj 1/24/2020, 2:52 PM
91031_000_001.png
en-us IDNMAkPe 1/24/2020, 2:52 PM
91034_000_001.png
en-us IDNMAkPl 1/24/2020, 2:52 PM
91035 ^ ^ ^ Red ^_000_001.png
en-us IDNMAnqC 1/24/2020, 2:52 PM
Add Keyword

91042_001_001.png

Last Modified By
ommited
1/24/2020, 2:52 PM

Version
1

Last Published By
ommited
1/24/2020, 2:52 PM

Published Version
1

Media Item Identity ID
IDNMAnqy

Public URL
https://images-us-sb.cms.commerce.dynamics.com/cms/api/rnbrmdbgjb/imageFileData/MAnr4

Media Item name
91042_001_001.png

Keywords
Products

UPLOADING PRODUCT IMAGES

Earlier in this guide we added a new product (we used a sample backpack) and added it to the channel hierarchy and assorted it. You should be able to find it in the category view and notice that it has no product image.

Note: You should be provided with a zip file with assets we will use in this section that can be used if you configured a backpack.

In this section we will upload a set of product images in the "Media Library".

PRODUCT IMAGE FILE NAME

The file names are very important and follow a pattern used for the master product image and its variants. It follows the same pattern used in Commerce POS today.

Notice there is a file "75001_000_001.png" this represents the first master image, you can provide up to five (PRODUCTNUMBER_000_001.png, PRODUCTNUMBER_000_002.png,…, PRODUCTNUMBER_000_005.png) . The first part of the file name before the "_" separator is the product id provided in Headquarters when the product was created.

The other image names would be "75001_000_002.png" … "75001_000_005.png".

You can also supply up to 5 images for each variant (size/style/color). Each variant can be specified between special "^" separators. For example, our backpack supports three sizes so notice we have provided 3 additional images:

75001 ^ ^ Small ^ ^_000_001.png

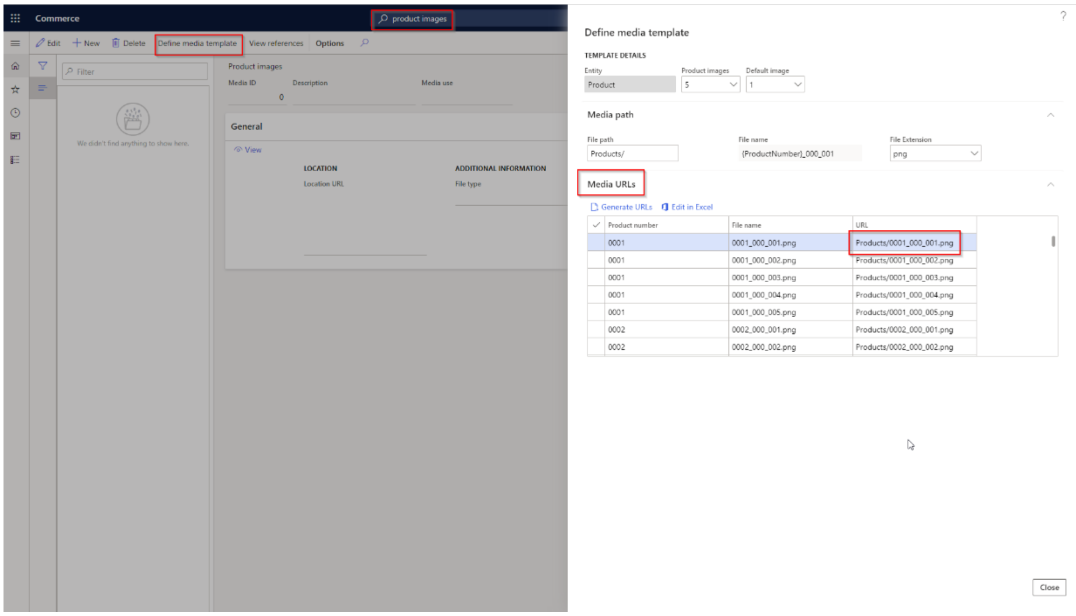
75001 ^ ^ Medium ^ ^_000_001.png

75001 ^ ^ Large ^ ^_000_001.png

The pattern uses "**ProductID ^ CONFIGURATION ^ SIZE ^ COLOR ^ STYLE_000_00x.png**" (where x can be between 1-5).

An example for a size 8, light blue(color) wide (style) shoe variant is: "**92074 ^ ^ 8 ^ Light Blue ^ Wide_000_001.png**"

Another way to see what product image file names should be is from Headquarters search for the "**Product images**" page and select "**Define media template**" as shown below.



Product images

Media ID: 0

Description

Media use

General

LOCATION

Location URL

ADDITIONAL INFORMATION

File type

Define media template

TEMPLATE DETAILS

Entity: Product

Product images: 5

Default image: 1

Media path

File path: Products/

File name: {ProductNumber}_000_001

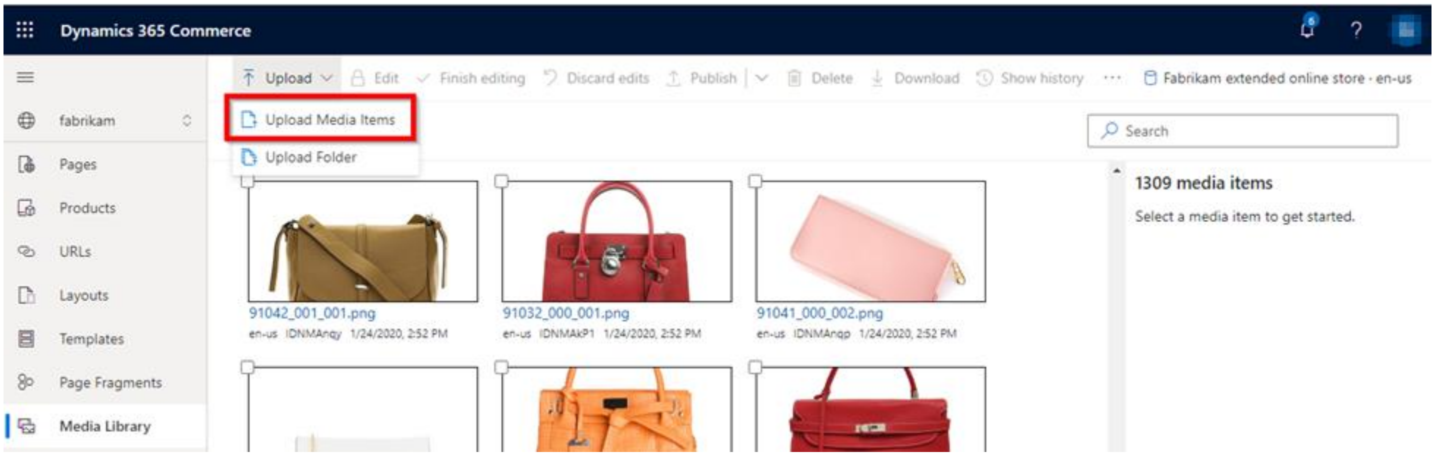
File Extension: png

Media URLs

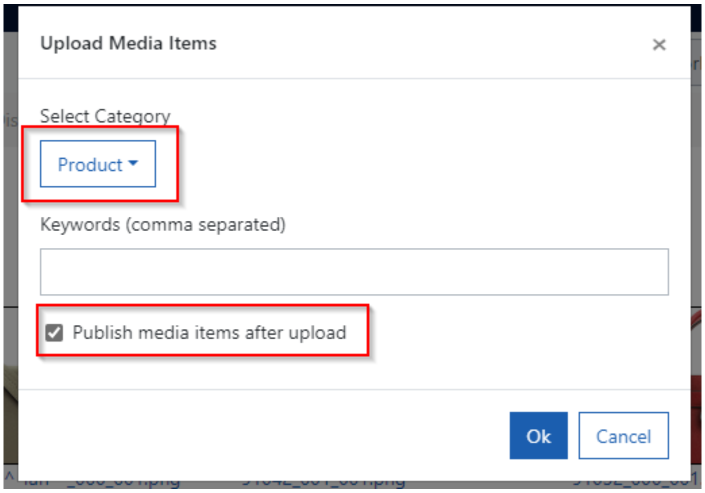
Product number	File name	URL
0001	0001_000_001.png	Products/0001_000_001.png
0001	0001_000_002.png	Products/0001_000_002.png
0001	0001_000_003.png	Products/0001_000_003.png
0001	0001_000_004.png	Products/0001_000_004.png
0001	0001_000_005.png	Products/0001_000_005.png
0002	0002_000_001.png	Products/0002_000_001.png
0002	0002_000_002.png	Products/0002_000_002.png

UPLOADING IMAGES

From within the site builder tool, select the "**Media Library**" tab followed by the "**Upload**" button and then "**Upload Media Items**". Navigate to the directory with the Adventure Works assets and select the four backpack images and click "Open" (or upload any images you need for the products you created – see note below on file naming).



Select "**Product**" as the category (this is very important and not easy to change after uploading). Enable the "**Publish assets after upload**" checkbox and click "**OK**".

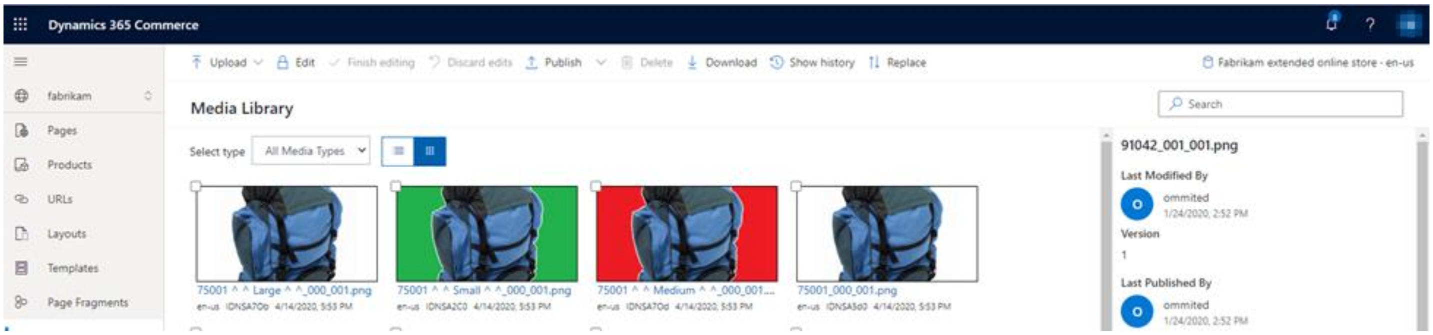

Upload Media Items

Select Category: Product

Keywords (comma separated):

☑ Publish media items after upload

You should now see all of the product images in the media library.



Select type: All Media Types

75001 ^ ^ Large ^ ^_000_001.png
en-us IDNSA7Ob 4/14/2020, 5:53 PM

75001 ^ ^ Small ^ ^_000_001.png
en-us IDNSA2C0 4/14/2020, 5:53 PM

75001 ^ ^ Medium ^ ^_000_001....
en-us IDNSA7Od 4/14/2020, 5:53 PM

75001_000_001.png
en-us IDNSA5d0 4/14/2020, 5:53 PM

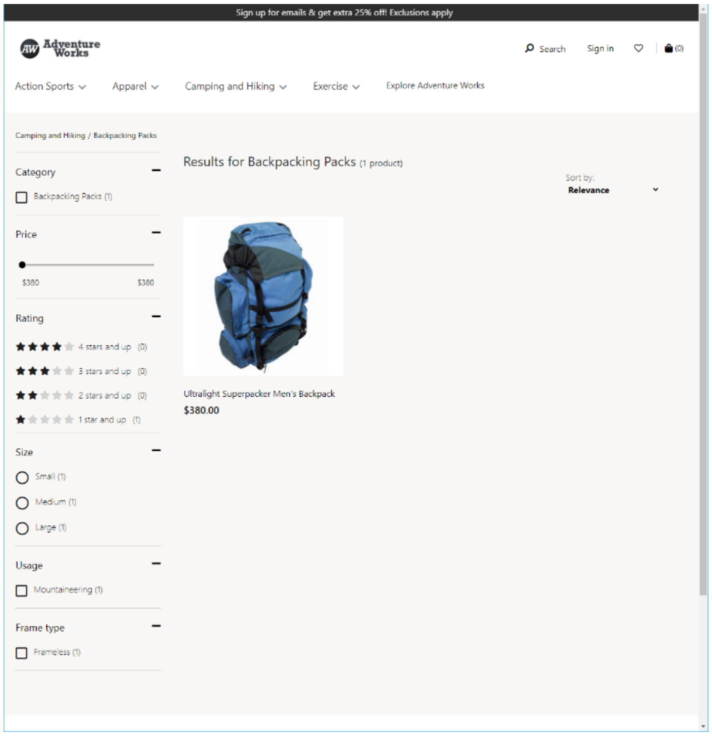
91042_001_001.png

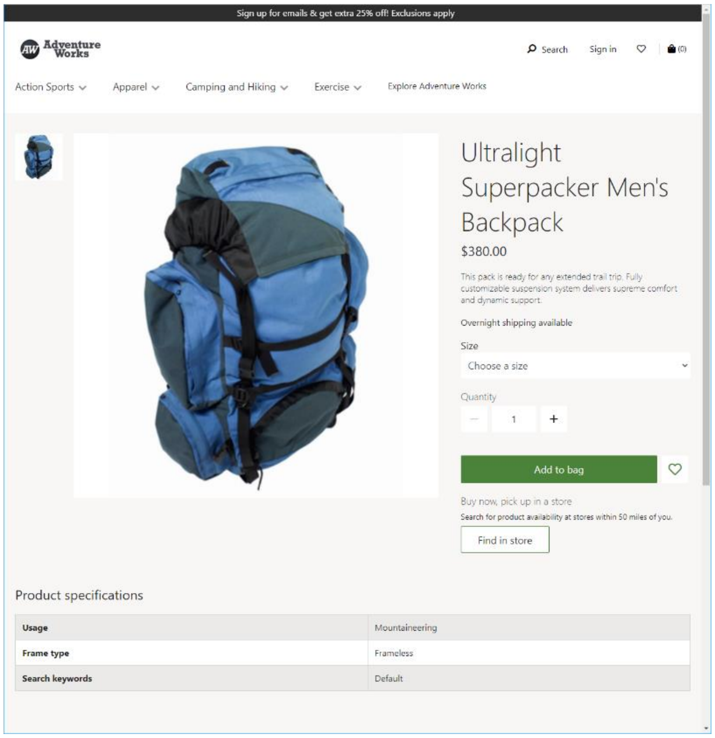
Last Modified By
ommited
1/24/2020, 2:52 PM

Version
1

Last Published By
ommited
1/24/2020, 2:52 PM

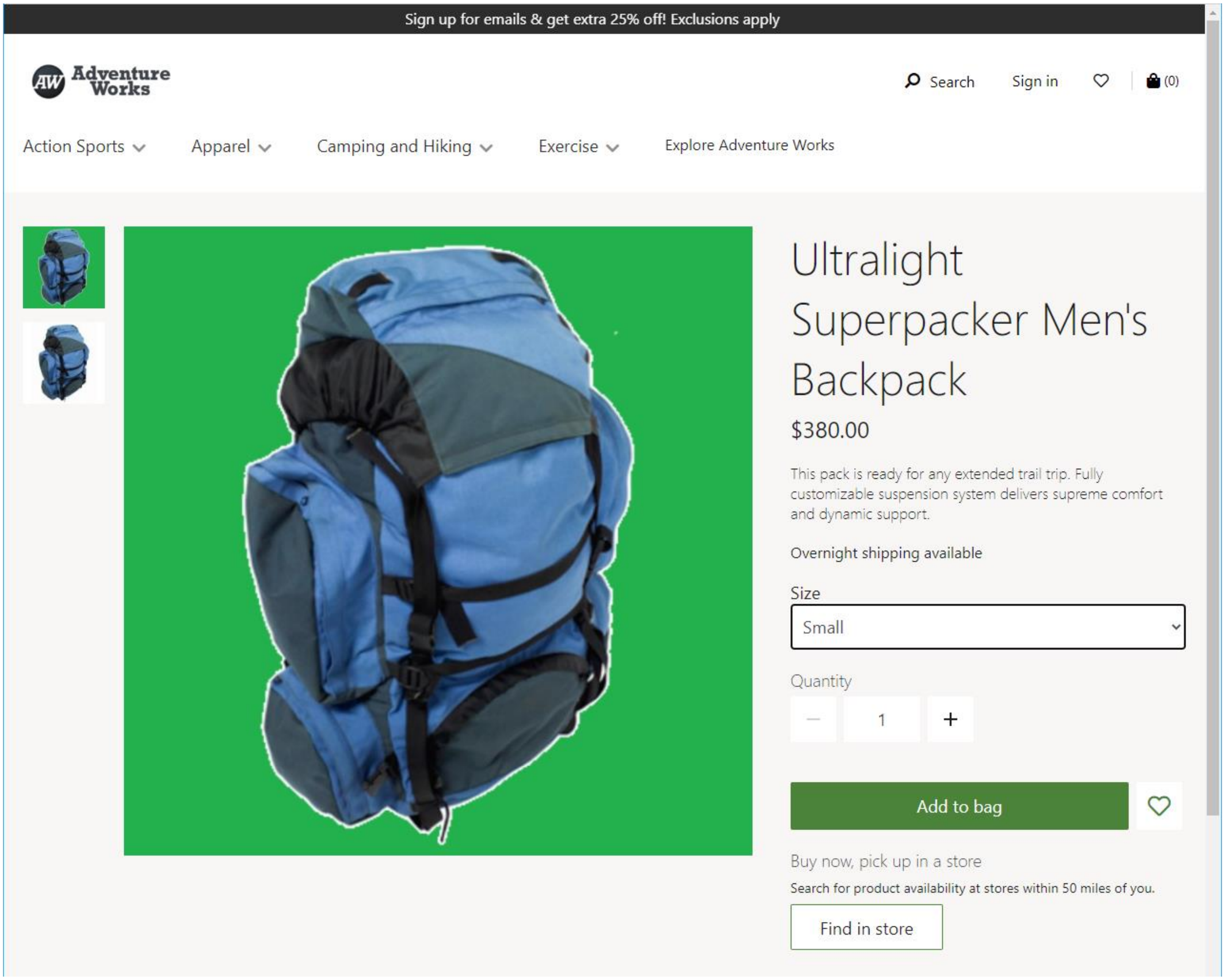
If you now navigate to the category and product details page you should see the images load up and changing the size variants should load the appropriate image.




Sign up for emails & get extra 25% off! Exclusions apply
Adventure Works
Search
Sign in
(0)
Action Sports
Apparel
Camping and Hiking
Exercise
Explore Adventure Works
Ultralight Superpacker Men's Backpack
$380.00
This pack is ready for any extended trail trip. Fully customizable suspension system delivers supreme comfort and dynamic support.
Overnight shipping available
Size
Choose a size
Quantity
1
Add to bag
Buy now, pick up in a store
Search for product availability at stores within 50 miles of you.
Find in store
Product specifications

Usage	Mountaineering
Frame type	Frameless
Search keywords	Default

Selecting a variant (Size) should show the specific variant image:



Ultralight Superpacker Men's Backpack

$380.00

This pack is ready for any extended trail trip. Fully customizable suspension system delivers supreme comfort and dynamic support.

Overnight shipping available

Size: Small

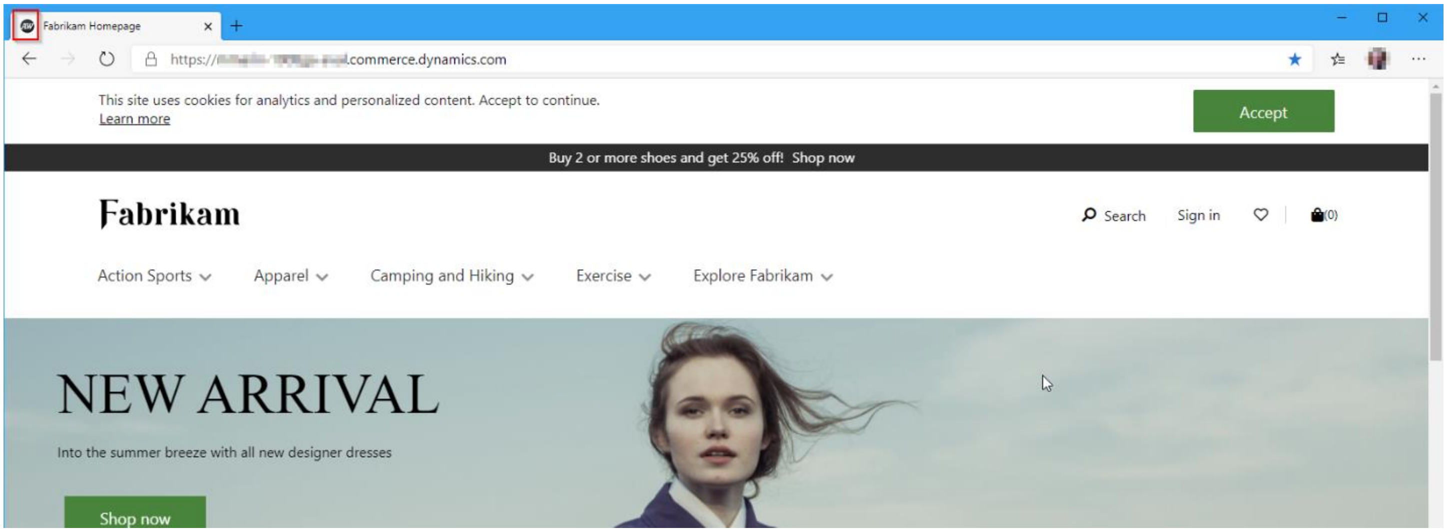
Quantity: 1

Buy now, pick up in a store

Search for product availability at stores within 50 miles of you.

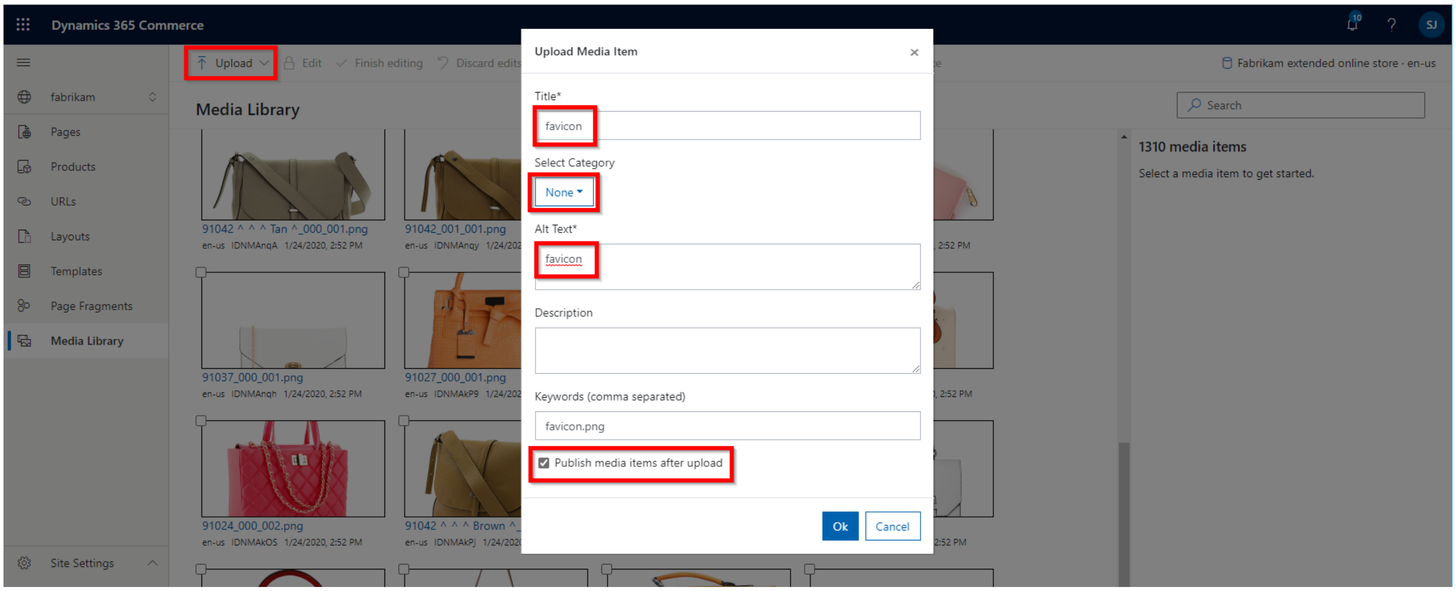
ADDING A FAVICON

A favicon is the icon that shows up on the browser tab as highlighted in the below image. This image is specified in the HTML metatags. The below steps will show how to upload an image and add it to the metatags section of the html pages.


Fabrikam Homepage
https://…commerce.dynamics.com
This site uses cookies for analytics and personalized content. Accept to continue.
Learn more
Accept
Buy 2 or more shoes and get 25% off! Shop now
Fabrikam
Search
Sign in
(0)
Action Sports
Apparel
Camping and Hiking
Exercise
Explore Fabrikam
NEW ARRIVAL
Into the summer breeze with all new designer dresses
Shop now

To add a favicon, first you need to upload the icon image to the site's Media Library and copy the public URL it provides to add to a meta tag link module.

If you are using the workshop asset zip file, upload the "favicon.png" file provided for this workshop. Leave the category as "None" and provide a name and alt text and ensure "**Publish media items after upload**" is enabled.



Upload Media Item

Title*: favicon

Select Category: None

Alt Text*: favicon

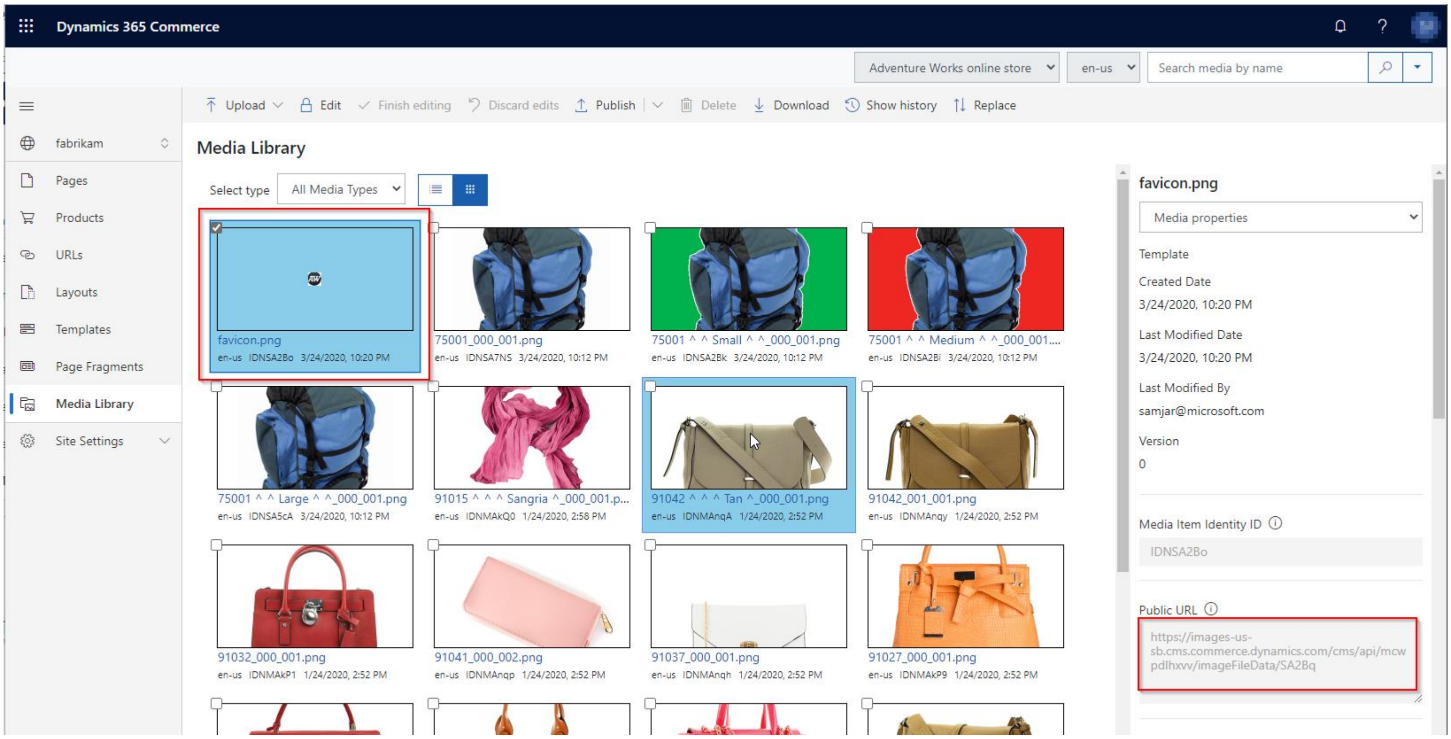
Description:

Keywords (comma separated): favicon.png

☑ Publish media items after upload

Ok Cancel

Copy the "Public URL" found in the right-hand column after you select the image.



Media Library

Select type: All Media Types

favicon.png
en-us IDNSA2Bo 3/24/2020, 10:20 PM

75001_000_001.png
en-us IDNSA7NS 3/24/2020, 10:12 PM

75001 ^ ^ Small ^ ^_000_001.png
en-us IDNSA2Bk 3/24/2020, 10:12 PM

75001 ^ ^ Medium ^ ^_000_001....
en-us IDNSA2Bi 3/24/2020, 10:12 PM

75001 ^ ^ Large ^ ^_000_001.png
en-us IDNSA5cA 3/24/2020, 10:12 PM

91015 ^ ^ ^ Sangria ^_000_001.p...
en-us IDNMAkQ0 1/24/2020, 2:58 PM

91042 ^ ^ ^ Tan ^_000_001.png
en-us IDNMAnqA 1/24/2020, 2:52 PM

91042_001_001.png
en-us IDNMAnqy 1/24/2020, 2:52 PM

91032_000_001.png
en-us IDNMAkP1 1/24/2020, 2:52 PM

91041_000_002.png
en-us IDNMAnqp 1/24/2020, 2:52 PM

91037_000_001.png
en-us IDNMAnqh 1/24/2020, 2:52 PM

91027_000_001.png
en-us IDNMAkP9 1/24/2020, 2:52 PM

favicon.png

Media properties

Template

Created Date
3/24/2020, 10:20 PM

Last Modified Date
3/24/2020, 10:20 PM

Last Modified By
samjar@microsoft.com

Version
0

Media Item Identity ID
IDNSA2Bo

Public URL
https://images-us-sb.cms.commerce.dynamics.com/cms/api/mcwpdlhxvv/imageFileData/SA2Bq

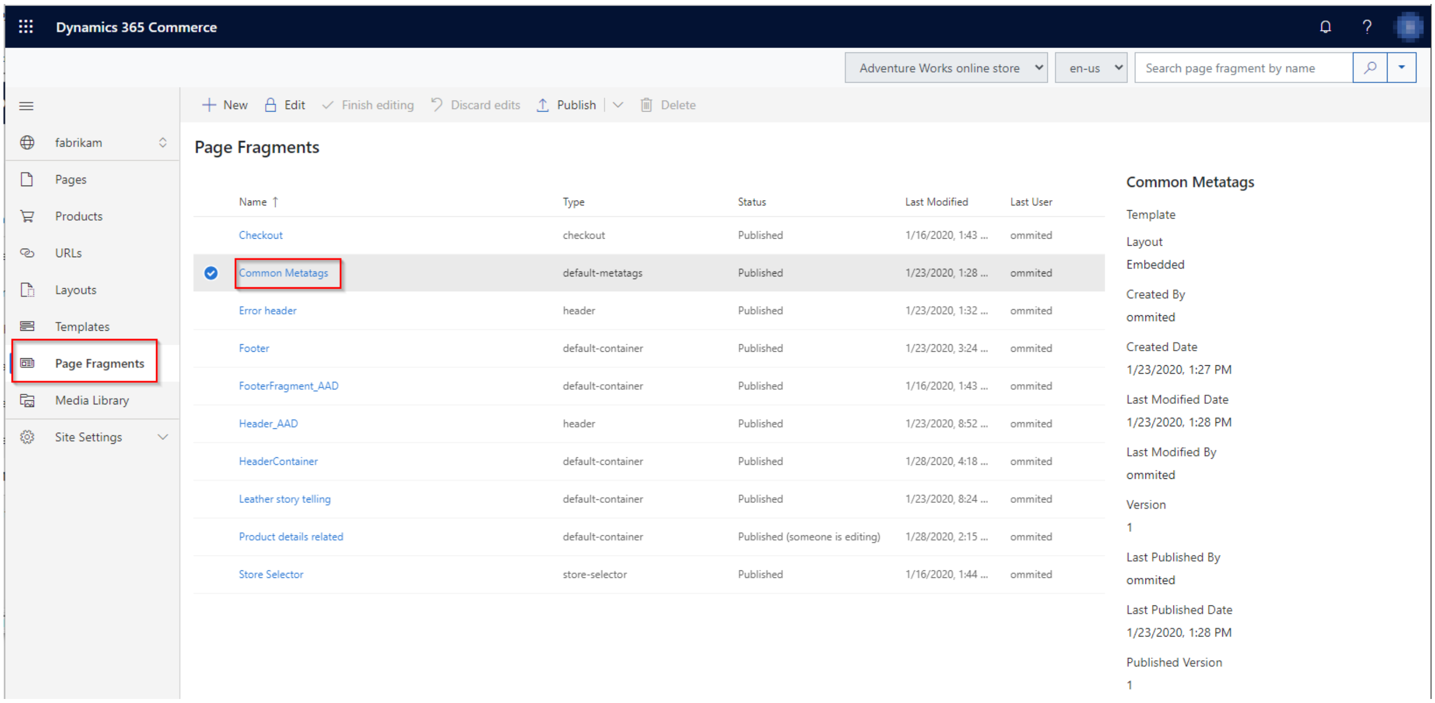
Then create link with the following format:

```
<link rel="shortcut icon" href="Public_URL_for_your_favicon">
```

For Example:

```
<link rel="shortcut icon" href="https://images-us-
sb.cms.commerce.dynamics.com/cms/api/kbkzpdnblk/imageFileData/SA5cI">
```

Next, you can find a **Page Fragment** called “Common Metatags” module as shown below.



Page Fragments

Name ↑	Type	Status	Last Modified	Last User
Checkout	checkout	Published	1/16/2020, 1:43 ...	ommited
Common Metatags	default-metatags	Published	1/23/2020, 1:28 ...	ommited
Error header	header	Published	1/23/2020, 1:32 ...	ommited
Footer	default-container	Published	1/23/2020, 3:24 ...	ommited
FooterFragment_AAD	default-container	Published	1/16/2020, 1:43 ...	ommited
Header_AAD	header	Published	1/23/2020, 8:52 ...	ommited
HeaderContainer	default-container	Published	1/28/2020, 4:18 ...	ommited
Leather story telling	default-container	Published	1/23/2020, 8:24 ...	ommited
Product details related	default-container	Published (someone is editing)	1/28/2020, 2:15 ...	ommited
Store Selector	store-selector	Published	1/16/2020, 1:44 ...	ommited

Common Metatags

Template

Layout
Embedded

Created By
ommited

Created Date
1/23/2020, 1:27 PM

Last Modified Date
1/23/2020, 1:28 PM

Last Modified By
ommited

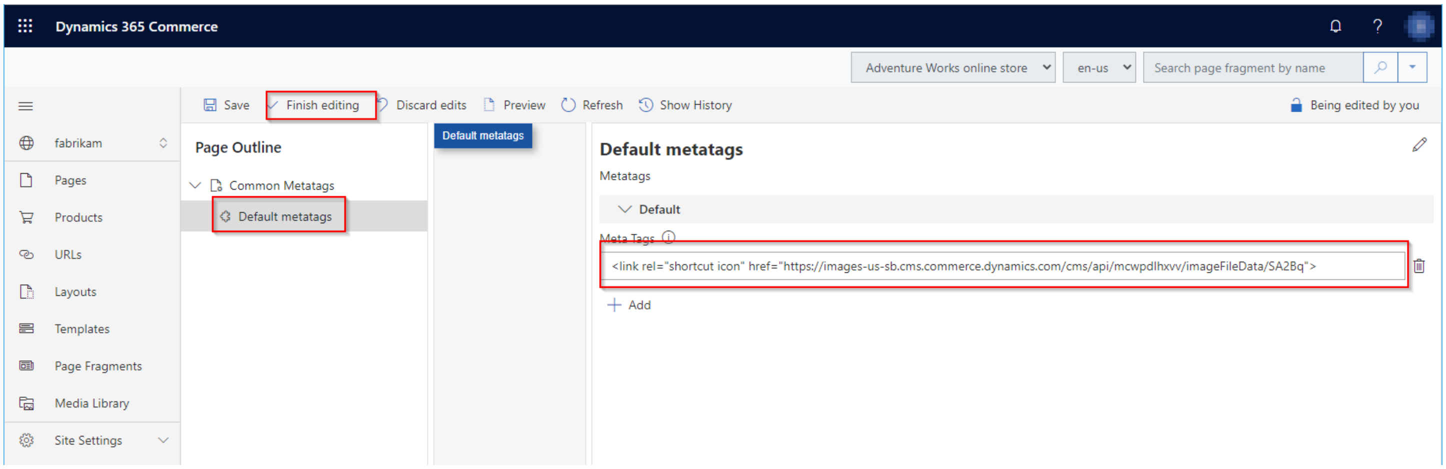
Version
1

Last Published By
ommited

Last Published Date
1/23/2020, 1:28 PM

Published Version
1

Select the “**Edit**” button the select the “**Default metatags**” module found under the “**Common Metatags**” node. In the right-hand column change the “**Meta Tags**” section to include the new link created above. Select “**Finish editing**” followed by “**Publish**”.



Default metatags

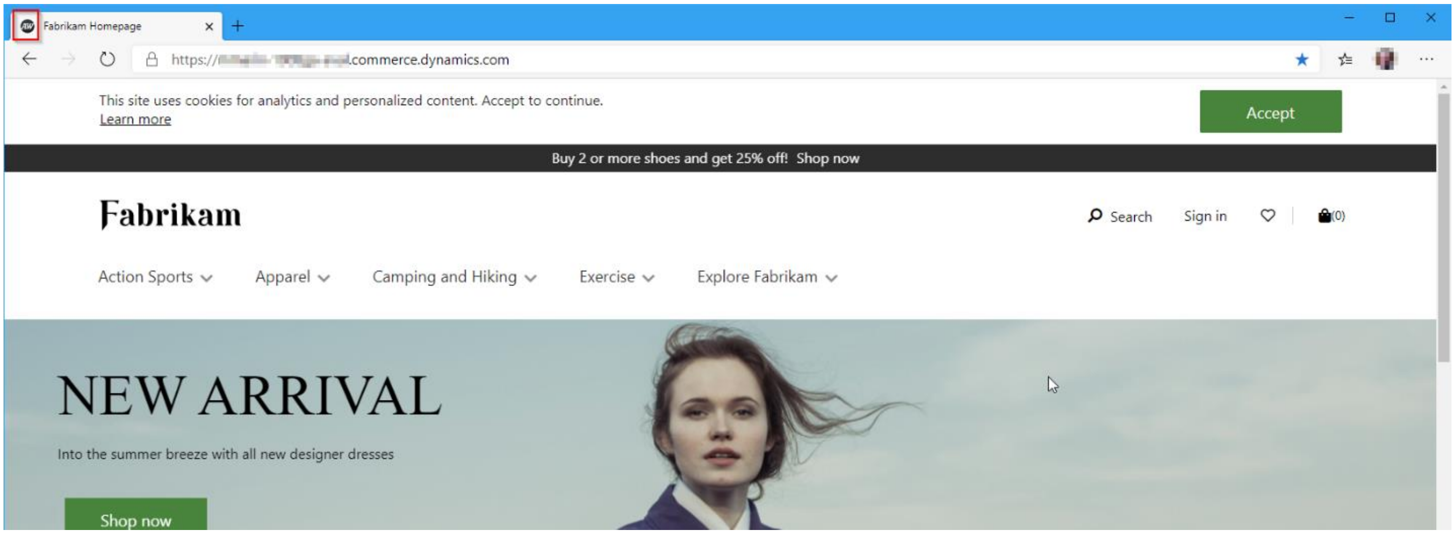
Metatags

Default

Meta Tags

```
<link rel="shortcut icon" href="https://images-us-sb.cms.commerce.dynamics.com/cms/api/mcwpdlhxvv/imageFileData/SA2Bq">
```

If you now refresh the e-Commerce site, you should see the favicon in the Browser tab as shown below.


Fabrikam Homepage
https://[redacted].commerce.dynamics.com
This site uses cookies for analytics and personalized content. Accept to continue.
Learn more
Accept
Buy 2 or more shoes and get 25% off! Shop now
Fabrikam
Search
Sign in
(0)
Action Sports
Apparel
Camping and Hiking
Exercise
Explore Fabrikam
NEW ARRIVAL
Into the summer breeze with all new designer dresses
Shop now

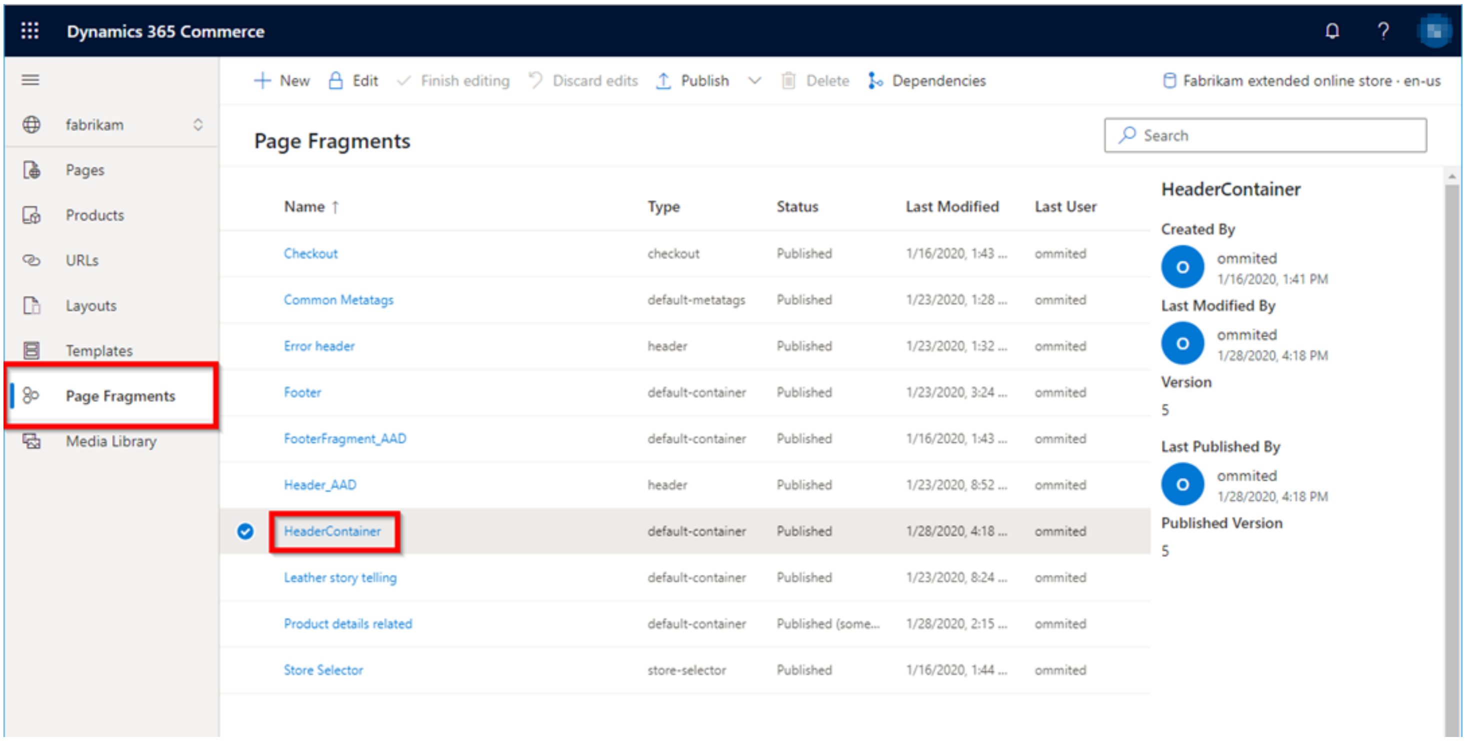
CONFIGURE HEADER, FOOTER & HOMEPAGE CONTENT

We will now change the logo in the header fragment as well as change the footer to match our new company.

Fragments are standalone sets of modules that can be configured once and placed on multiple pages. The header and footer are created as fragments which allow it to be configured once and every page that uses them will pick up and changes published.

HEADER

Select the “**Page Fragments**” tab and then the “**HeaderContainer**” fragment.



Page Fragments

Name ↑	Type	Status	Last Modified	Last User
Checkout	checkout	Published	1/16/2020, 1:43 ...	ommited
Common Metatags	default-metatags	Published	1/23/2020, 1:28 ...	ommited
Error header	header	Published	1/23/2020, 1:32 ...	ommited
Footer	default-container	Published	1/23/2020, 3:24 ...	ommited
FooterFragment_AAD	default-container	Published	1/16/2020, 1:43 ...	ommited
Header_AAD	header	Published	1/23/2020, 8:52 ...	ommited
HeaderContainer	default-container	Published	1/28/2020, 4:18 ...	ommited
Leather story telling	default-container	Published	1/23/2020, 8:24 ...	ommited
Product details related	default-container	Published (some...	1/28/2020, 2:15 ...	ommited
Store Selector	store-selector	Published	1/16/2020, 1:44 ...	ommited

HeaderContainer

Created By
ommited
1/16/2020, 1:41 PM

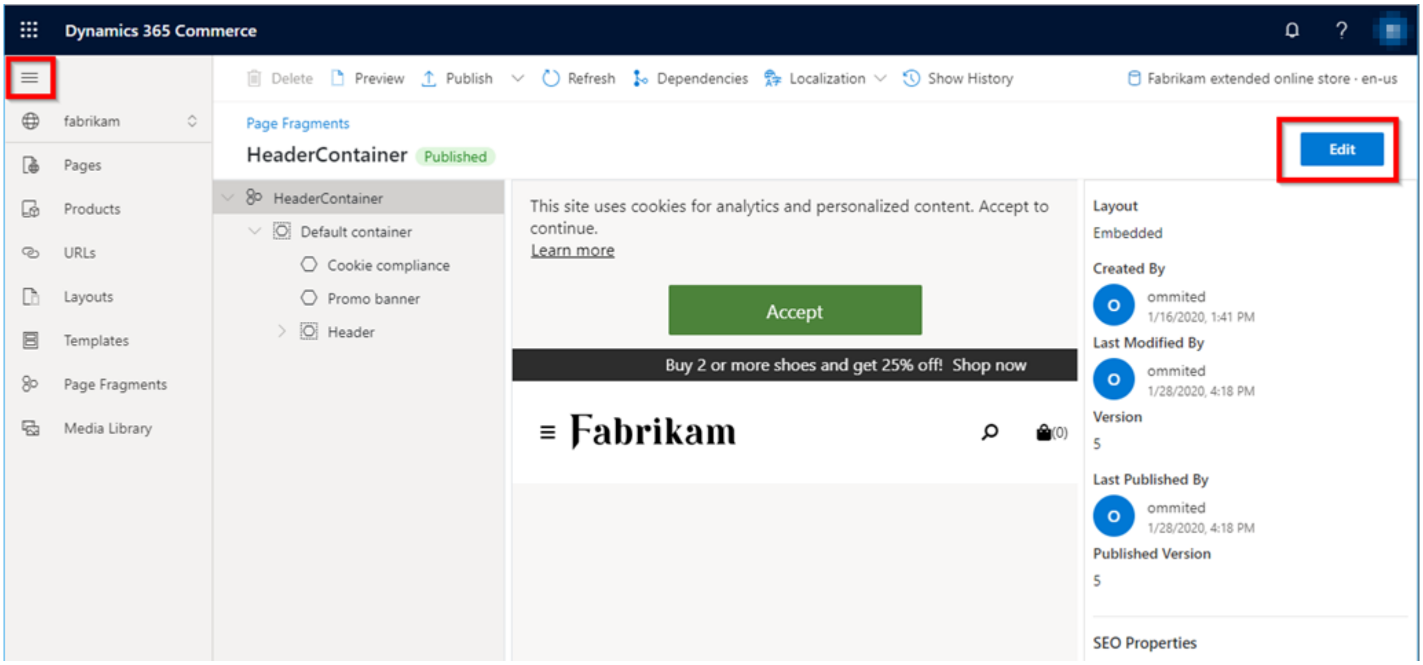
Last Modified By
ommited
1/28/2020, 4:18 PM

Version
5

Last Published By
ommited
1/28/2020, 4:18 PM

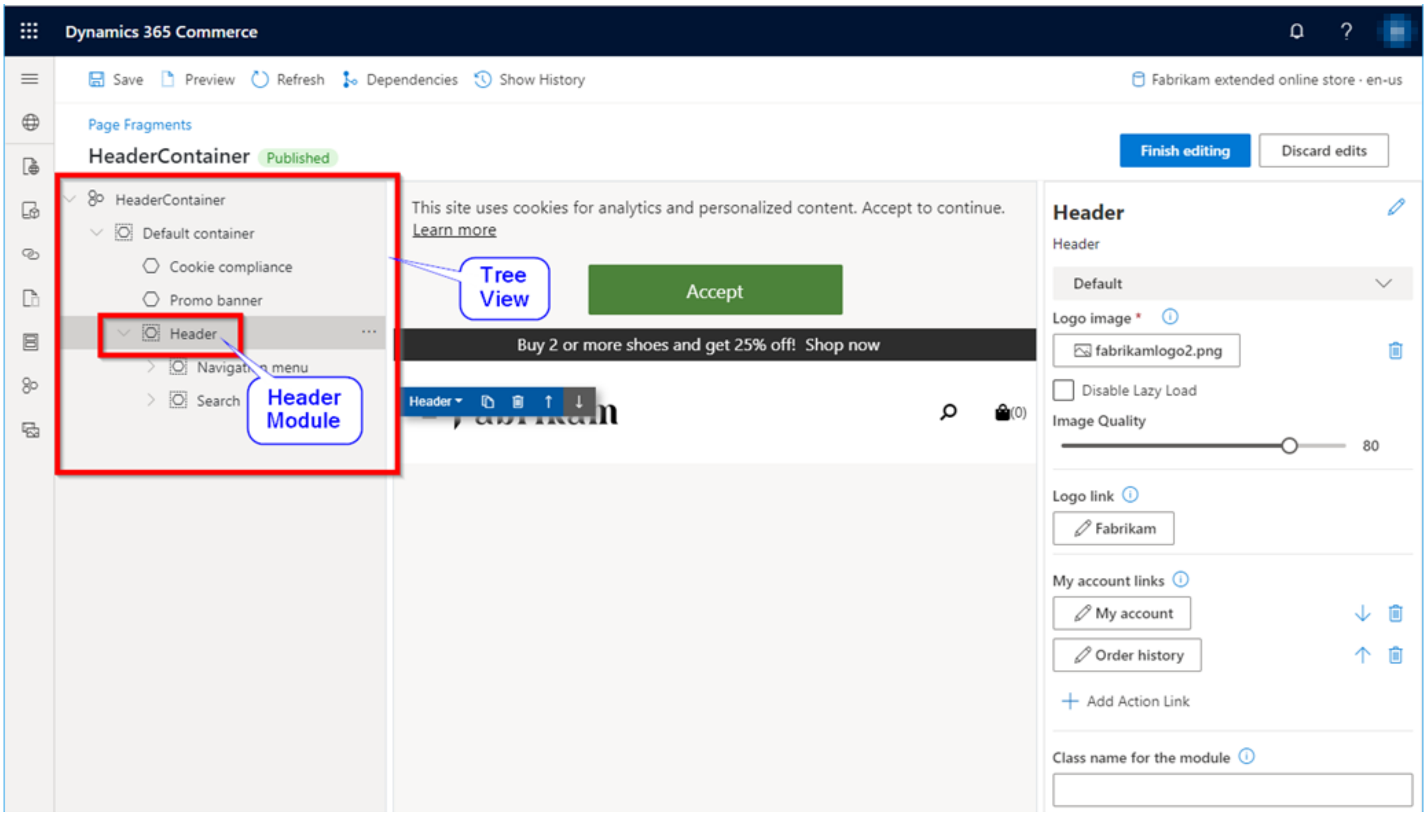
Published Version
5

Once inside the fragment editor, select the “**Edit**” button. You can also hit the hamburger menu on the top left to collapse the tabs to give more room to work on the fragment.



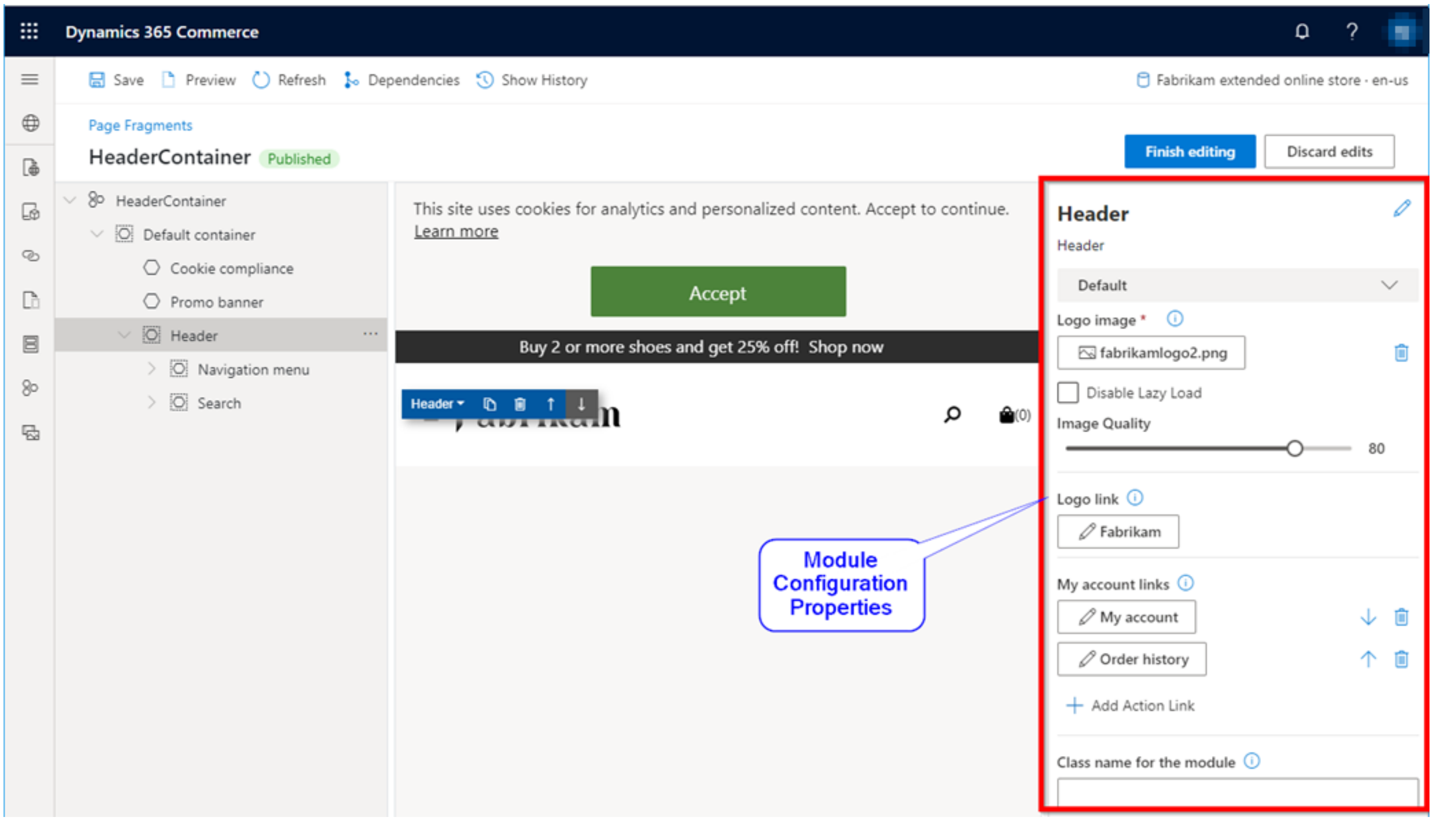
Layout
Embedded
Created By
ommited
1/16/2020, 1:41 PM
Last Modified By
ommited
1/28/2020, 4:18 PM
Version
5
Last Published By
ommited
1/28/2020, 4:18 PM
Published Version
5

Next, inside of the tree view of the fragment (the left column highlighted below), select the “**Header**” module.



Default
Logo image * fabrikamlogo2.png
☐ Disable Lazy Load
Image Quality 80
Logo link: Fabrikam
My account links: My account, Order history
Class name for the module

You will see a list of module configurations on the right-hand column of the page.



HeaderContainer Published

This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Buy 2 or more shoes and get 25% off! Shop now

Header

Header

Default

Logo image * fabrikamlogo2.png

□ Disable Lazy Load

Image Quality 80

Logo link Fabrikam

My account links

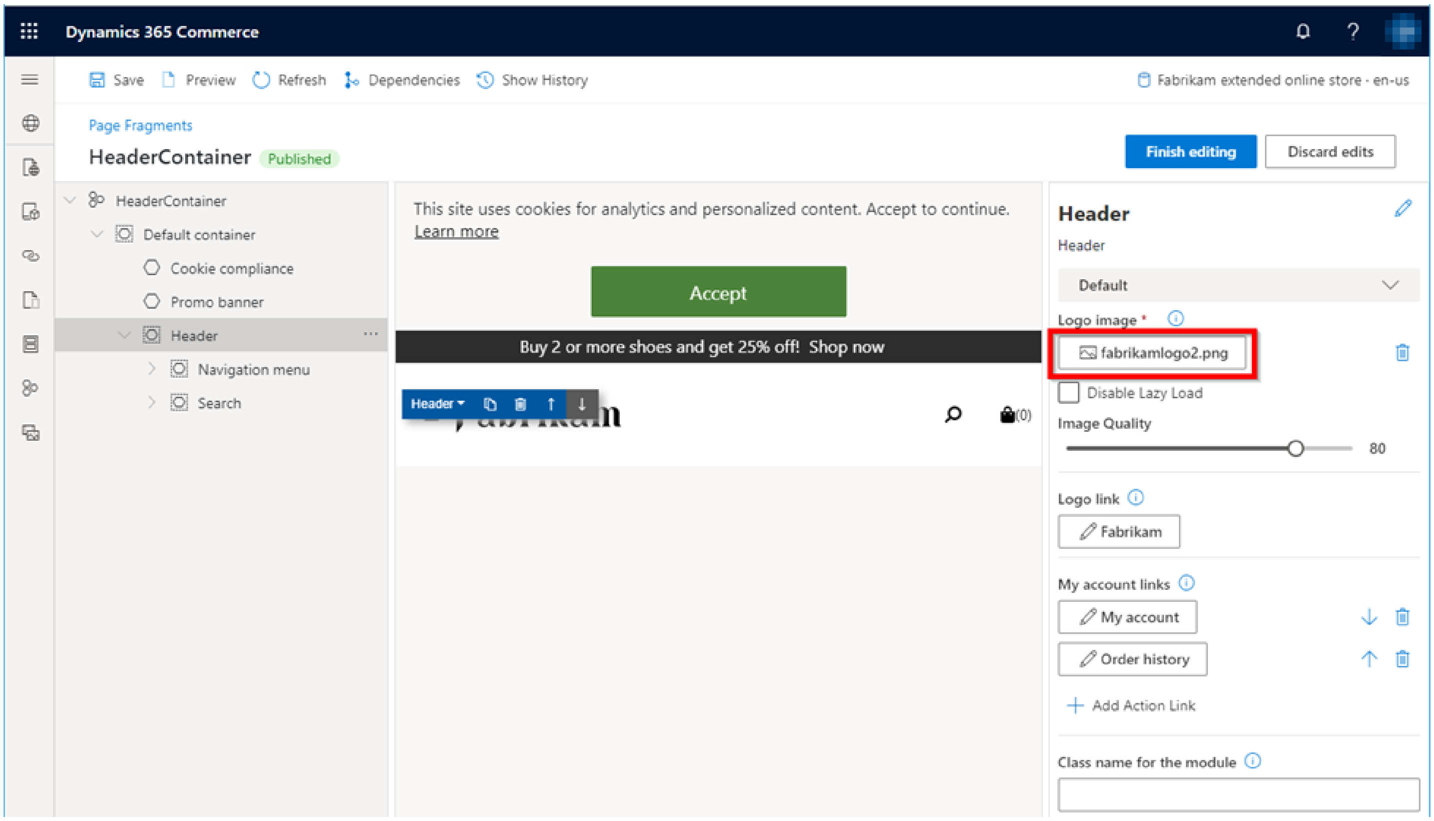
My account

Order history

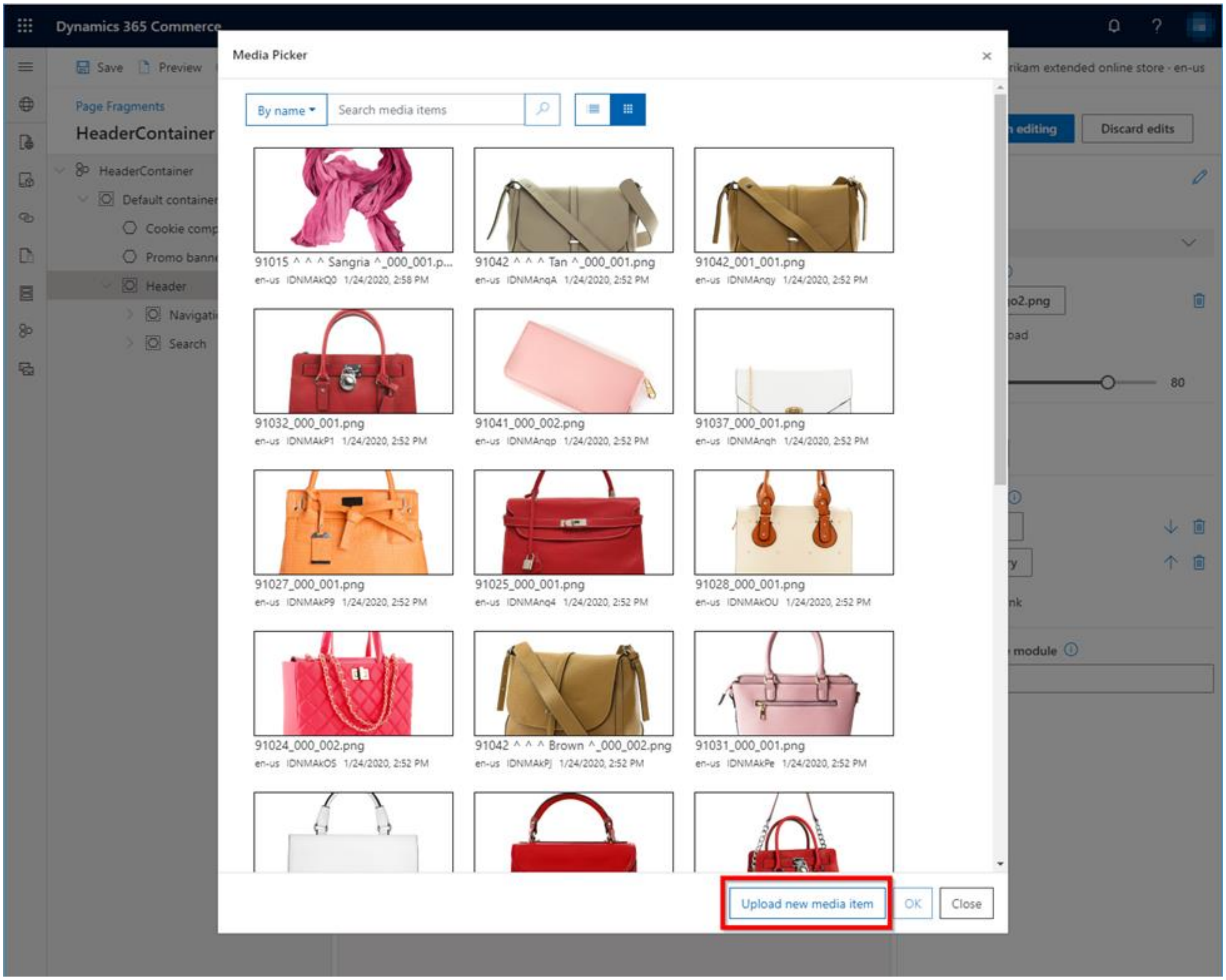
Class name for the module

Module Configuration Properties

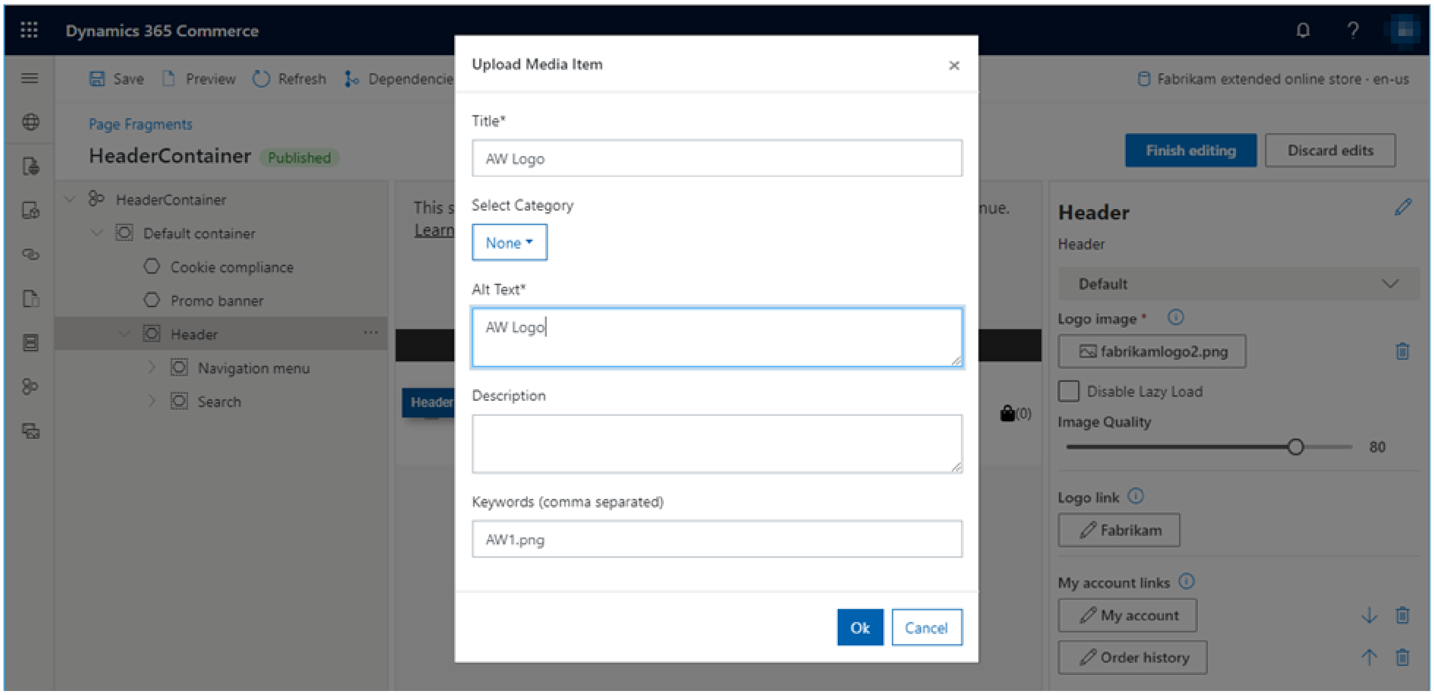
Select the config property "**Logo image**" file name (should say "fabrikamlogo2.png").



The Media Picker will appear as shown below, you will see other images already uploaded as part of the Fabrikam fashion marketing and product data. Click on the “Upload new media item” button.


Dynamics 365 Commerce
Media Picker
By name
Search media items
91015 ^ ^ ^ Sangria ^_000_001.p...
en-us IDNMAkQ0 1/24/2020, 2:58 PM
91042 ^ ^ ^ Tan ^_000_001.png
en-us IDNMAnqA 1/24/2020, 2:52 PM
91042_001_001.png
en-us IDNMAnqy 1/24/2020, 2:52 PM
91032_000_001.png
en-us IDNMAkP1 1/24/2020, 2:52 PM
91041_000_002.png
en-us IDNMAnqp 1/24/2020, 2:52 PM
91037_000_001.png
en-us IDNMAnqh 1/24/2020, 2:52 PM
91027_000_001.png
en-us IDNMAkP9 1/24/2020, 2:52 PM
91025_000_001.png
en-us IDNMAnq4 1/24/2020, 2:52 PM
91028_000_001.png
en-us IDNMAkOU 1/24/2020, 2:52 PM
91024_000_002.png
en-us IDNMAkOS 1/24/2020, 2:52 PM
91042 ^ ^ ^ Brown ^_000_002.png
en-us IDNMAkPJ 1/24/2020, 2:52 PM
91031_000_001.png
en-us IDNMAkPe 1/24/2020, 2:52 PM
Upload new media item
OK
Close
Save
Preview
Page Fragments
HeaderContainer
HeaderContainer
Default container
Cookie comp
Promo bann
Header
Navigati
Search
rikam extended online store - en-us
editing
Discard edits
o2.png
oad
80
nk
module

Upload the new logo image (a sample is provided is asset zip file called AW1.png which can be used), select it and click the “**OK**” button. You will now need to provide an image title and alt text such as “AW logo” and leave category set to “None”).


Upload Media Item

Title*: AW Logo

Select Category: None

Alt Text*: AW Logo

Description:

Keywords (comma separated): AW1.png

Logo image *: fabrikamlogo2.png

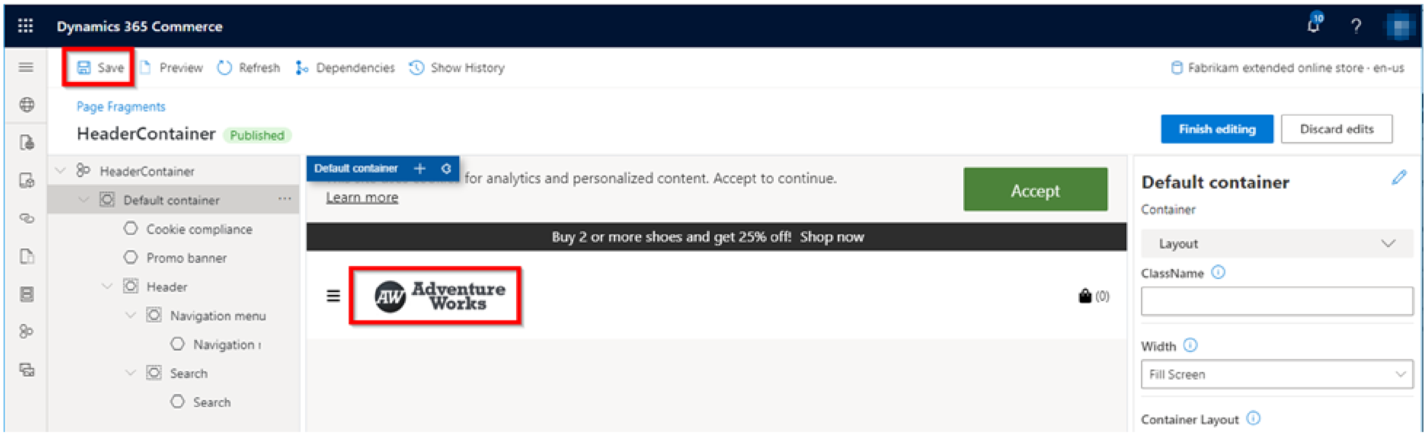
□ Disable Lazy Load

Image Quality: 80

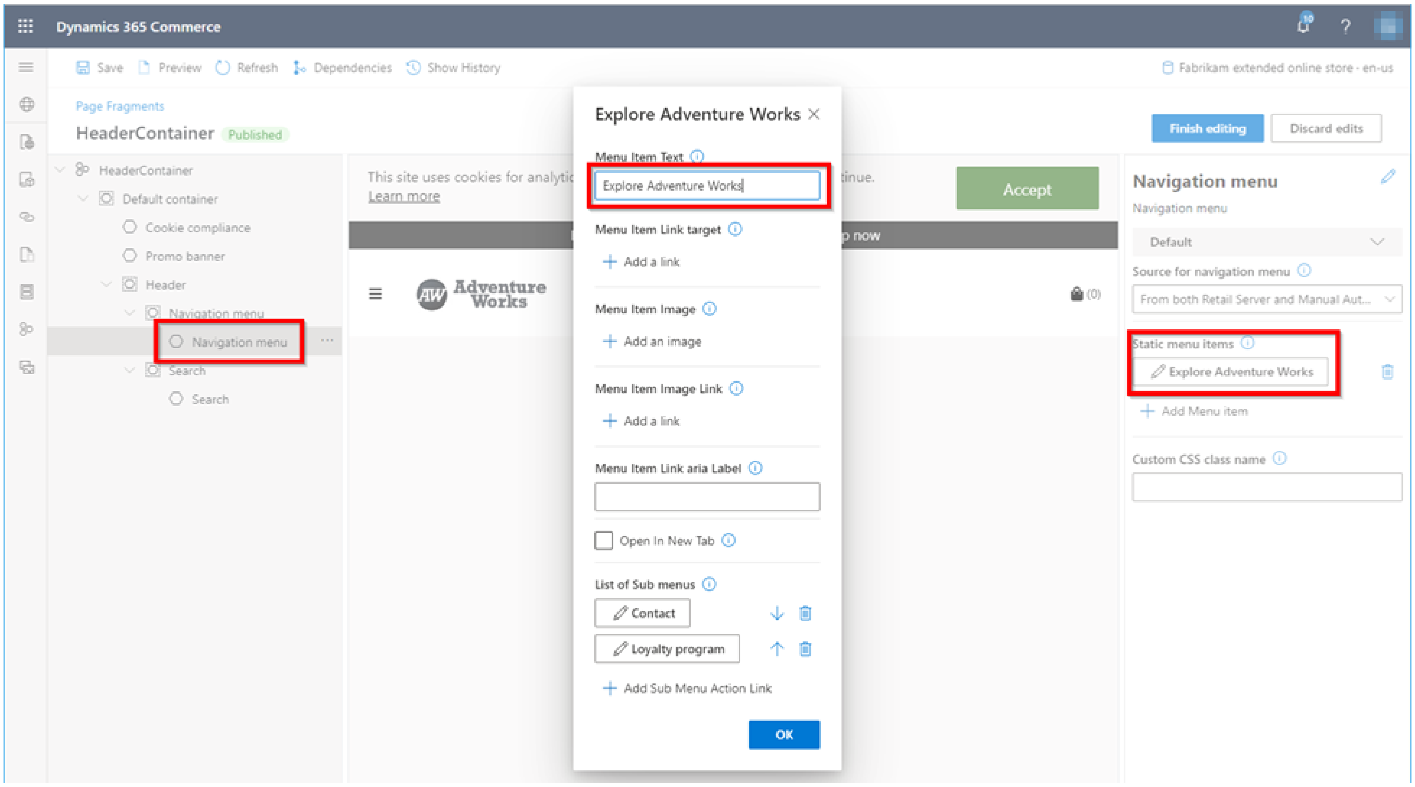
Logo link: Fabrikam

My account links: My account, Order history

You need to click “**Save**” to see the updated image in the preview section as shown below.



Next, we will change the custom navigation menu item that appears next to the navigation menu. Select the “**Navigation menu**” module and change the “Explore Fabrikam” to a better name for your site such as “Explore Adventure Works”.



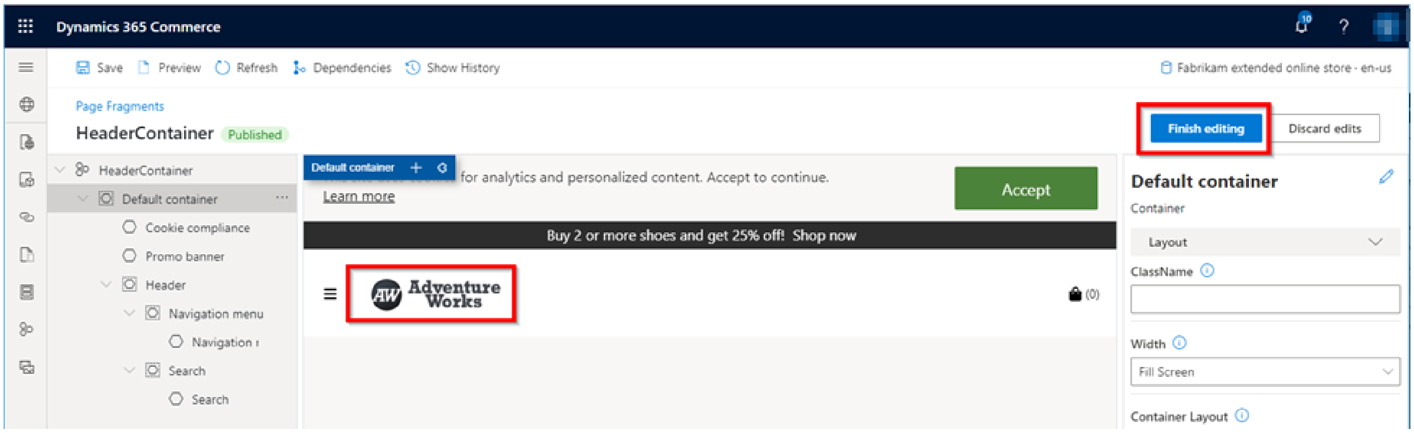
Menu Item Text: Explore Adventure Works

Default

Source for navigation menu: From both Retail Server and Manual Aut...

Static menu items: Explore Adventure Works

You can then click “**Finish editing**” and “**Publish**” the header fragment when complete.



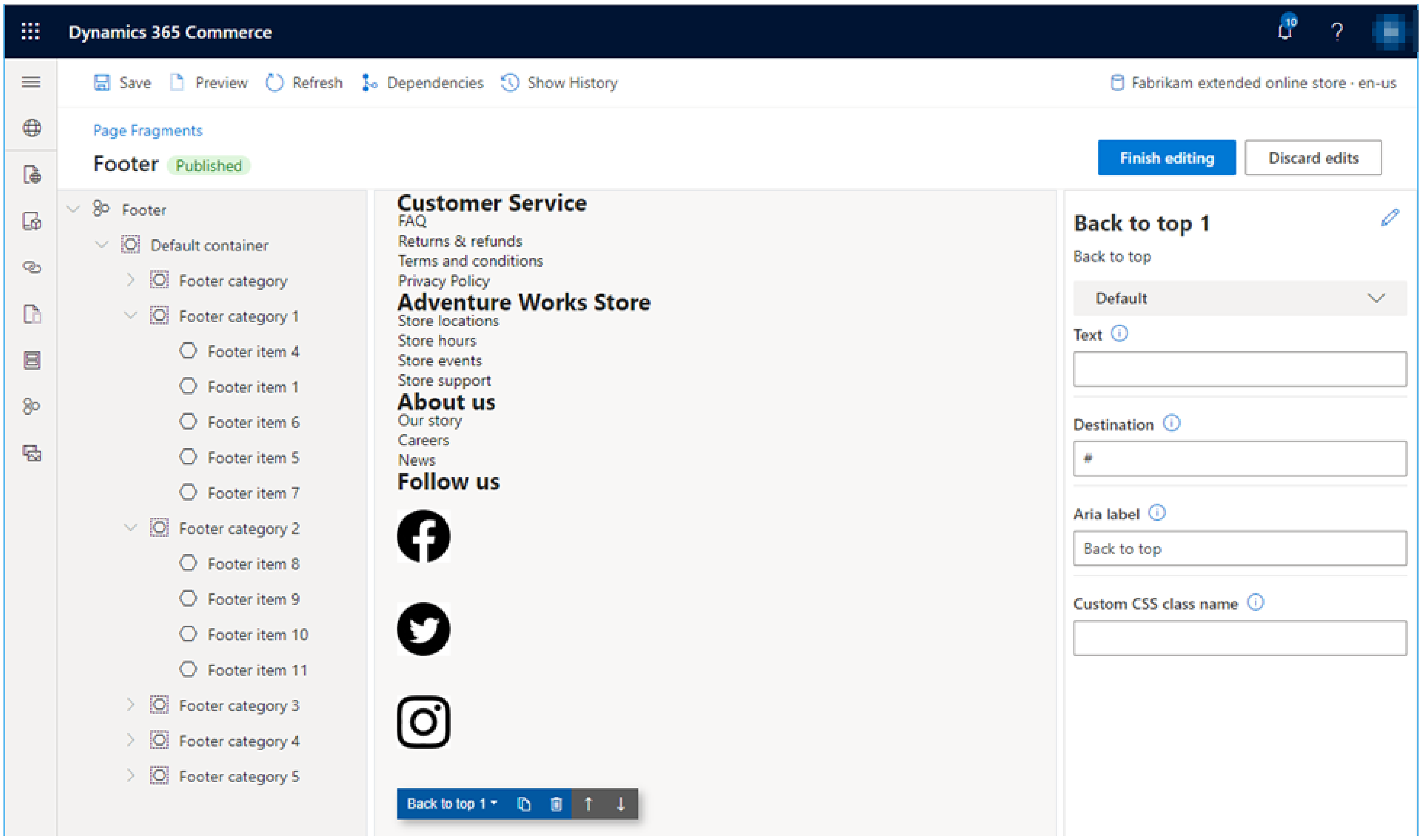
Layout

Width: Fill Screen

FOOTER

In this section we will be modifying the footer fragment to remove references to “Fabrikam”.

Select the “**Footer**” fragment and then the “**Edit**” button. Select the various modules in the tree nodes (or select the text directly in the preview section in the middle) and change any reference to “Fabrikam” to your new company name as shown below. When complete select “Finish editing” and publish the fragment.



Customer Service
FAQ
Returns & refunds
Terms and conditions
Privacy Policy
Adventure Works Store
Store locations
Store hours
Store events
Store support
About us
Our story
Careers
News
Follow us

Back to top 1
Back to top
Default
Text:
Destination: #
Aria label: Back to top
Custom CSS class name:

If you now preview the site, it should now have the new logo in the header and footer changes on all pages. Remember you can get to the homepage of your site through either the LCS tool or by going to the homepage page in site builder and selecting “Preview”.

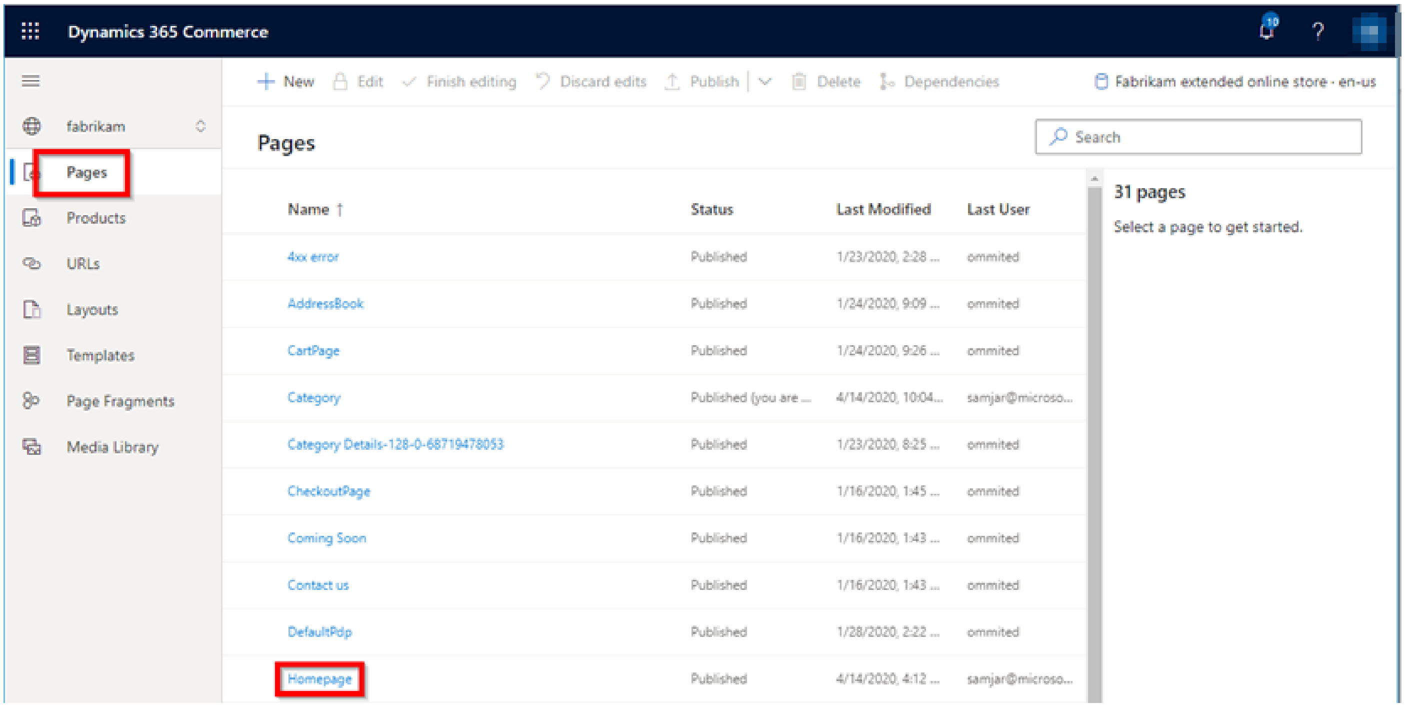
HOMEPAGE MARKETING CONTENT

We will now change the Homepage marketing content to match the new brand. You can change however you like, below guide shows how to change to an “Adventure Works” sporting goods branded store.

We are going to do the following:

- Configure 2 Content Block modules inside the carousel module to have new images and delete the third (sample images provided in asset zip file). Configure the CTA “Shop Now” links to point to specific category pages.
- Configure text block modules to change the text to more appropriate content
- Remove the video module
- Change the category images and links in the lower set of content blocks.

Open the Homepage in the site builder tool under the “**Pages**” tab.



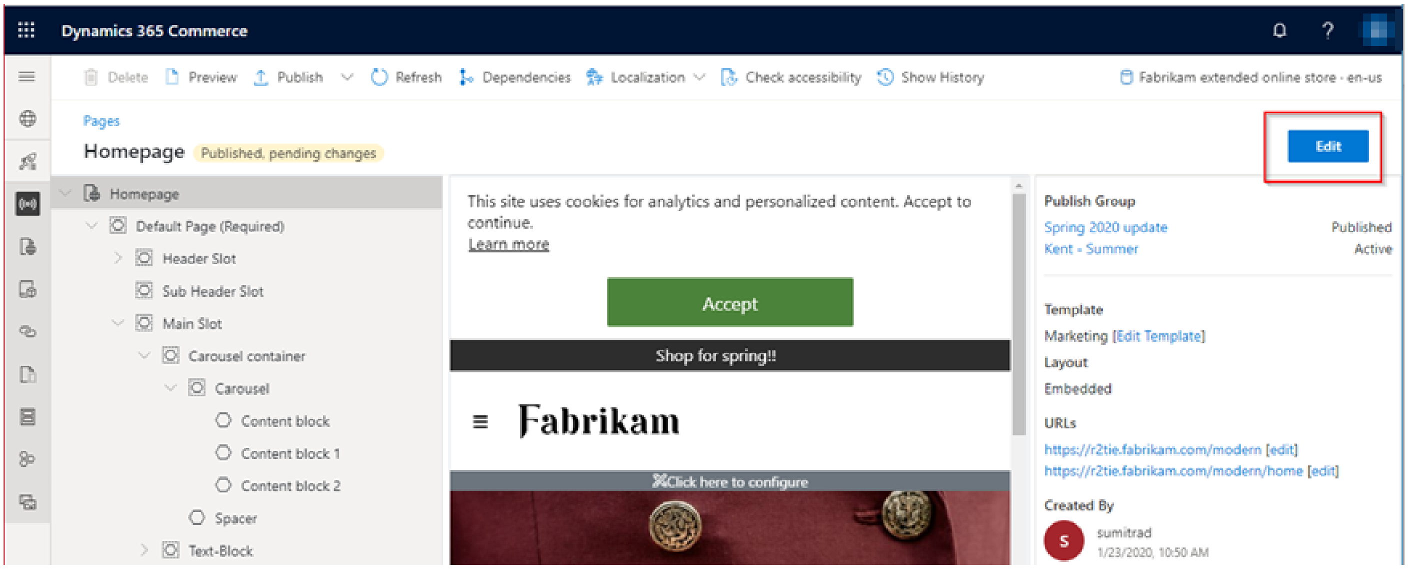
Pages

Name ↑	Status	Last Modified	Last User
4xx error	Published	1/23/2020, 2:28 ...	ommited
AddressBook	Published	1/24/2020, 9:09 ...	ommited
CartPage	Published	1/24/2020, 9:26 ...	ommited
Category	Published (you are ...	4/14/2020, 10:04...	samjar@microso...
Category Details-128-0-68719478053	Published	1/23/2020, 8:25 ...	ommited
CheckoutPage	Published	1/16/2020, 1:45 ...	ommited
Coming Soon	Published	1/16/2020, 1:43 ...	ommited
Contact us	Published	1/16/2020, 1:43 ...	ommited
DefaultPdp	Published	1/28/2020, 2:22 ...	ommited
Homepage	Published	4/14/2020, 4:12 ...	samjar@microso...

31 pages

Select a page to get started.

Select “**Edit**”.



Homepage Published, pending changes

This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Publish Group

Spring 2020 update	Published
Kent - Summer	Active

Template
Marketing [Edit Template]

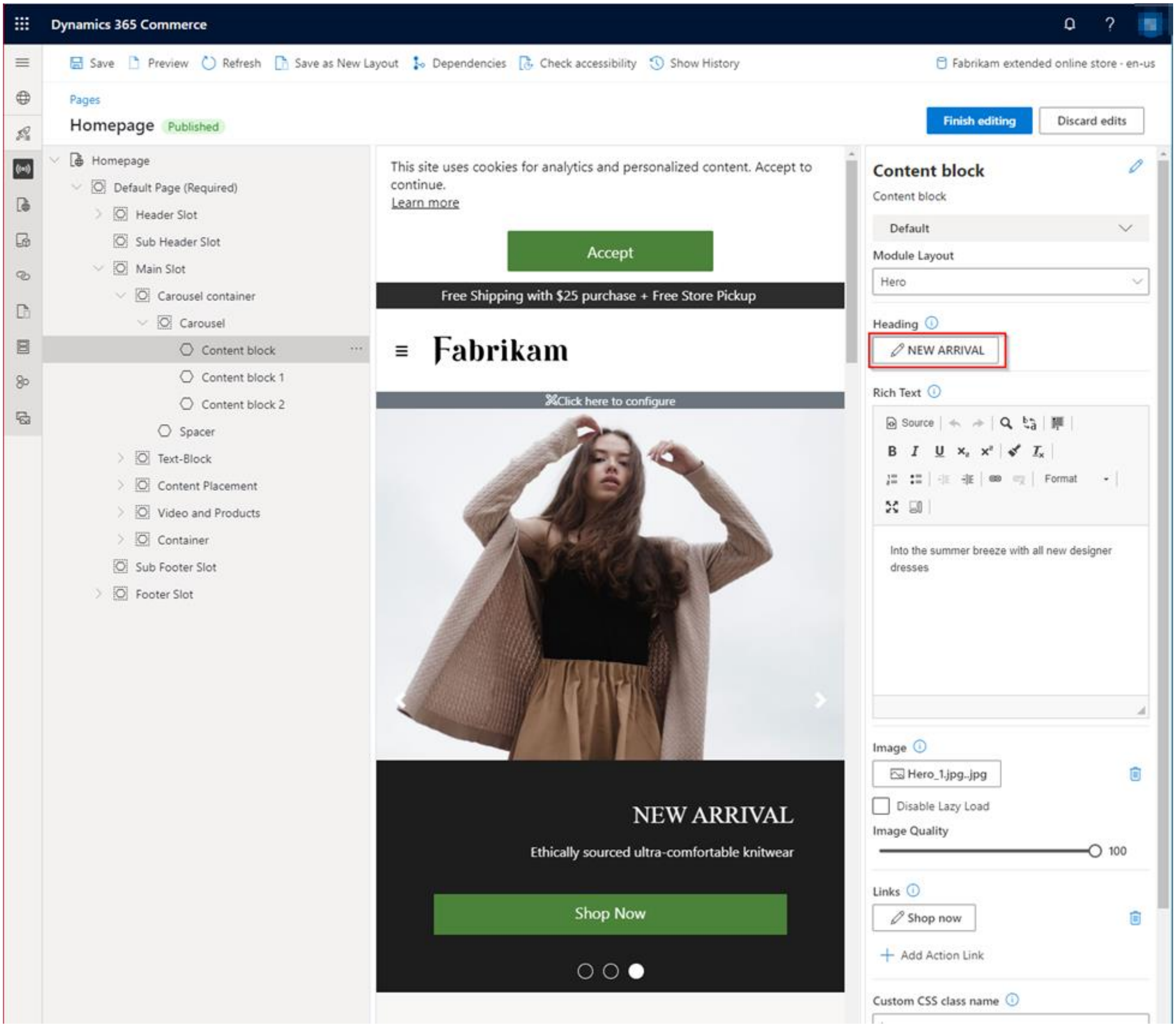
Layout
Embedded

URLs
https://r2tie.fabrikam.com/modern [edit]
https://r2tie.fabrikam.com/modern/home [edit]

Created By
S sumitrad
1/23/2020, 10:50 AM

CONFIGURE CAROUSEL MODULE

In the “Page Outline” section select the first “Content block” under the “Homepage” -> “Default Page” -> “Main Slot” -> “Carousel container” -> “Carousel” module as shown below.



Content block

Content block

Default

Module Layout: Hero

Heading: NEW ARRIVAL

Rich Text: Into the summer breeze with all new designer dresses

Image: Hero_1.jpg..jpg

☐ Disable Lazy Load

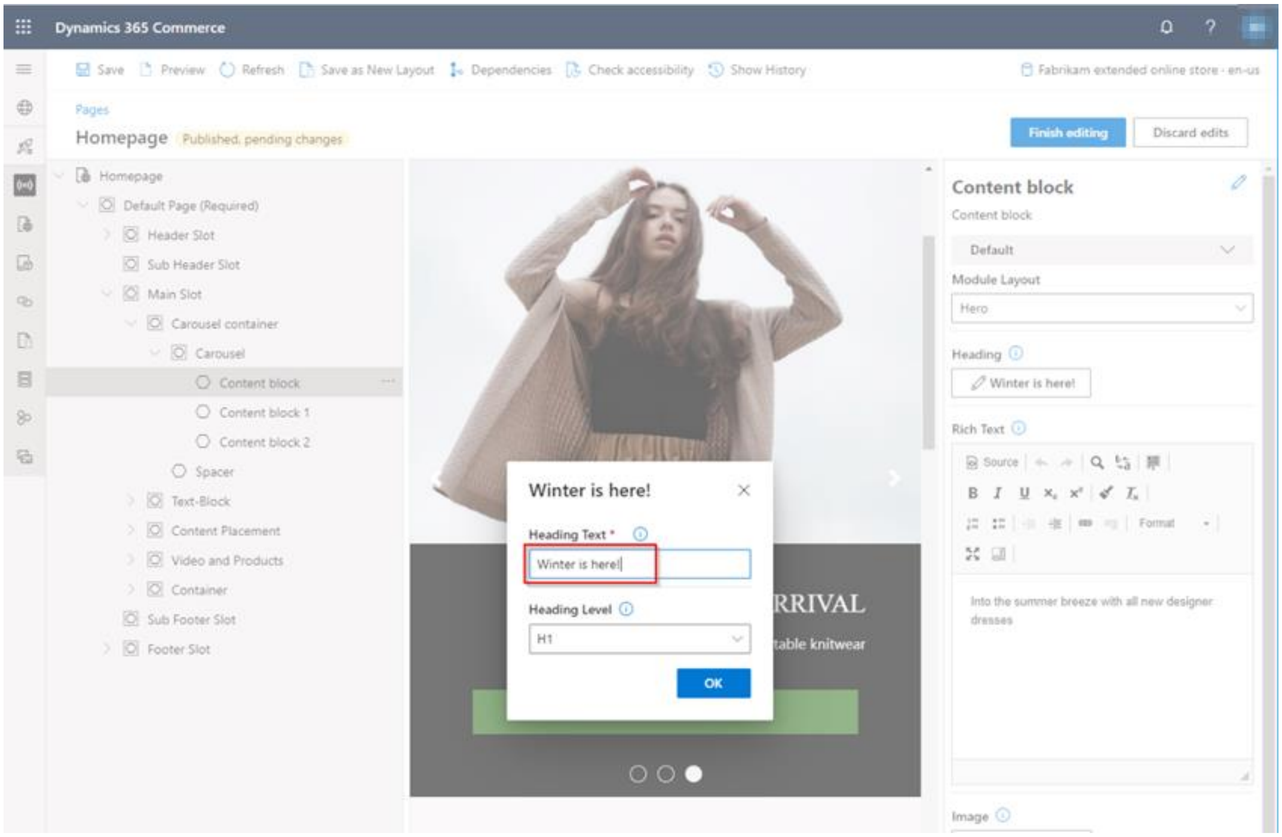
Image Quality: 100

Links: Shop now

Custom CSS class name

Select the “Heading” configuration property where it says “NEW ARRIVAL” as shown above.

Change the heading text as desired, see example below then select the “OK” button.



Winter is here!

Heading Text * : Winter is here!

Heading Level: H1

Content block

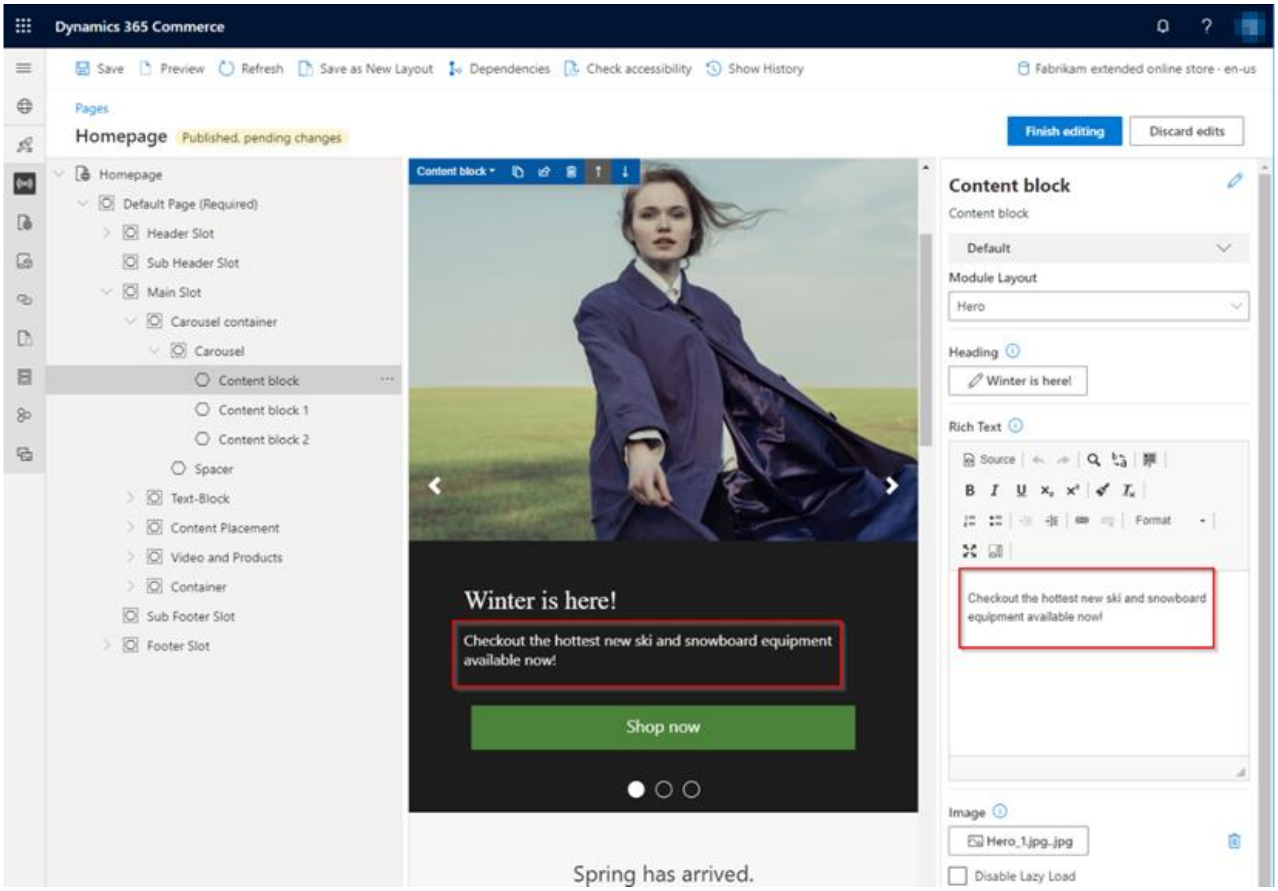
Content block: Default

Module Layout: Hero

Heading: Winter is here!

Rich Text: Into the summer breeze with all new designer dresses

Change the Rich Text to something appropriate, notice as you change the text you can see the text changes in the middle column rendered view as shown below.



Default

Module Layout: Hero

Heading: Winter is here!

Rich Text: Checkout the hottest new ski and snowboard equipment available now!

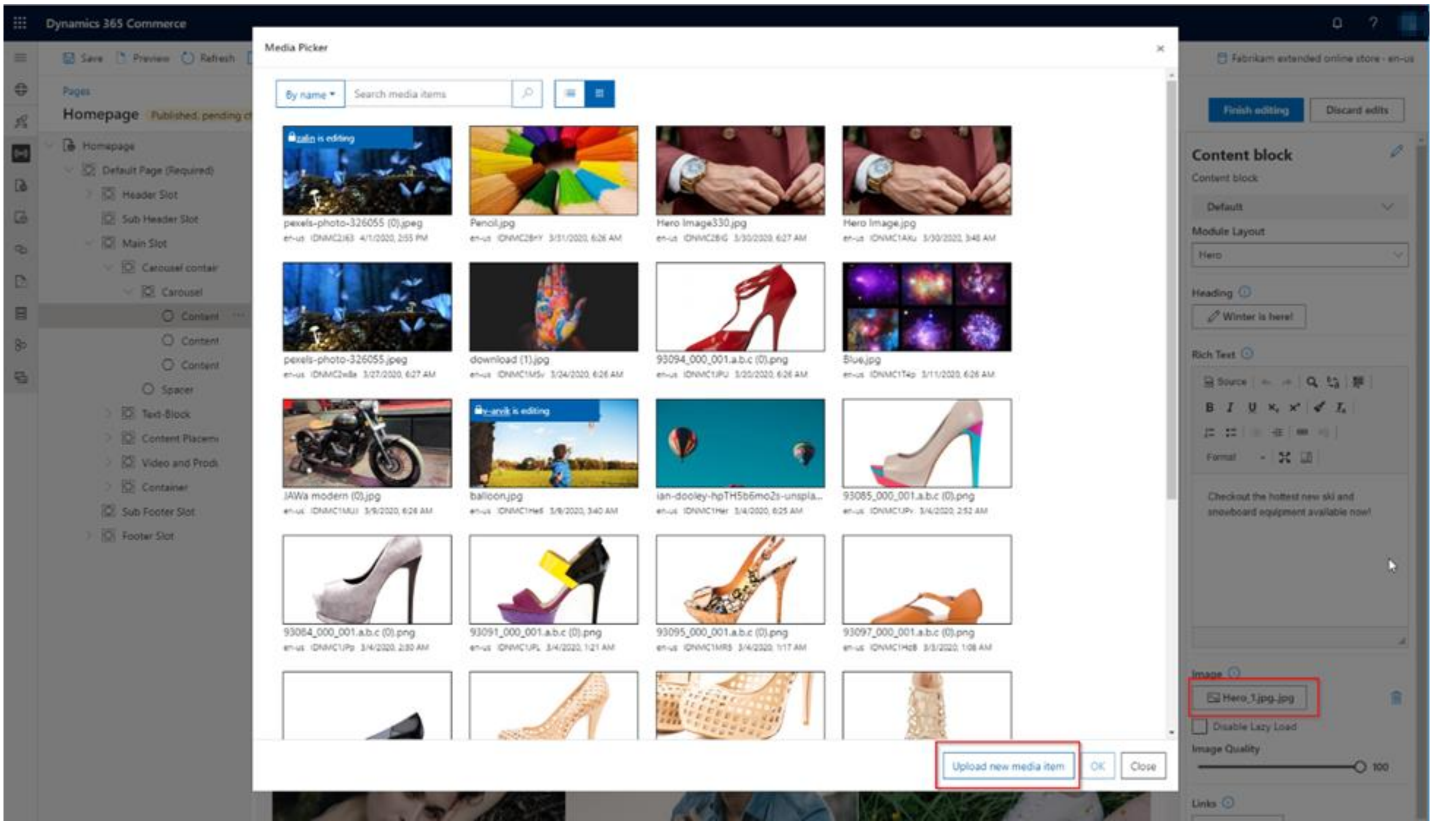
Image: Hero_1.jpg.jpg

□ Disable Lazy Load

Winter is here!

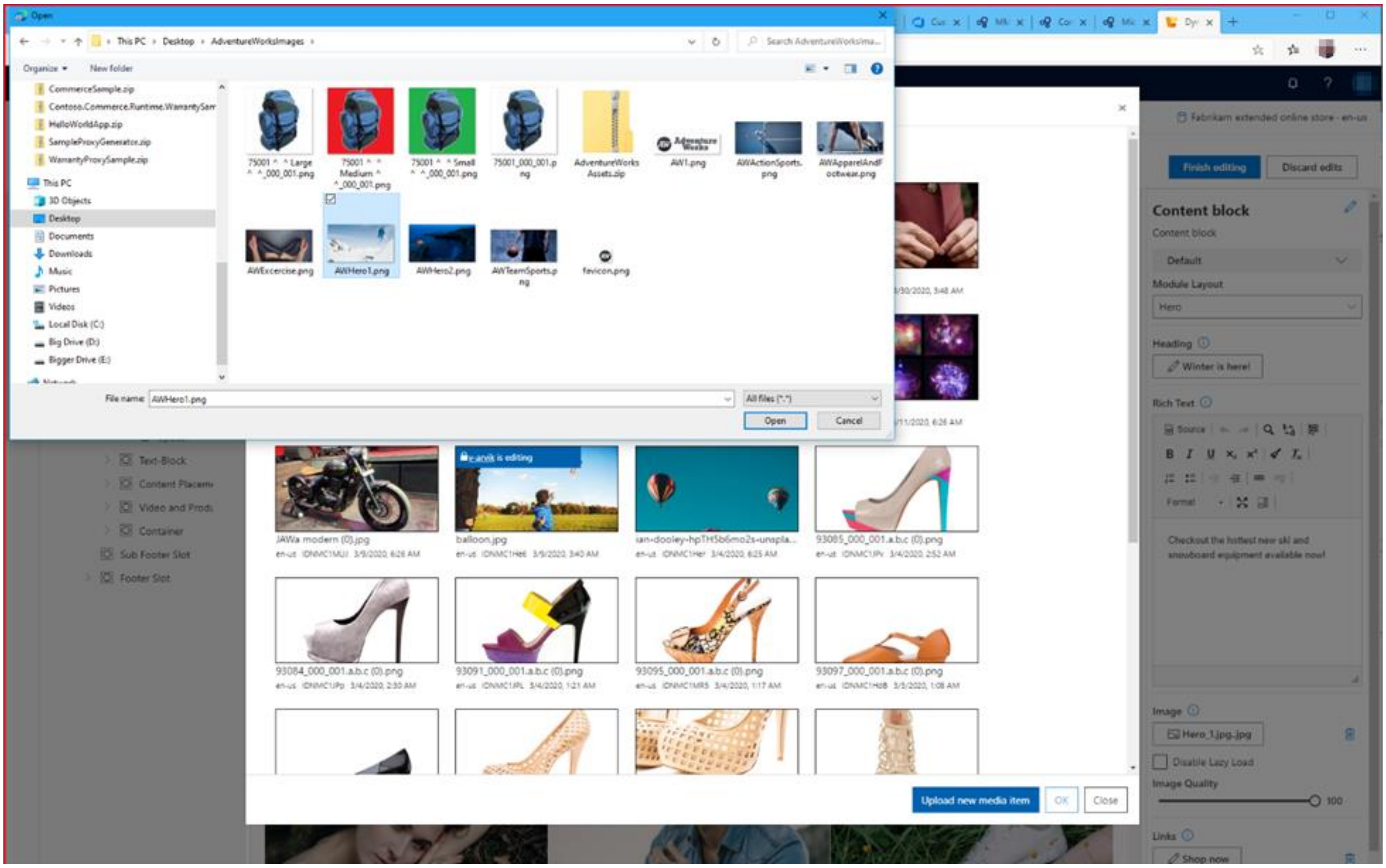
Checkout the hottest new ski and snowboard equipment available now!

Select the image button to bring up the media picker dialog box as shown below and select the “Upload new media item” button.


Dynamics 365 Commerce
Save
Preview
Refresh
Pages
Homepage
Published, pending c
Homepage
Default Page (Required)
Header Slot
Sub Header Slot
Main Slot
Carousel contair
Carousel
Content
Content
Content
Spacer
Text-Block
Content Placem
Video and Prod
Container
Sub Footer Slot
Footer Slot
Media Picker
By name
Search media items
zalin is editing
pexels-photo-326055 (0).jpeg
en-us IDNMC2J63 4/1/2020, 2:55 PM
Pencil.jpg
en-us IDNMC28rY 3/31/2020, 6:26 AM
Hero Image330.jpg
en-us IDNMC28iG 3/30/2020, 6:27 AM
Hero Image.jpg
en-us IDNMC1AKu 3/30/2020, 3:48 AM
pexels-photo-326055.jpeg
en-us IDNMC2w8a 3/27/2020, 6:27 AM
download (1).jpg
en-us IDNMC1M5v 3/24/2020, 6:26 AM
93094_000_001.a.b.c (0).png
en-us IDNMC1JPU 3/20/2020, 6:26 AM
Blue.jpg
en-us IDNMC1T4p 3/11/2020, 6:26 AM
JAWa modern (0).jpg
en-us IDNMC1MUJ 3/9/2020, 6:26 AM
v-arvik is editing
balloon.jpg
en-us IDNMC1He6 3/9/2020, 3:40 AM
ian-dooley-hpTH5b6mo2s-unspla...
en-us IDNMC1Her 3/4/2020, 6:25 AM
93085_000_001.a.b.c (0).png
en-us IDNMC1JPv 3/4/2020, 2:52 AM
93084_000_001.a.b.c (0).png
en-us IDNMC1JPs 3/4/2020, 2:30 AM
93091_000_001.a.b.c (0).png
en-us IDNMC1JPL 3/4/2020, 1:21 AM
93095_000_001.a.b.c (0).png
en-us IDNMC1MR8 3/4/2020, 1:17 AM
93097_000_001.a.b.c (0).png
en-us IDNMC1Hz8 3/3/2020, 1:08 AM
Upload new media item
OK
Close
Fabrikam extended online store - en-us
Finish editing
Discard edits
Content block
Source
Format
Disable Lazy Load
Links

Content block: Default
Module Layout: Hero
Heading: Winter is here!
Rich Text: Checkout the hottest new ski and snowboard equipment available now!
Image: Hero_1.jpg.jpg
Image Quality: 100

You can now select a hero image file provided or provide your own:



File name: AWHero1.png

All files (*.*)

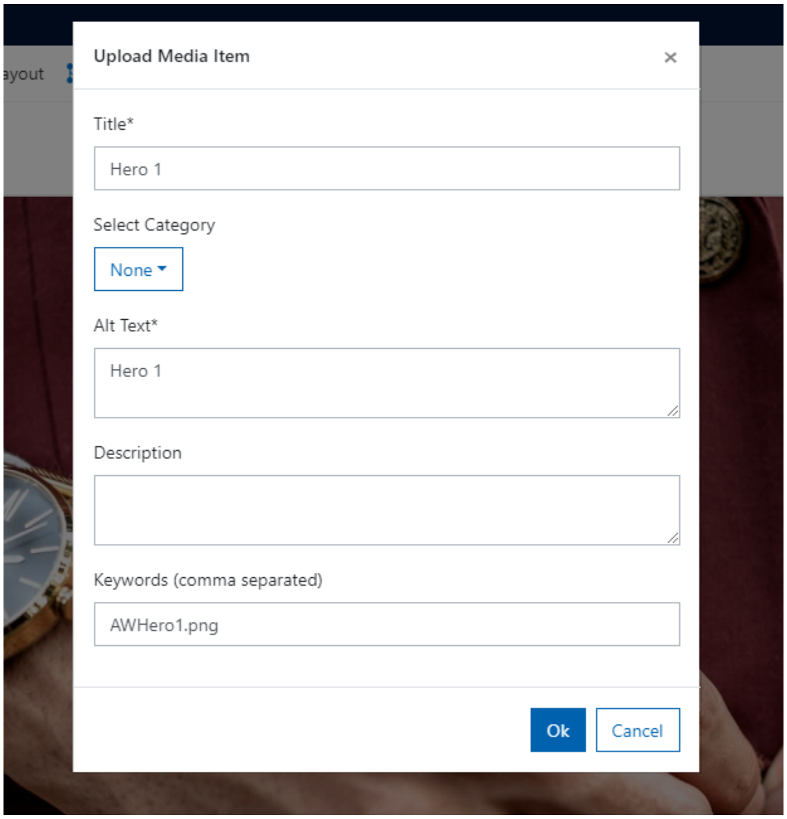
Heading: Winter is here!

Rich Text: Checkout the hottest new ski and snowboard equipment available now!

Image: Hero_1.jpg-.jpg

Links: Shop now

Provide an image title and alt text then click ok.



Upload Media Item

Title*

Hero 1

Select Category

None

Alt Text*

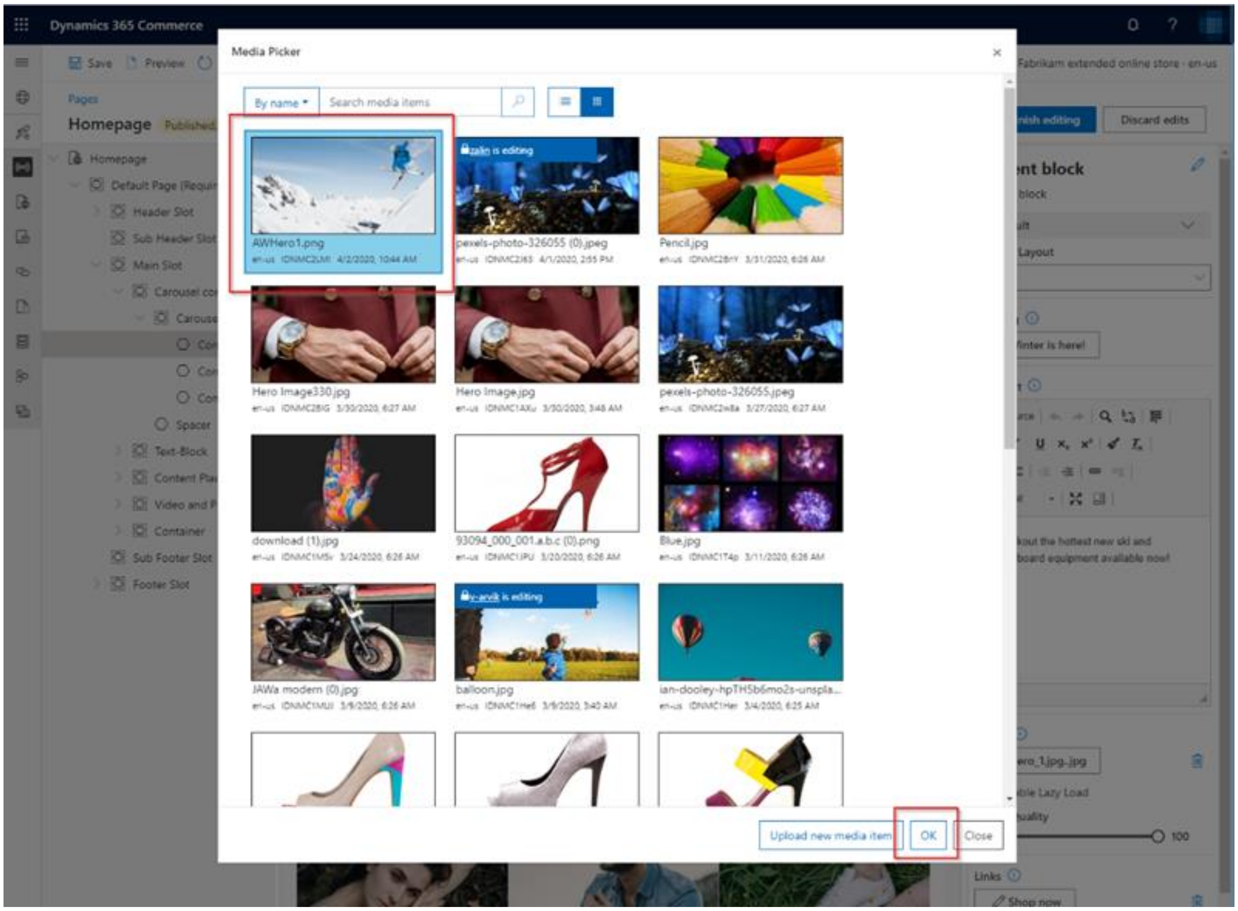
Hero 1

Description

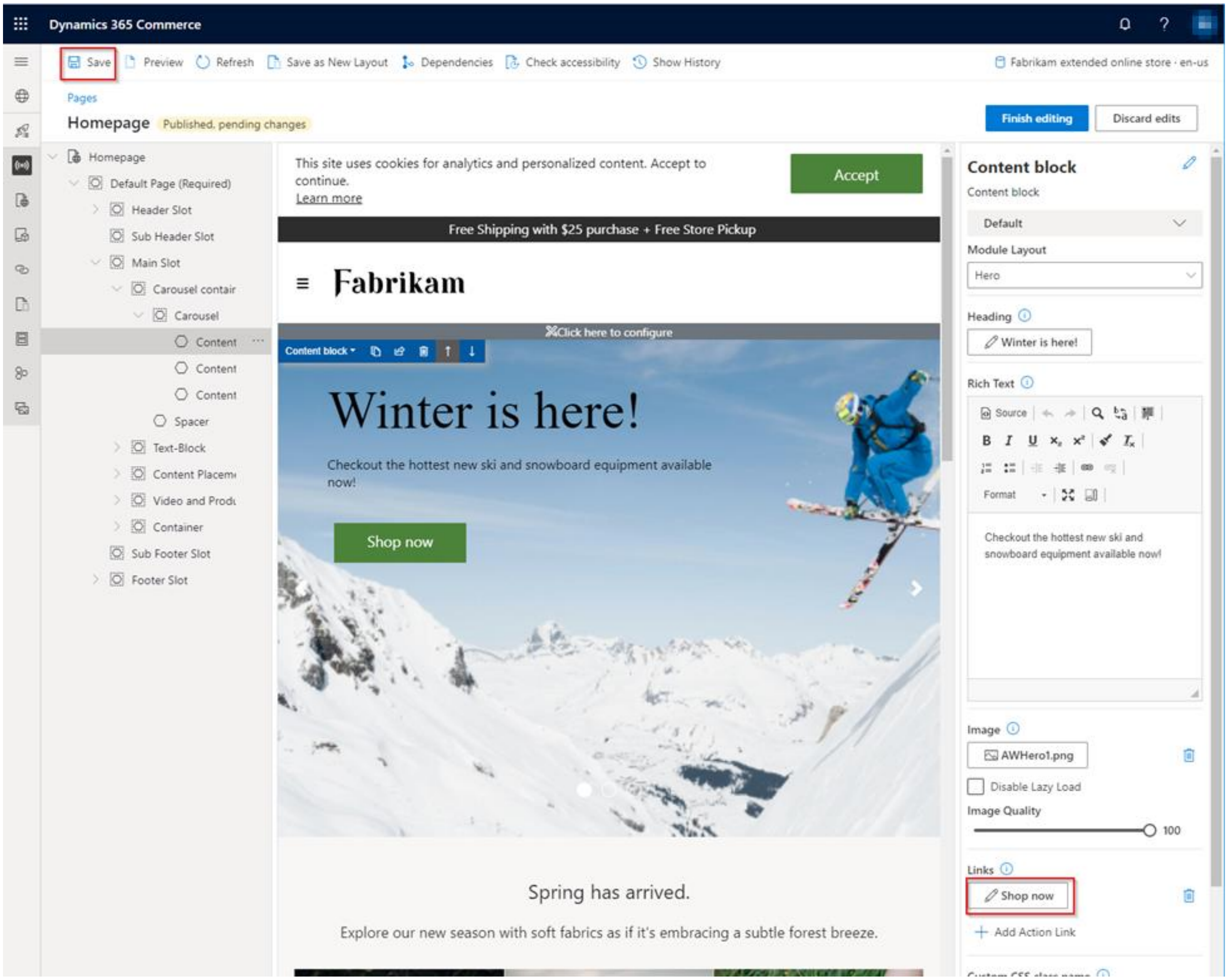
Keywords (comma separated)

AWHero1.png

Now you can select the image and click “OK”.


Dynamics 365 Commerce
Save
Preview
Pages
Homepage
Published
Fabrikam extended online store - en-us
Finish editing
Discard edits
Homepage
Default Page (Requir
Header Slot
Sub Header Slot
Main Slot
Carousel co
Carouse
Spacer
Text-Block
Content Pla
Video and P
Container
Sub Footer Slot
Footer Slot
Media Picker
By name
Search media items
AWHero1.png
en-us IDNMC2LMI 4/2/2020, 10:44 AM
zalin is editing
pexels-photo-326055 (0).jpeg
en-us IDNMC2I63 4/1/2020, 2:55 PM
Pencil.jpg
en-us IDNMC28rY 3/31/2020, 6:26 AM
Hero Image330.jpg
en-us IDNMC28IG 3/30/2020, 6:27 AM
Hero Image.jpg
en-us IDNMC1AXu 3/30/2020, 3:48 AM
pexels-photo-326055.jpeg
en-us IDNMC2w8a 3/27/2020, 6:27 AM
download (1).jpg
en-us IDNMC1M5r 3/24/2020, 6:26 AM
93094_000_001.a.b.c (0).png
en-us IDNMC1JPU 3/20/2020, 6:26 AM
Blue.jpg
en-us IDNMC1T4p 3/11/2020, 6:26 AM
JAWa modern (0).jpg
en-us IDNMC1MUI 3/9/2020, 6:26 AM
v-anvik is editing
balloon.jpg
en-us IDNMC1He6 3/9/2020, 3:40 AM
ian-dooley-hpTH5b6mo2s-unspla...
en-us IDNMC1Her 3/4/2020, 6:25 AM
Upload new media item
OK
Close
ent block
block
Layout
Winter is here!
kout the hottest new ski and
board equipment available now!
ero_1.jpg..jpg
ble Lazy Load
uality
100
Links
Shop now

Images will not be rendered until you click the “Save” button at the top.



This site uses cookies for analytics and personalized content. Accept to continue.

Free Shipping with $25 purchase + Free Store Pickup

Winter is here!

Checkout the hottest new ski and snowboard equipment available now!

Spring has arrived.

Explore our new season with soft fabrics as if it's embracing a subtle forest breeze.

Content block

Default

Module Layout: Hero

Heading: Winter is here!

Rich Text: Checkout the hottest new ski and snowboard equipment available now!

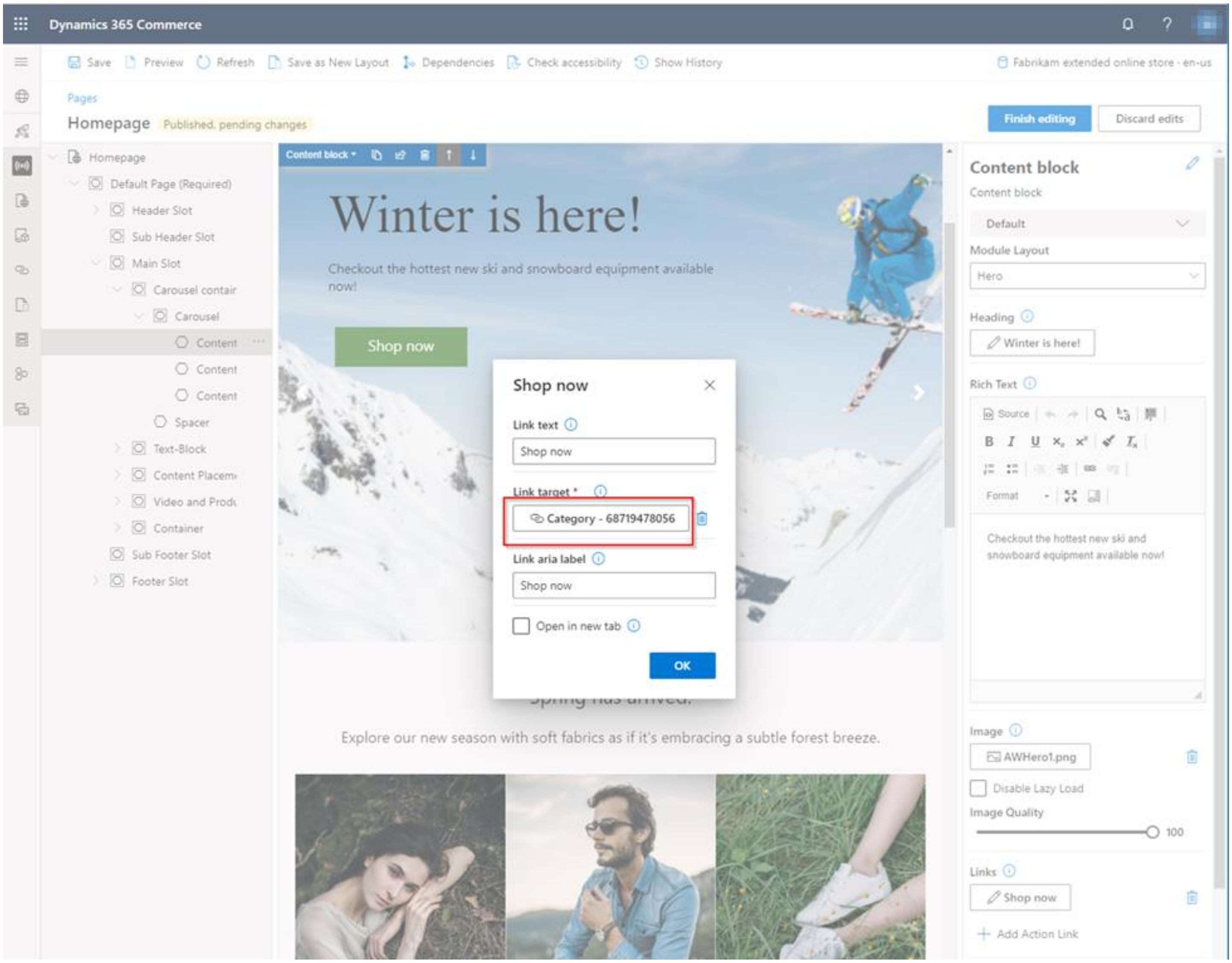
Image: AWHero1.png

□ Disable Lazy Load

Image Quality: 100

Links: Shop now

Next, we will configure the “Shop now” button to take us to one of our new categories. Select the button in the right column to bring up the link configuration.



Shop now

Link text: Shop now

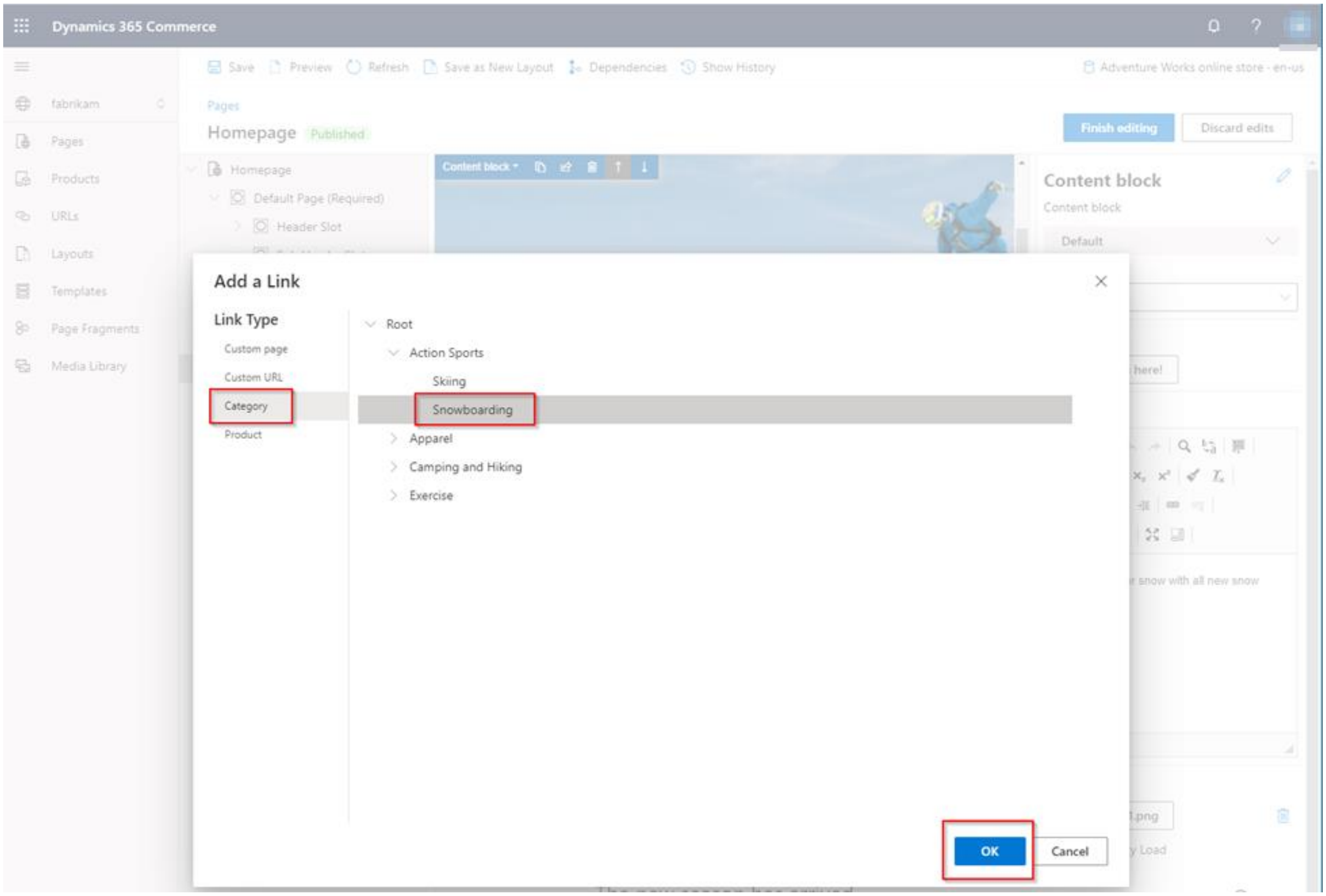
Link target *: Category - 68719478056

Link aria label: Shop now

□ Open in new tab

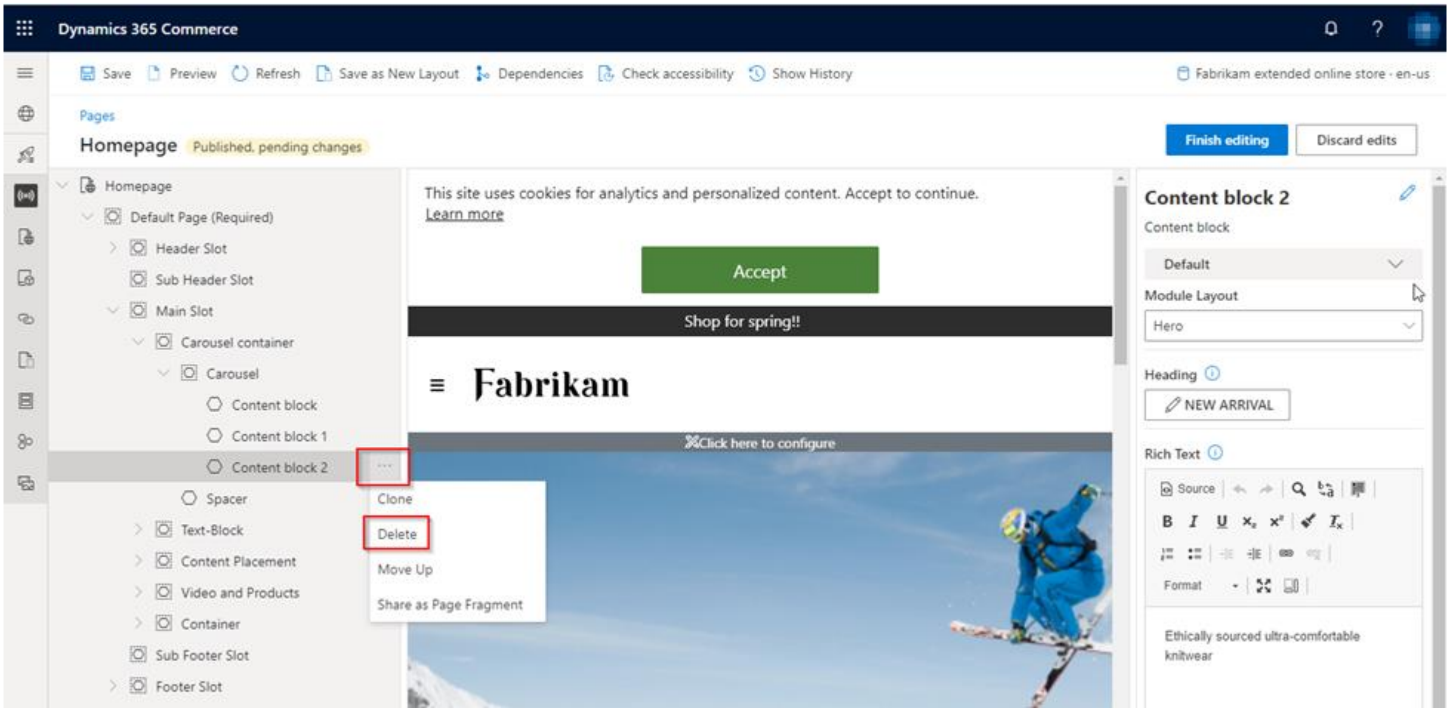
OK

Select the “Link target” to bring up the link selection. You can now select “Category” and the page you want selected, for example in below screenshot the link will go to the “Snowboarding” category page.



We are now done configuring the first “Hero” content block inside of the carousel. Follow the same steps to configure the second one.

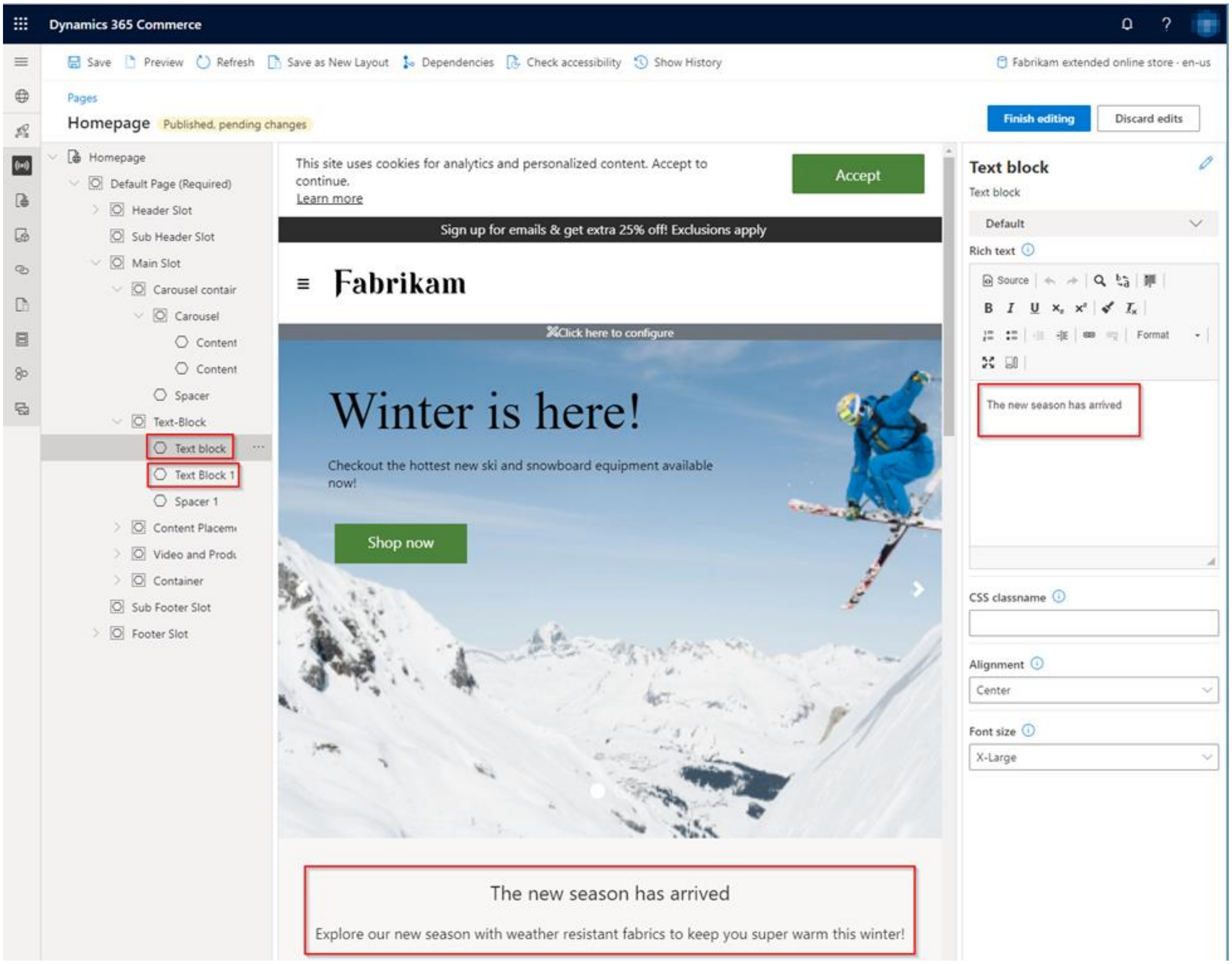
We will now delete the 3rd Content block. Select the module then the "…" followed by "Delete".



This site uses cookies for analytics and personalized content. Accept to continue. Learn more

Ethically sourced ultra-comfortable knitwear

We will now modify the “Text block” modules on the page. Select the first “Text block” module in the tree as shown below. And change the “Rich text” to something relevant to your branded store. Do the same for “Text Block 1”. Notice the text changes in the rendered preview as shown below.



This site uses cookies for analytics and personalized content. Accept to continue.

Sign up for emails & get extra 25% off! Exclusions apply

Winter is here!

Checkout the hottest new ski and snowboard equipment available now!

The new season has arrived

Explore our new season with weather resistant fabrics to keep you super warm this winter!

Text block

Text block

Default

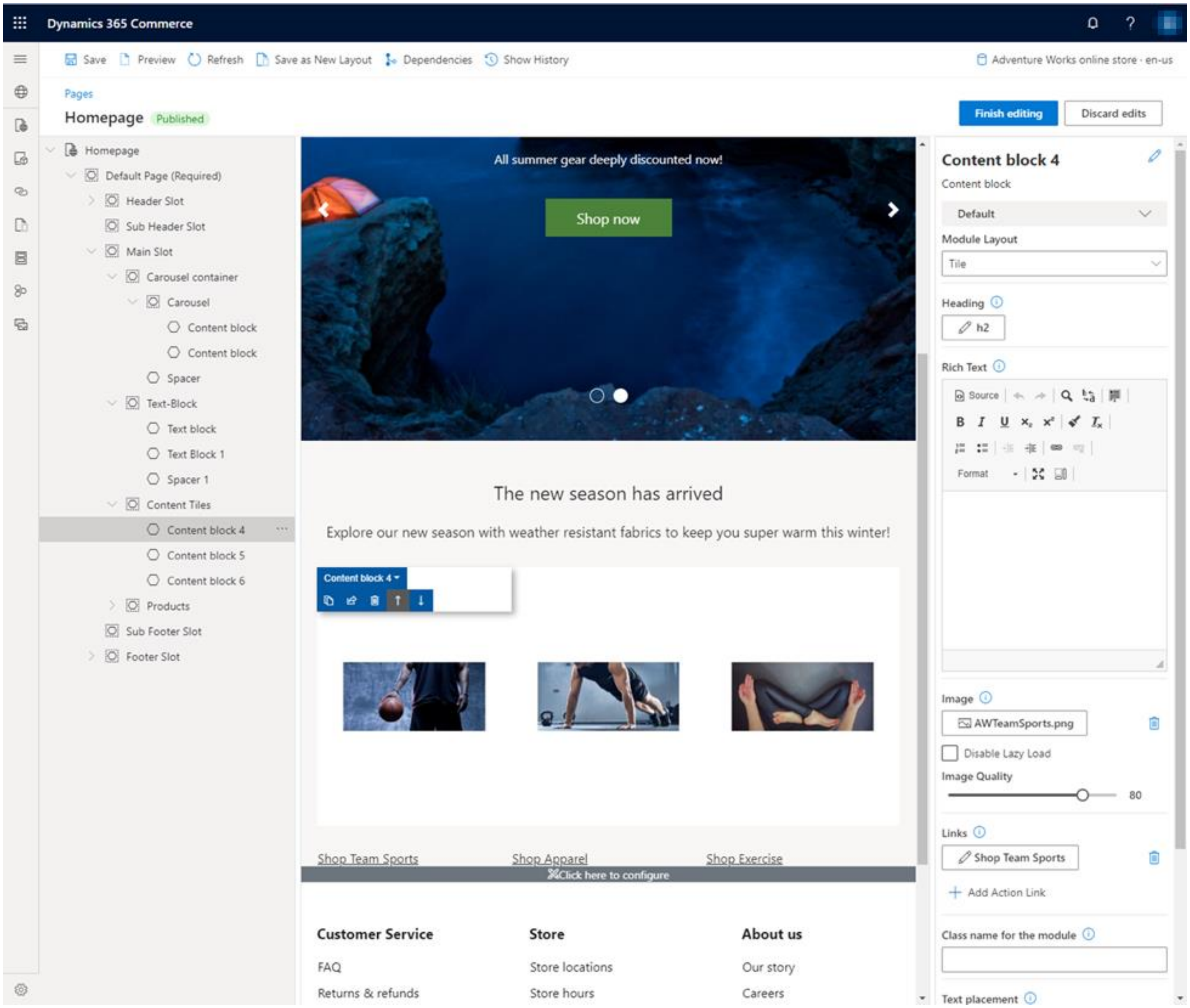
Rich text: The new season has arrived

CSS classname:

Alignment: Center

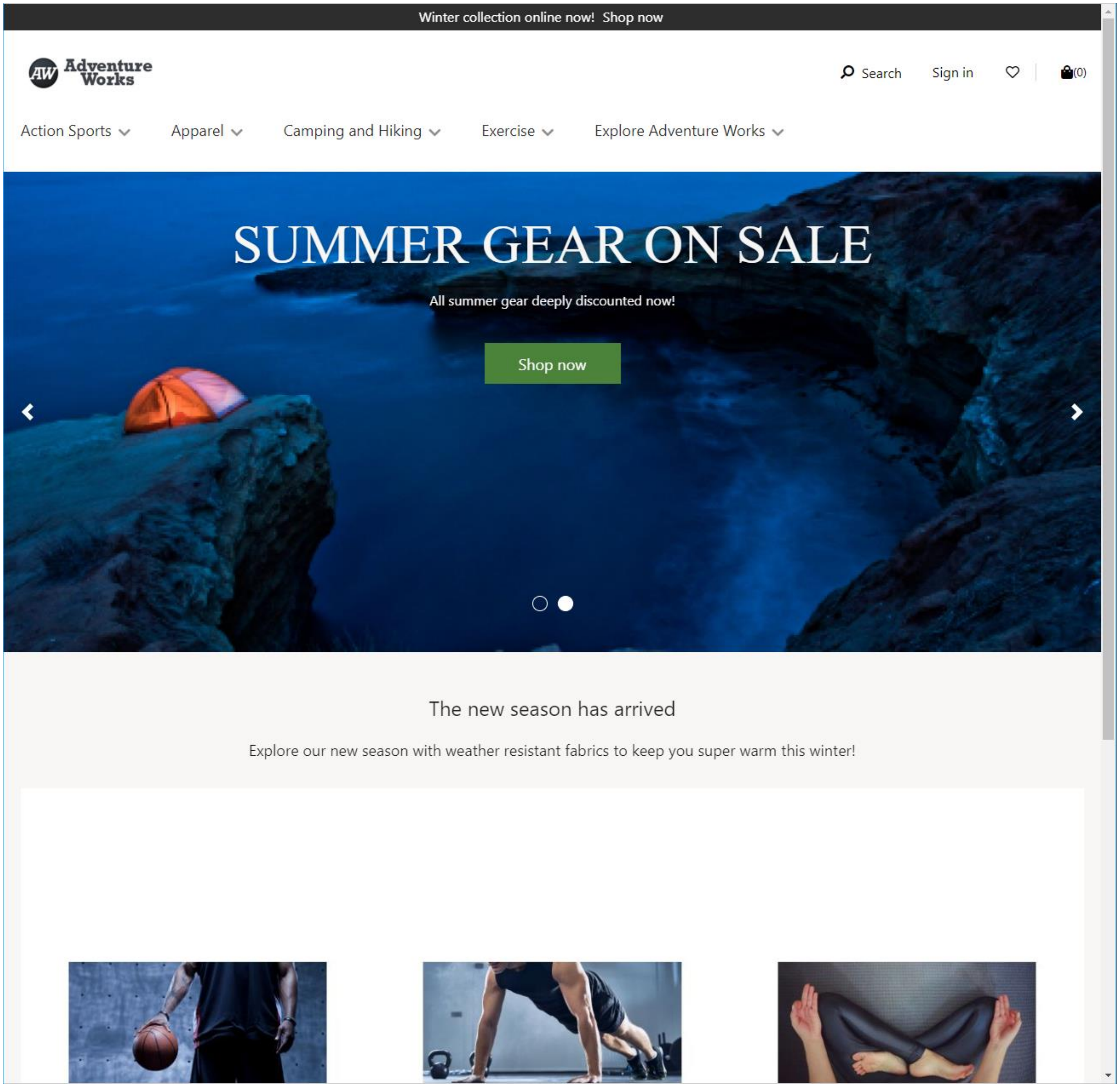
Font size: X-Large

There is another "Content Tiles" section (it may be named differently), this section has several "content blocks" setup as having a "Tile" layout. As an exercise select each one and update the image, text, and link to be more relevant. There are sample images that can be used in the provided zip folder. Ensure you press save when complete to see the images. Finally delete the "Video and Products" node. You may notice extra white space around these images since they are a different aspect ratio than the ones used on Fabrikam, in Unit 3 on extensibility we will show how this can be fixed.



We are now complete with editing the home page. Select "**Finish editing**" then publish changes

Preview the site and you should now have an “Adventure Works” branded site. However, the colors, style, fonts, … will still be Fabrikam theme based. We will update those in a later section.



The new season has arrived

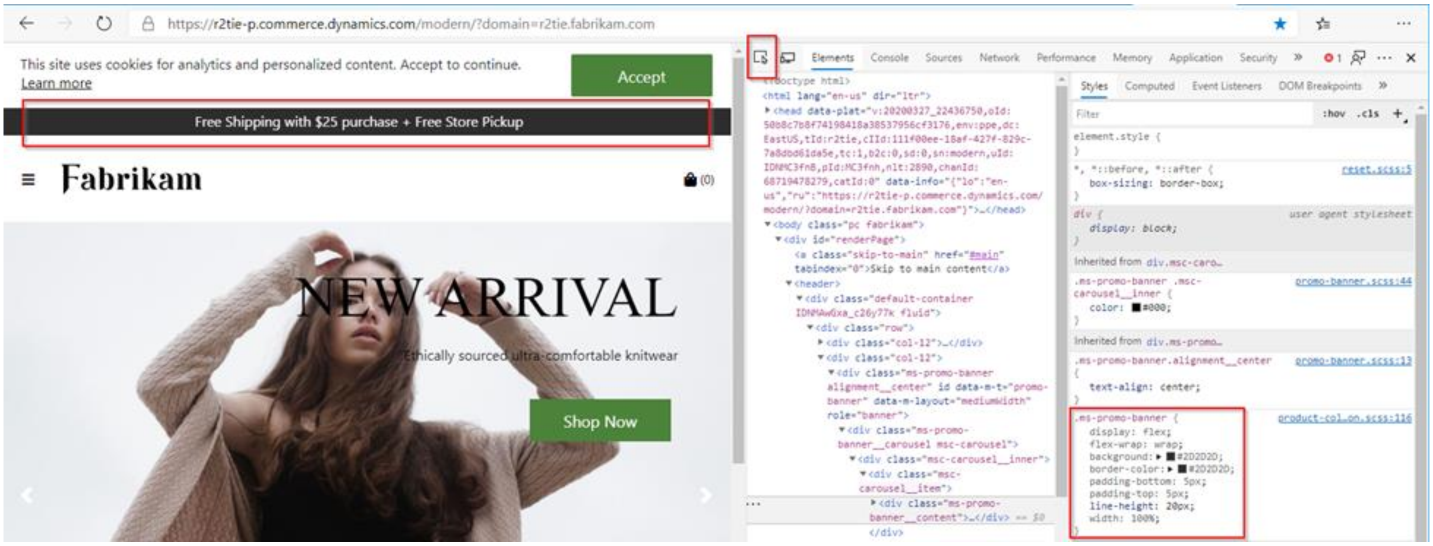
Explore our new season with weather resistant fabrics to keep you super warm this winter!

SITE THEMING WITH CSS OVERRIDES

Site theming is generally done by creating a SCSS based theme using the online store SDK, which requires setting up a development environment (see Unit 3 for details). However, you can make minor CSS changes directly in the site with a feature called "CSS Overrides".

To create a CSS override, start by creating a text file (you can use notepad) and name the file with the file extension .css, for example (siteOverrides.css).

In this file you can now add any CSS you would like. A great way to find appropriate CSS names to change is using a web browser's F12 tools on the site. For example, in Microsoft Edge or Chrome, you can click F12 on the homepage to open the explore tools and click the "inspect" button followed by an HTML element on the page. Notice in the below screen shot the promo banner is selected.


https://r2tie-p.commerce.dynamics.com/modern/?domain=r2tie.fabrikam.com
This site uses cookies for analytics and personalized content. Accept to continue.
Learn more
Accept
Free Shipping with $25 purchase + Free Store Pickup
Fabrikam
(0)
NEW ARRIVAL
Ethically sourced ultra-comfortable knitwear
Shop Now
Elements
Console
Sources
Network
Performance
Memory
Application
Security
1
Styles
Computed
Event Listeners
DOM Breakpoints
Filter
:hov
.cls
+

```
<!doctype html>
<html lang="en-us" dir="ltr">
 ▸<head data-plat="v:20200327_22436750,oId:
 50b8c7b8f74198418a38537956cf3176,env:ppe,dc:
 EastUS,tId:r2tie,cIId:111f00ee-18af-427f-829c-
 7a8dbd61da5e,tc:1,b2c:0,sd:0,sn:modern,uId:
 IDNMC3fn8,pId:MC3fnh,nlt:2890,chanId:
 68719478279,catId:0" data-info="{"lo":"en-
 us","ru":"https://r2tie-p.commerce.dynamics.com/
 modern/?domain=r2tie.fabrikam.com"}">…</head>
 ▾<body class="pc fabrikam">
   ▾<div id="renderPage">
       <a class="skip-to-main" href="#main"
       tabindex="0">Skip to main content</a>
     ▾<header>
       ▾<div class="default-container
       IDNMAwGxa_c26y77k fluid">
         ▾<div class="row">
           ▸<div class="col-12">…</div>
           ▾<div class="col-12">
             ▾<div class="ms-promo-banner
             alignment__center" id data-m-t="promo-
             banner" data-m-layout="mediumWidth"
             role="banner">
               ▾<div class="ms-promo-
               banner__carousel msc-carousel">
                 ▾<div class="msc-carousel__inner">
                   ▾<div class="msc-
                   carousel__item">
...                  ▸<div class="ms-promo-
                     banner__content">…</div> == $0
                     </div>
```

```
element.style {
}
*, *::before, *::after {                     reset.scss:5
   box-sizing: border-box;
}
div {                                 user agent stylesheet
   display: block;
}
Inherited from div.msc-caro…
.ms-promo-banner .msc-               promo-banner.scss:44
carousel__inner {
   color: ■#000;
}
Inherited from div.ms-promo…
.ms-promo-banner.alignment__center   promo-banner.scss:13
{
   text-align: center;
}
.ms-promo-banner {                 product-col…on.scss:116
   display: flex;
   flex-wrap: wrap;
   background: ▸ ■#2D2D2D;
   border-color: ▸ ■#2D2D2D;
   padding-bottom: 5px;
   padding-top: 5px;
   line-height: 20px;
   width: 100%;
}
```

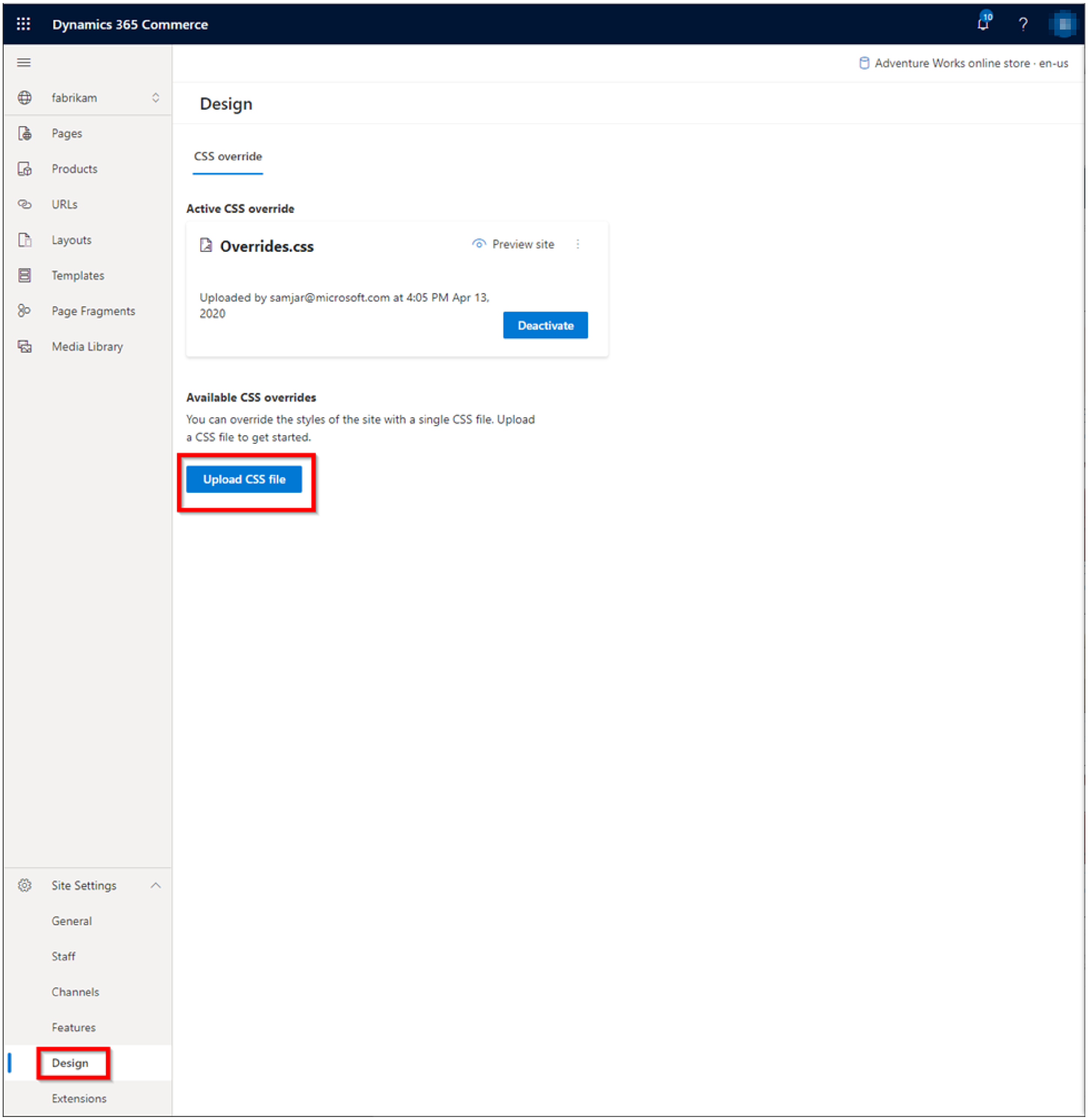
Now you can copy the css from the right side and modify. For example, if we want to only change the color of the banner add the following to my override file. This CSS file will then be applied on the site.

```
.ms-promo-banner {
   background: red;
}
```

To upload the CSS file, follow the steps below.

Expand the “**Site Settings**” tab on the bottom left and select the “**Design**” tab. You can now select the “Upload CSS file” button and upload the new file. You can now preview the site with the overrides using the “Preview site” button. When you’re ready to turn the overrides on, select the “**Activate**” button and it will be live immediately on the production site.

Note: Many override files can be uploaded but only 1 can be active on the site at a time.



Design

Active CSS override

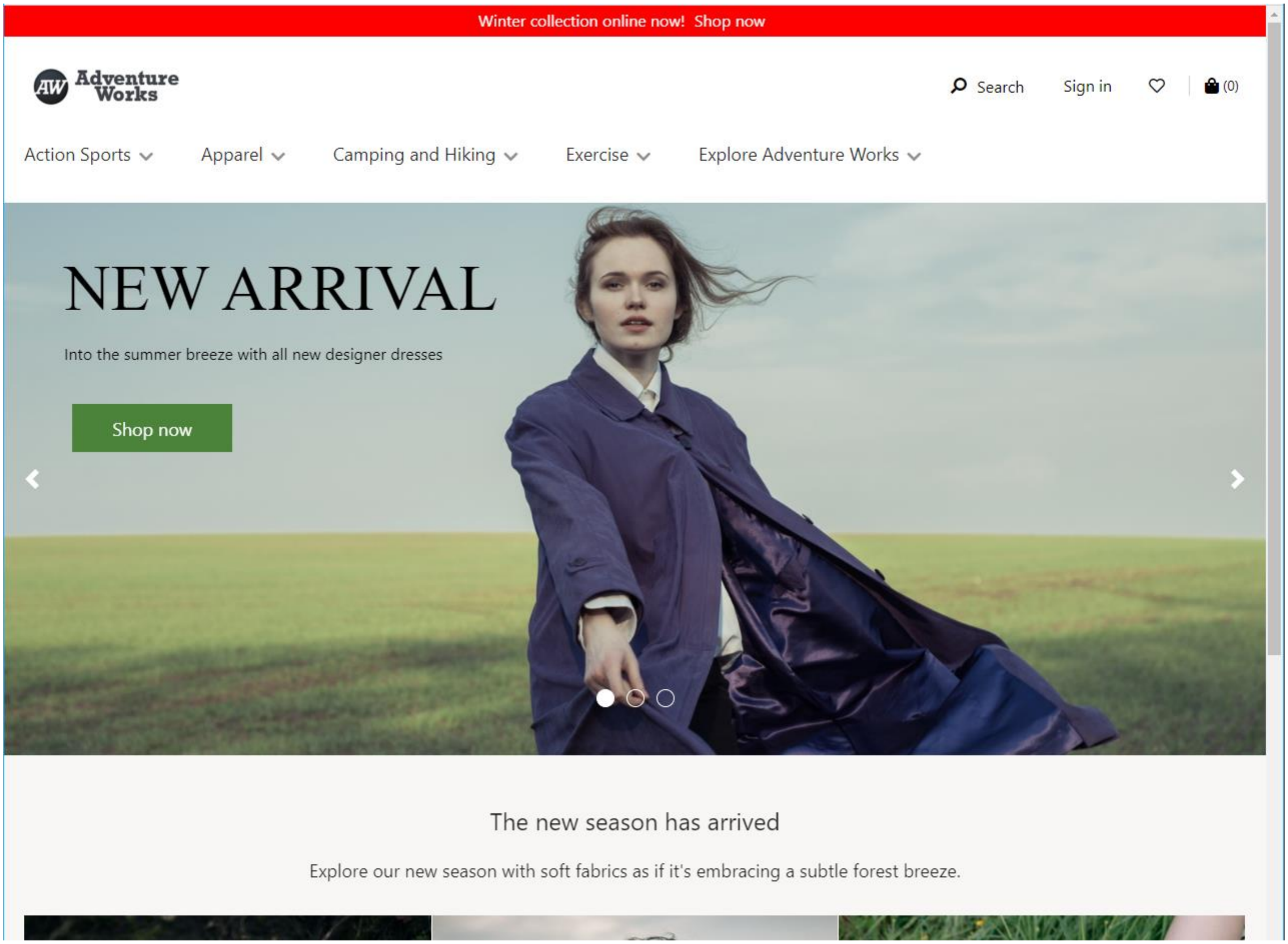
Overrides.css

Uploaded by samjar@microsoft.com at 4:05 PM Apr 13, 2020

Available CSS overrides

You can override the styles of the site with a single CSS file. Upload a CSS file to get started.

Previewing the site should now show with your overrides as shown below…



Into the summer breeze with all new designer dresses

The new season has arrived

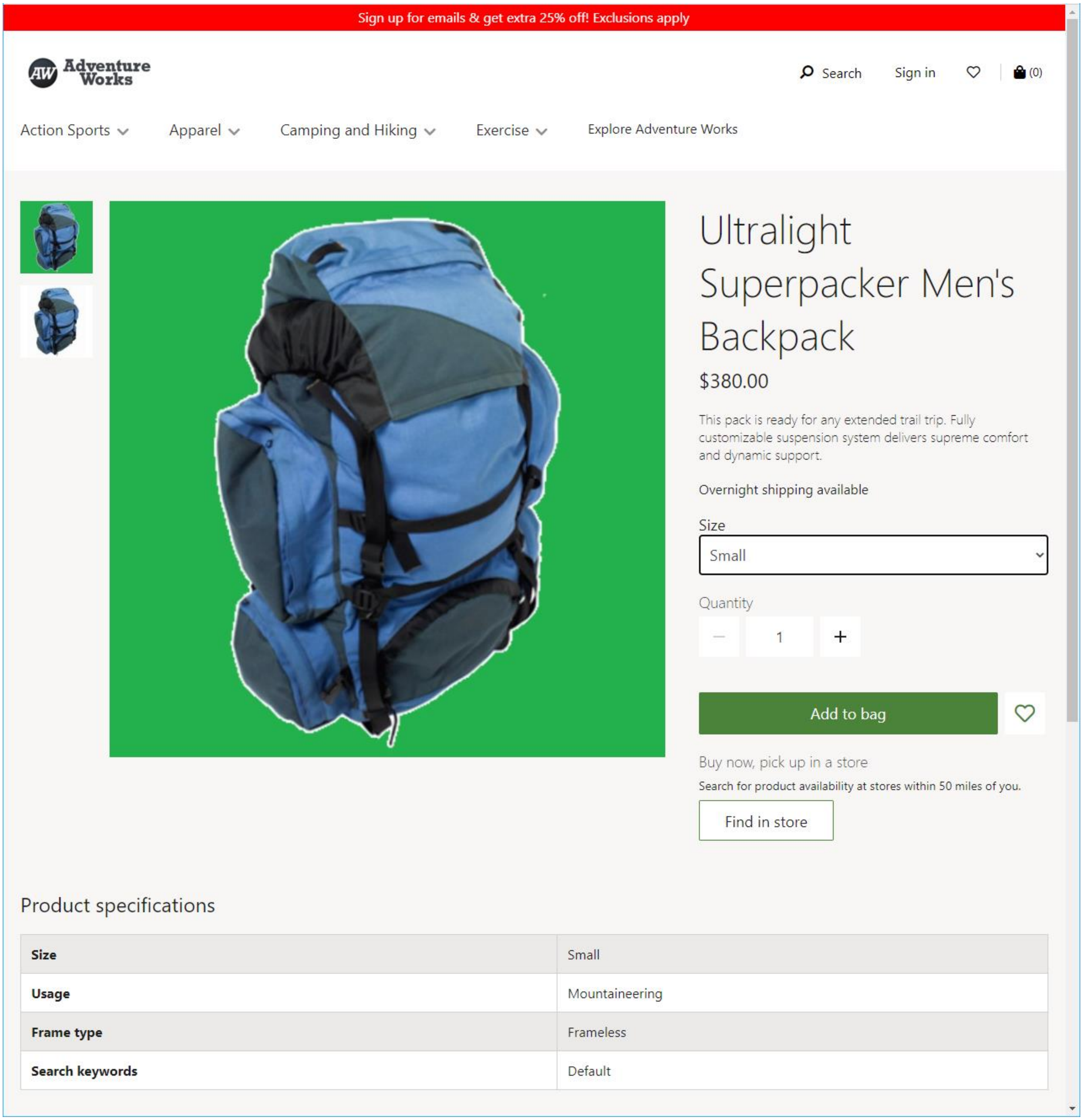
Explore our new season with soft fabrics as if it's embracing a subtle forest breeze.

END TO END SITE TESTING

In this section we will guide you to test your site which will include adding a product to the cart, making a purchase and picking up part of the order in a store.

ADDING PRODUCTS TO CART AND CHECKOUT

Navigate to a product page, select a product variant, and select the “Add to bag” button. Add a couple items to the cart.



Sign up for emails & get extra 25% off! Exclusions apply

Ultralight Superpacker Men's Backpack

$380.00

This pack is ready for any extended trail trip. Fully customizable suspension system delivers supreme comfort and dynamic support.

Overnight shipping available

Size: Small

Quantity: 1

Buy now, pick up in a store

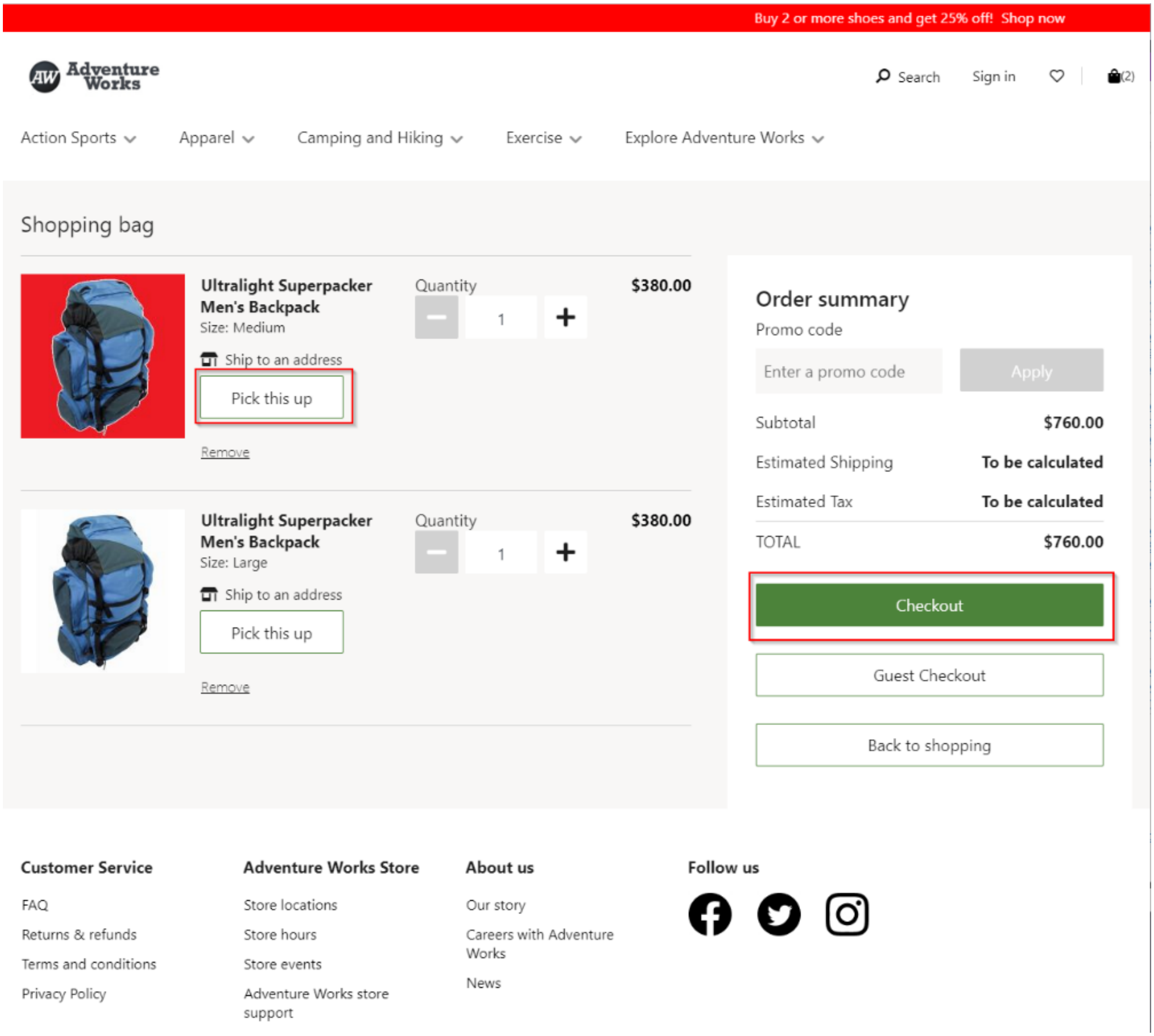
Search for product availability at stores within 50 miles of you.

Product specifications

Size	Small
Usage	Mountaineering
Frame type	Frameless
Search keywords	Default

Try selecting one of the products to pick up in “San Francisco” store and checkout.

Note: Please note at this stage if Azure Active Directory B2C hasn’t been setup you will need to do a “Guest Checkout”



Buy 2 or more shoes and get 25% off! Shop now

Adventure Works

Search Sign in (2)

Action Sports Apparel Camping and Hiking Exercise Explore Adventure Works

Shopping bag

Ultralight Superpacker Men's Backpack
Size: Medium
Ship to an address
Pick this up
Remove
Quantity 1 **$380.00**

Ultralight Superpacker Men's Backpack
Size: Large
Ship to an address
Pick this up
Remove
Quantity 1 **$380.00**

Order summary

Promo code

Enter a promo code Apply

Subtotal	**$760.00**
Estimated Shipping	**To be calculated**
Estimated Tax	**To be calculated**
TOTAL	**$760.00**

Checkout

Guest Checkout

Back to shopping

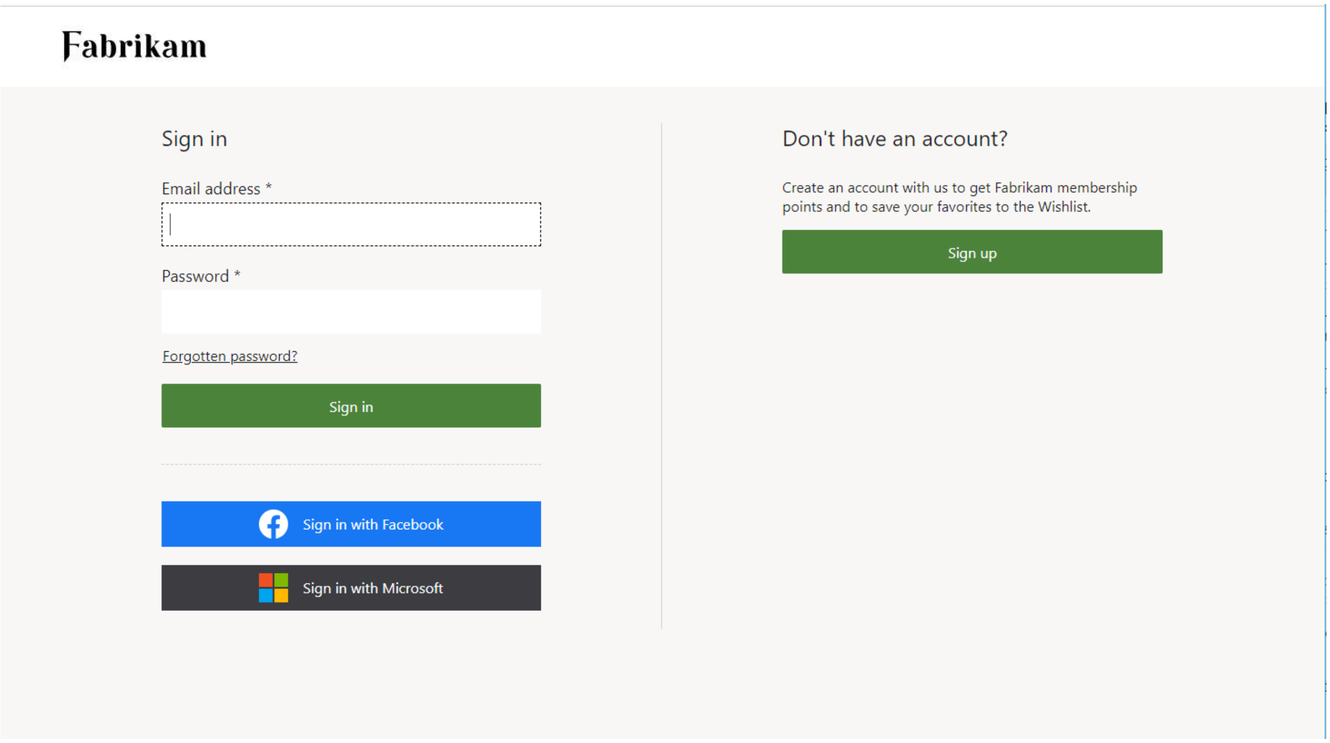
Customer Service
FAQ
Returns & refunds
Terms and conditions
Privacy Policy

Adventure Works Store
Store locations
Store hours
Store events
Adventure Works store support

About us
Our story
Careers with Adventure Works
News

Follow us

If you have not configured B2C yet please use guess checkout.


Fabrikam

Sign in

Email address *

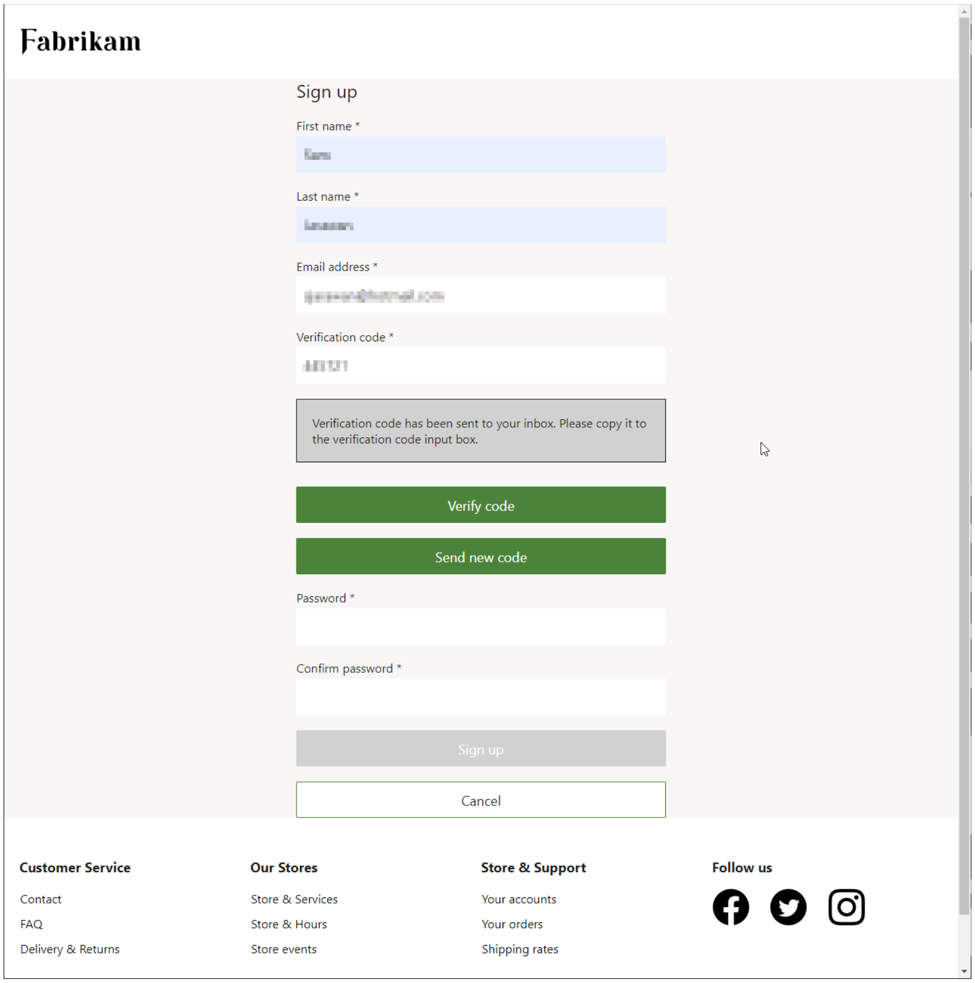
Password *

Forgotten password?

Don't have an account?

Create an account with us to get Fabrikam membership points and to save your favorites to the Wishlist.

Once created click Sign-Up to create an account, it will send a verification code and then sign in.



Fabrikam

Sign up

First name *

[redacted]

Last name *

[redacted]

Email address *

[redacted]

Verification code *

[redacted]

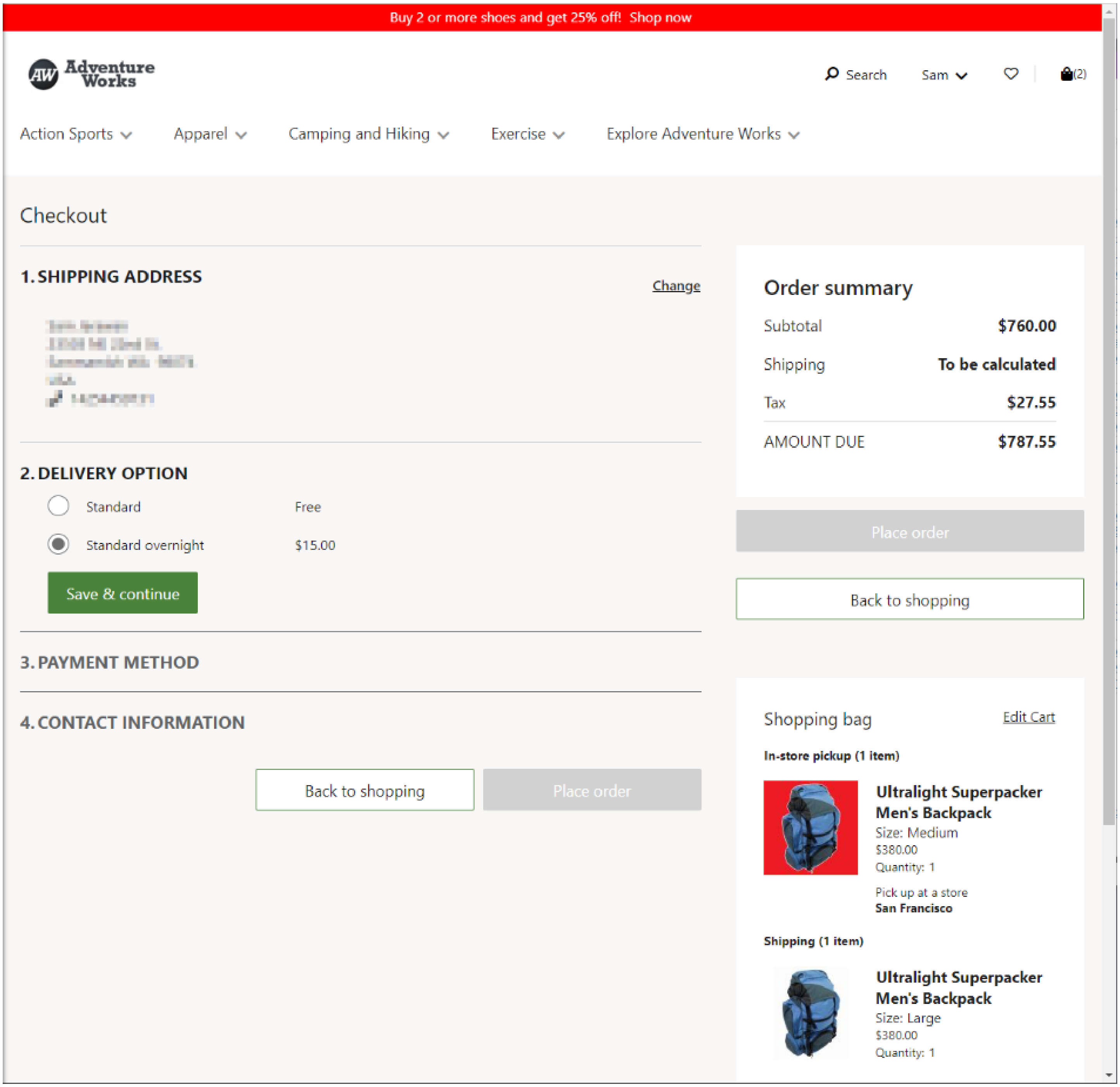
Verification code has been sent to your inbox. Please copy it to the verification code input box.

Password *

Confirm password *

Fill out Shipping Address section and select “Save & continue”.

Select a delivery mode and “Save & continue” (if none are available – ensure you ran process delivery modes in HQ or run again).



Buy 2 or more shoes and get 25% off! Shop now

Checkout

1. SHIPPING ADDRESS Change

[redacted]

2. DELIVERY OPTION

○ Standard	Free
● Standard overnight	$15.00

Save & continue

3. PAYMENT METHOD

4. CONTACT INFORMATION

Back to shopping | Place order

Order summary

Subtotal	**$760.00**
Shipping	**To be calculated**
Tax	**$27.55**
AMOUNT DUE	**$787.55**

Place order

Back to shopping

Shopping bag — Edit Cart

In-store pickup (1 item)

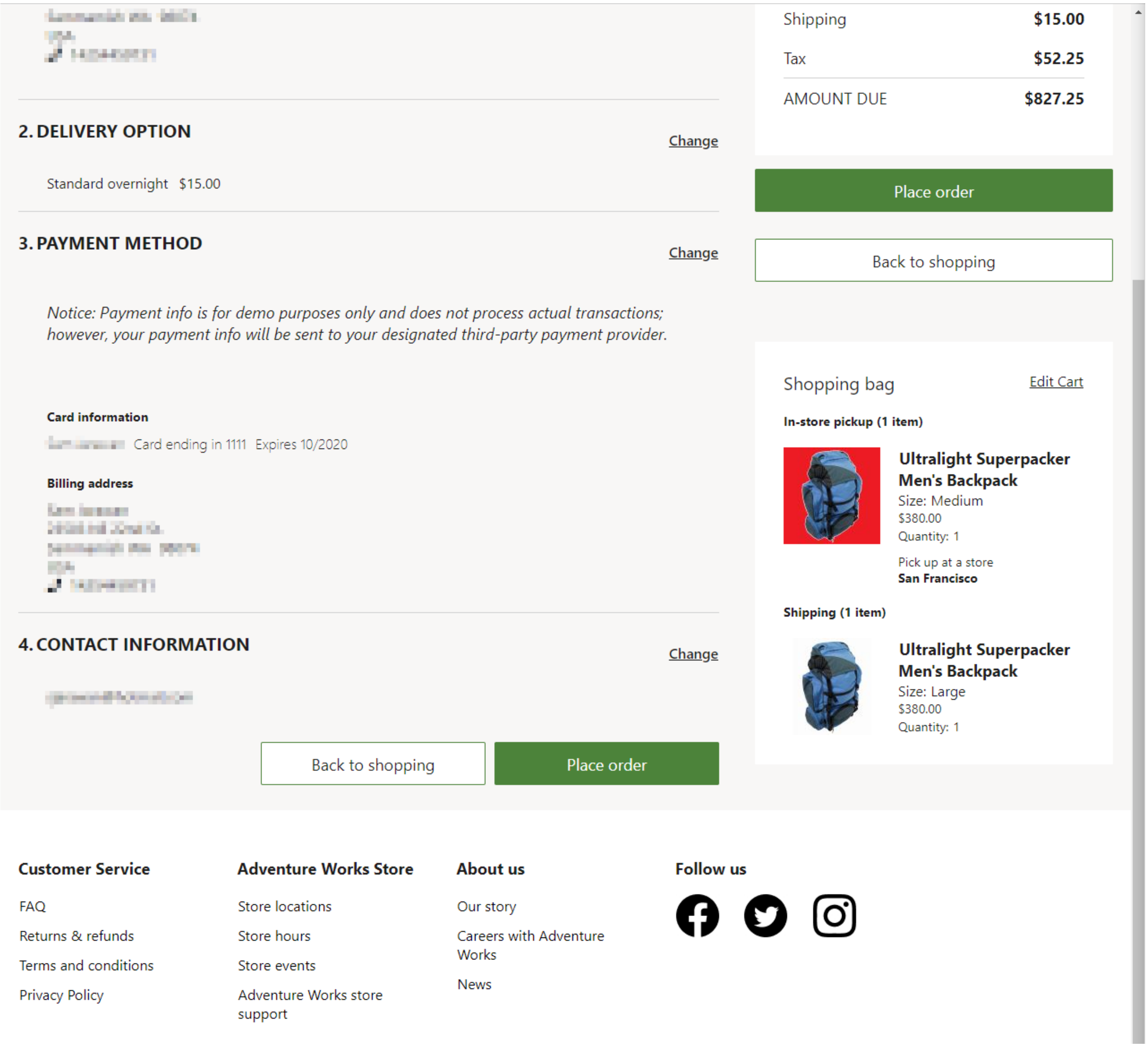
Ultralight Superpacker Men's Backpack
Size: Medium
$380.00
Quantity: 1

Pick up at a store
San Francisco

Shipping (1 item)

Ultralight Superpacker Men's Backpack
Size: Large
$380.00
Quantity: 1

Fill out payment method use any name and credit card number **4111 1111 1111 1111** expiry **03/30** CVC **737**, provide email in contact information and click on “**Place order**”. You should get a confirmation email.



[redacted]

2. DELIVERY OPTION

Change

Standard overnight $15.00

3. PAYMENT METHOD

Change

Notice: Payment info is for demo purposes only and does not process actual transactions; however, your payment info will be sent to your designated third-party payment provider.

Card information

[redacted] Card ending in 1111 Expires 10/2020

Billing address

[redacted]

4. CONTACT INFORMATION

Change

[redacted]

Shipping	**$15.00**
Tax	**$52.25**
AMOUNT DUE	**$827.25**

Shopping bag

Edit Cart

In-store pickup (1 item)

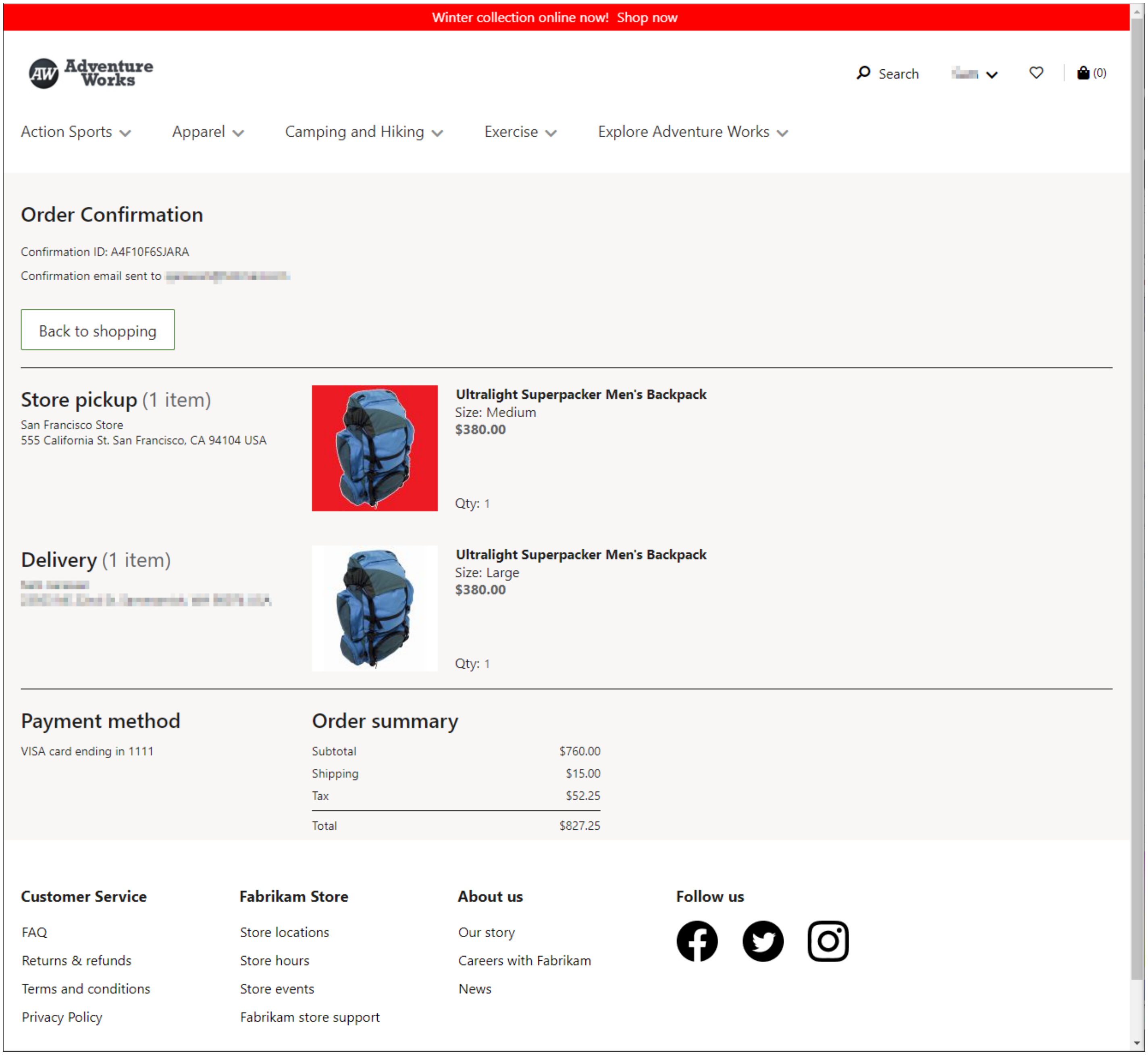
Ultralight Superpacker Men's Backpack
Size: Medium
$380.00
Quantity: 1

Pick up at a store
San Francisco

Shipping (1 item)

Ultralight Superpacker Men's Backpack
Size: Large
$380.00
Quantity: 1

You should get an order confirmation page like below.



Order Confirmation

Confirmation ID: A4F10F6SJARA

Confirmation email sent to [redacted]

Store pickup (1 item)

San Francisco Store
555 California St. San Francisco, CA 94104 USA

Ultralight Superpacker Men's Backpack
Size: Medium
$380.00

Qty: 1

Delivery (1 item)

[redacted]

Ultralight Superpacker Men's Backpack
Size: Large
$380.00

Qty: 1

Payment method

VISA card ending in 1111

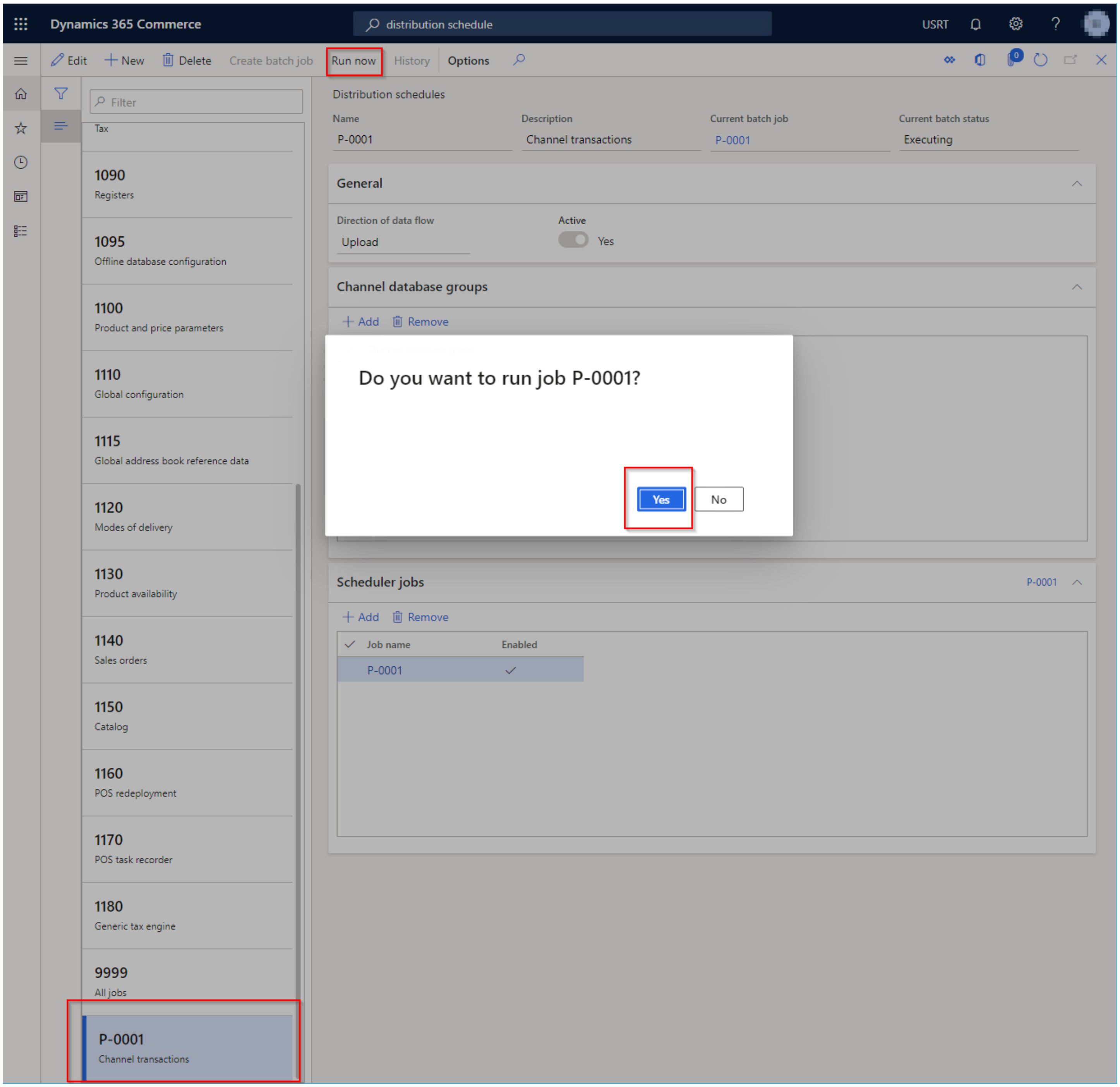
Order summary

Subtotal	$760.00
Shipping	$15.00
Tax	$52.25
Total	$827.25

SEEING THE ORDER IN HEADQUARTERS

Before we can see the transaction in the Headquarters tool, we need to run the “P-0001” job and “Synchronize Orders” to pull in the orders from the Commerce scale units.

- Search for the “**Distribution schedule**” page
- Select the “**P-0001**” job and click “**Run now**”



Distribution schedules

Name	Description	Current batch job	Current batch status
P-0001	Channel transactions	P-0001	Executing

Direction of data flow: Upload

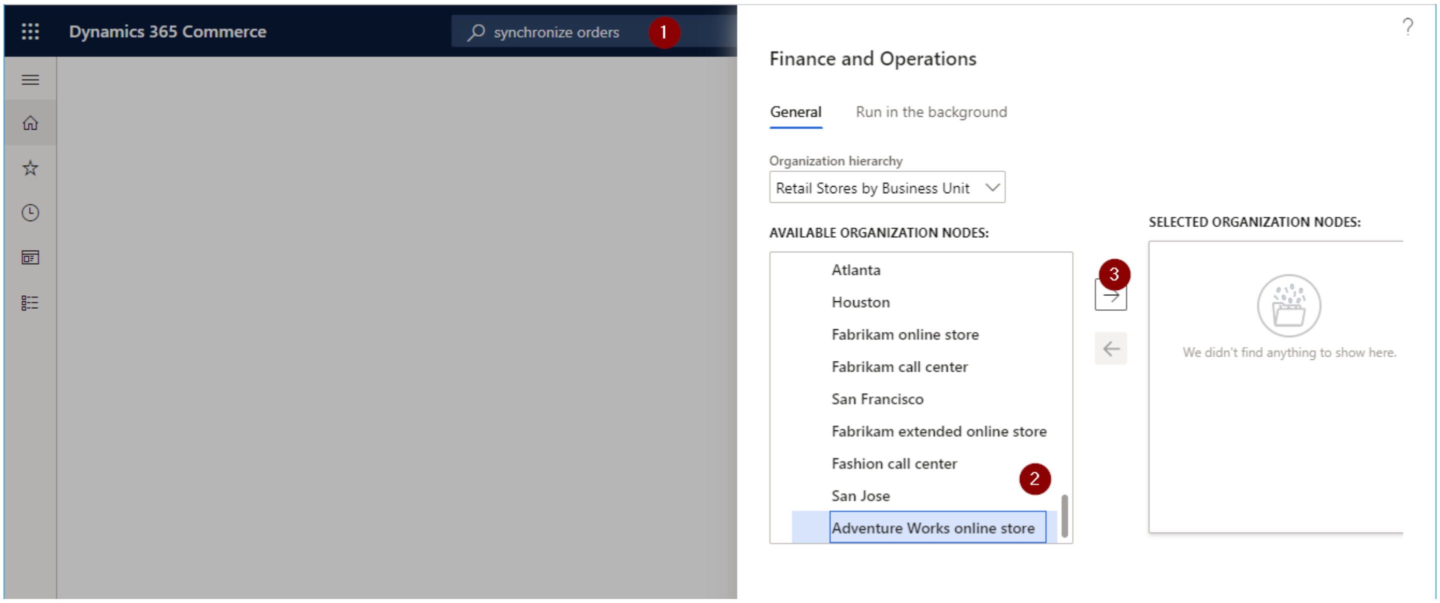
Active: Yes

Job name	Enabled
P-0001	✓

Do you want to run job P-0001?

Yes No

- Next, we need to synchronize the orders down from the scale units. Search for “**Synchronize Orders**”, select the online channel you are working with (such as “Adventure Works online store” and add it to the “SELECTED ORGANIZATION NODES” followed by clicking the “**OK**” button.



Organization hierarchy: Retail Stores by Business Unit

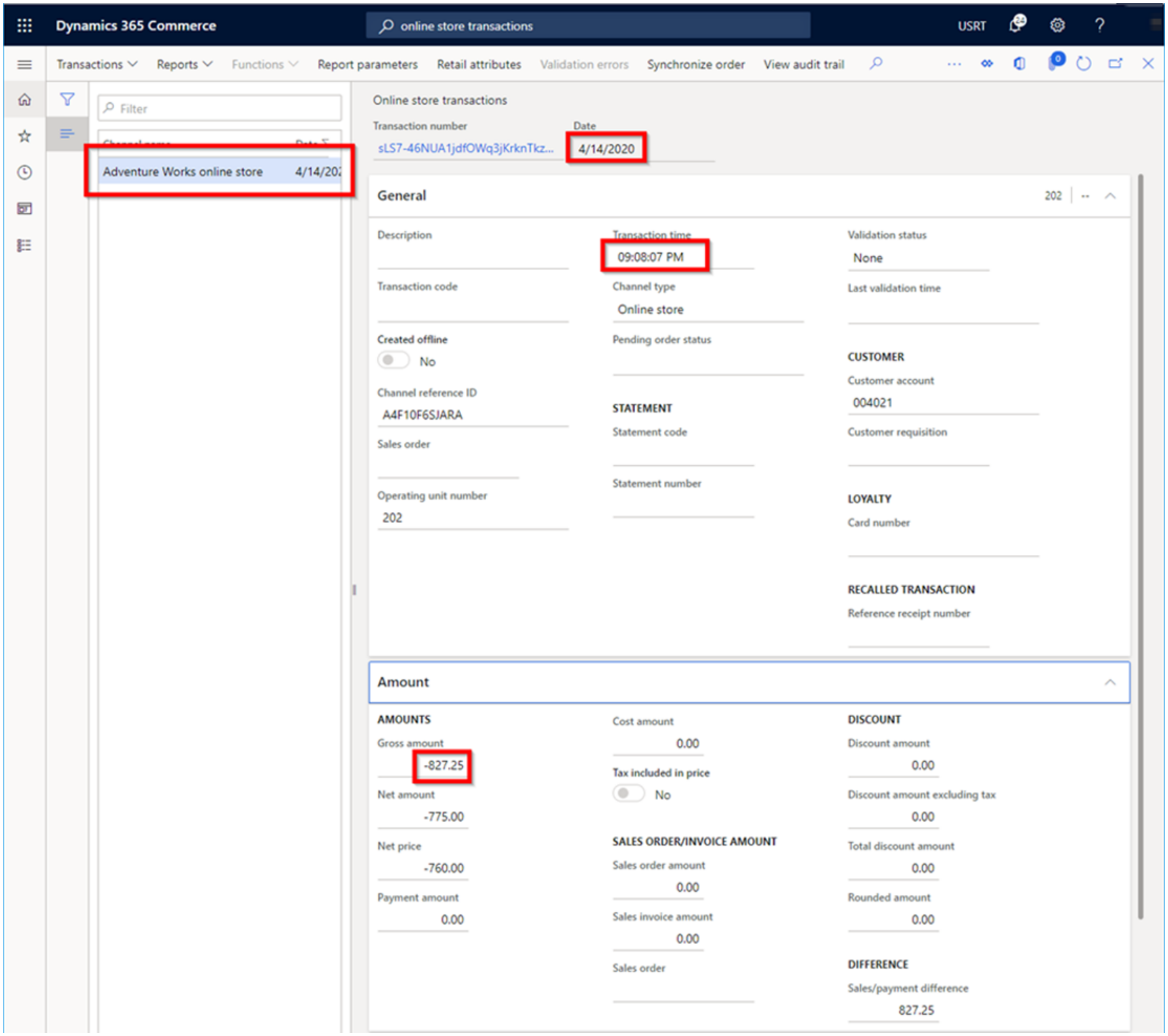
AVAILABLE ORGANIZATION NODES:

Atlanta
Houston
Fabrikam online store
Fabrikam call center
San Francisco
Fabrikam extended online store
Fashion call center
San Jose
Adventure Works online store

SELECTED ORGANIZATION NODES:

- Lastly, we can search for “Online store transactions” and you should find the order in the transaction list.

You should be able to match the date, time, and order charge to the order you created on the site.



Adventure Works online store 4/14/202

Online store transactions

Transaction number: sLS7-46NUA1jdfOWq3jKrknTkz...
Date: 4/14/2020

General 202

Description:
Transaction time: 09:08:07 PM
Validation status: None
Transaction code:
Channel type: Online store
Last validation time:
Created offline: No
Pending order status:
Channel reference ID: A4F10F6SJARA
Sales order:
Operating unit number: 202

STATEMENT
Statement code:
Statement number:

CUSTOMER
Customer account: 004021
Customer requisition:

LOYALTY
Card number:

RECALLED TRANSACTION
Reference receipt number:

Amount

AMOUNTS
Gross amount: -827.25
Net amount: -775.00
Net price: -760.00
Payment amount: 0.00

Cost amount: 0.00
Tax included in price: No

SALES ORDER/INVOICE AMOUNT
Sales order amount: 0.00
Sales invoice amount: 0.00
Sales order:

DISCOUNT
Discount amount: 0.00
Discount amount excluding tax: 0.00
Total discount amount: 0.00
Rounded amount: 0.00

DIFFERENCE
Sales/payment difference: 827.25

PICKING PRODUCTS UP IN A RETAIL STORE WITH POINT OF SALE

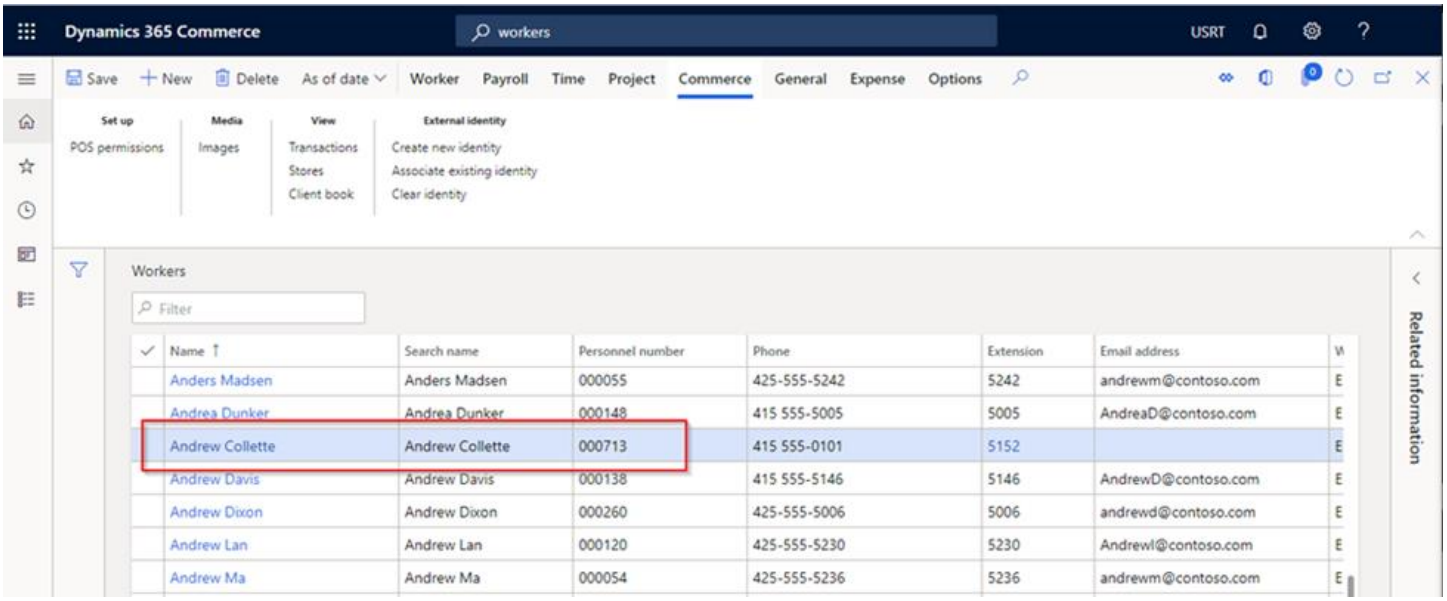
We can now launch the San Francisco point of sale and demonstrate a customer pickup.

Before you log into cloud point of sale (POS), you need to associate the AAD account you are using with the worker you will be logging in with. We will mostly be using the San Francisco store, which is the most up to date configured retail channel that has the same set of products configured as the Fabrikam extended online channel. There is also a pre-configured user for this channel (000713 - Andrew Collette).

ASSOCIATE AAD USER TO WORKER

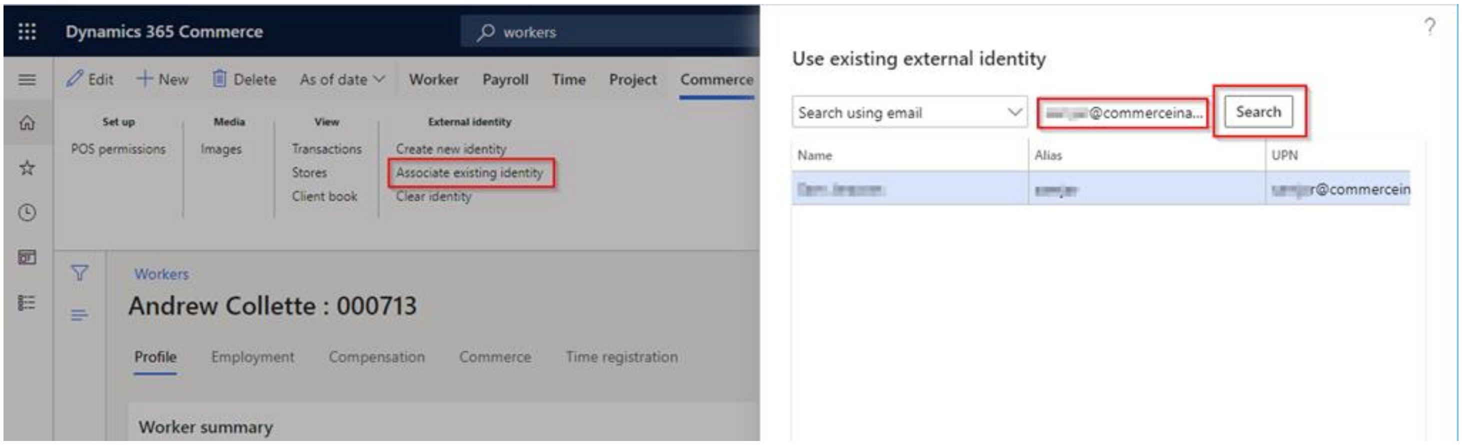
Follow the steps below to associate the AAD account to user 000713.

- Launch the Headquarters tool and search for the "**workers"** page
- Select the user "**Andrew Collette**", this user with id **000713** is configured for the retail "San Francisco" channel.



✓	Name ↑	Search name	Personnel number	Phone	Extension	Email address	W
	Anders Madsen	Anders Madsen	000055	425-555-5242	5242	andrewm@contoso.com	E
	Andrea Dunker	Andrea Dunker	000148	415 555-5005	5005	AndreaD@contoso.com	E
	Andrew Collette	Andrew Collette	000713	415 555-0101	5152		E
	Andrew Davis	Andrew Davis	000138	415 555-5146	5146	AndrewD@contoso.com	E
	Andrew Dixon	Andrew Dixon	000260	425-555-5006	5006	andrewd@contoso.com	E
	Andrew Lan	Andrew Lan	000120	425-555-5230	5230	Andrewl@contoso.com	E
	Andrew Ma	Andrew Ma	000054	425-555-5236	5236	andrewm@contoso.com	E

- Click on the "**Associate existing identity**" and enter in the AAD account you are using for your environment followed by clicking the "**Search**" button, then the "**OK**" button.



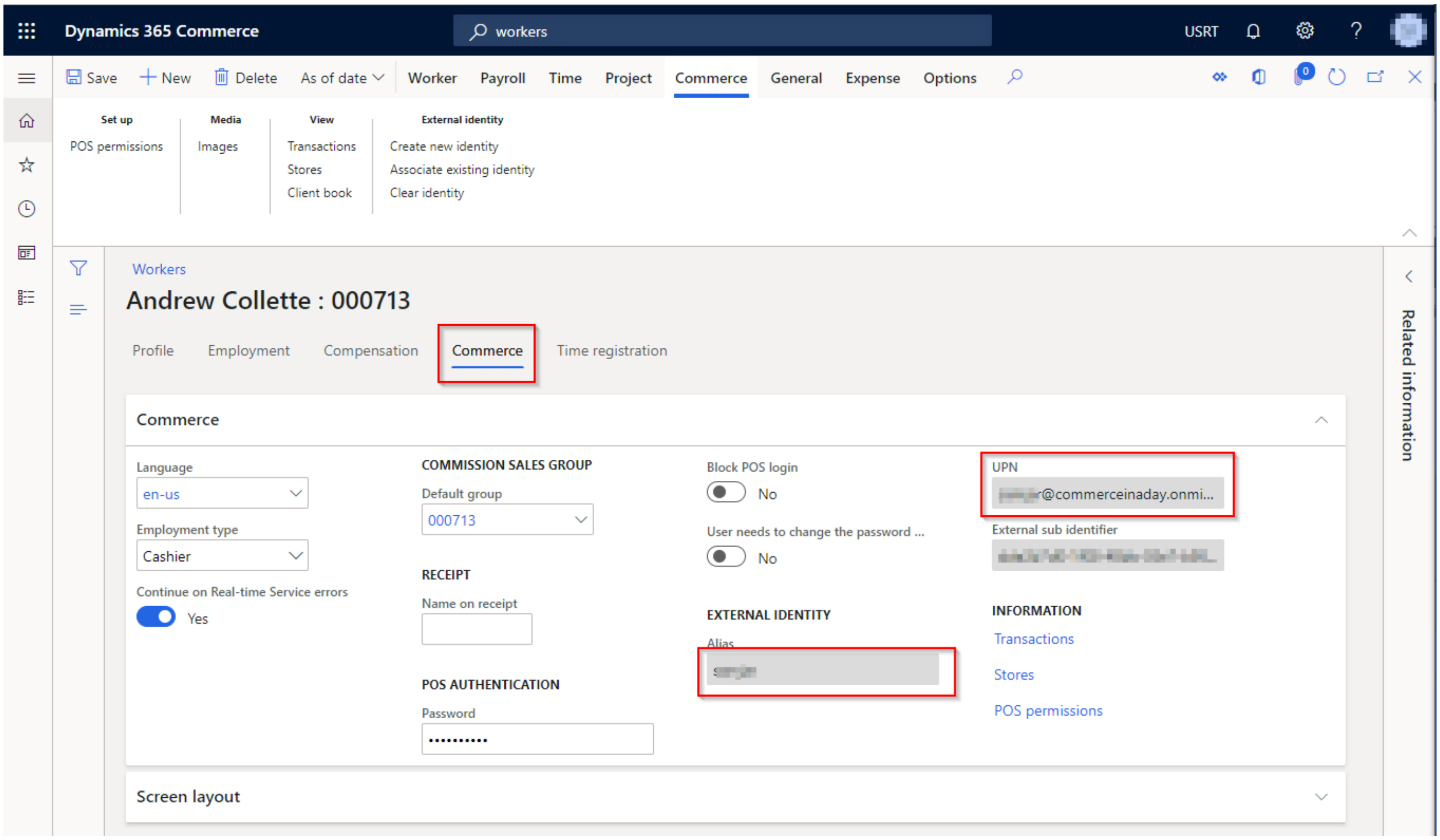
Andrew Collette : 000713

Use existing external identity

Search using email: [redacted]@commerceina...

Name	Alias	UPN
[redacted]	[redacted]	[redacted]r@commercein

If you now select the Commerce tab you should see the AAD account and alias listed as shown below.



Andrew Collette : 000713

Commerce

Language: en-us

Employment type: Cashier

Continue on Real-time Service errors: Yes

COMMISSION SALES GROUP

Default group: 000713

RECEIPT

Name on receipt:

POS AUTHENTICATION

Password: ••••••••••

Block POS login: No

User needs to change the password ...: No

EXTERNAL IDENTITY

Alias: [redacted]

UPN: [redacted]r@commerceinaday.onmi...

External sub identifier: [redacted]

INFORMATION

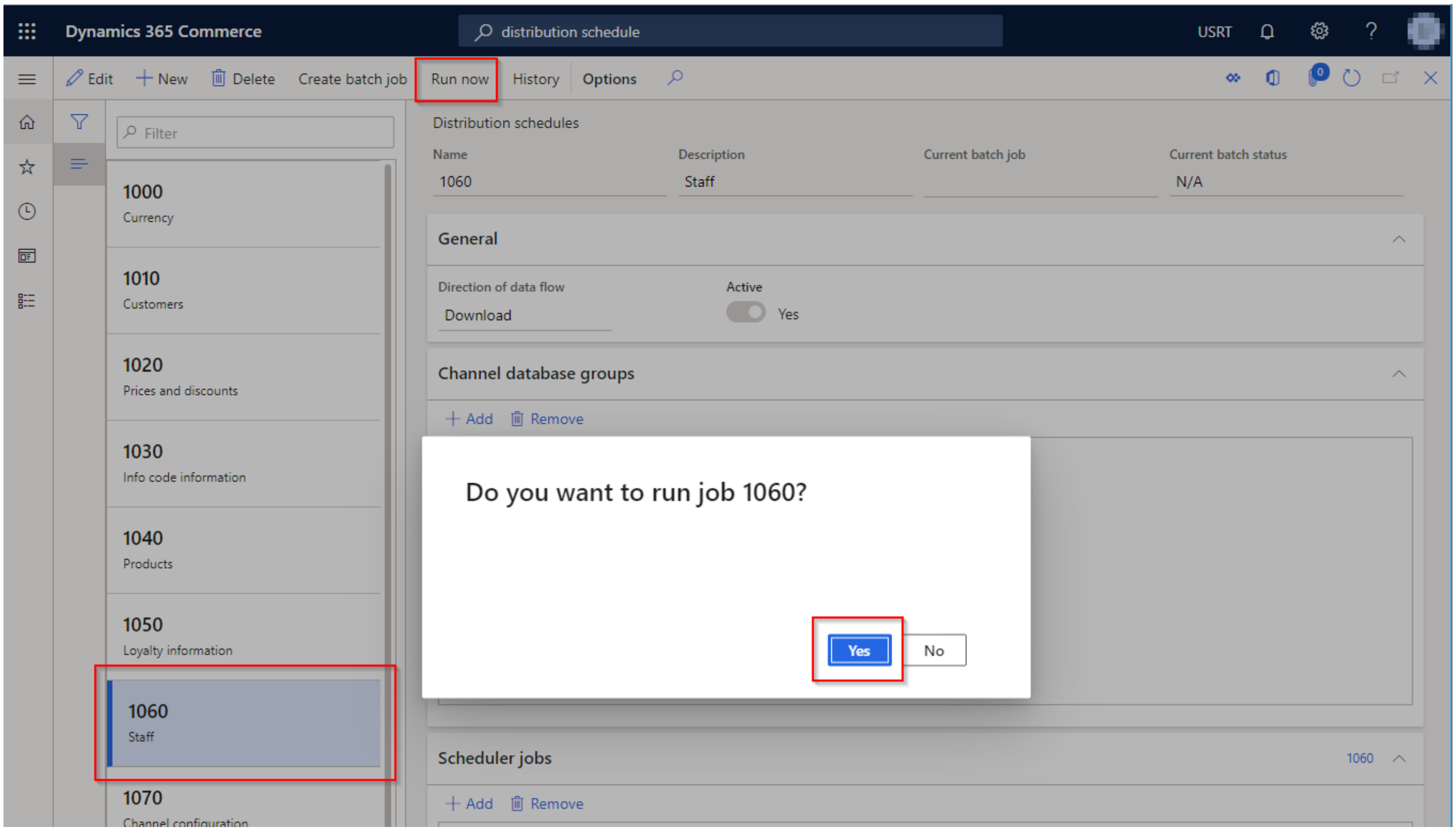
Transactions

Stores

POS permissions

The last thing needed it to run the "1060 – Staff" job.

- Search for the "**Distribution schedule**" page
- Select **1060 Staff** from the list and click the "Run now" button and select "**Yes**" to add the job to the queue.



Distribution schedules

Name	Description	Current batch job	Current batch status
1060	Staff		N/A

General

Direction of data flow: Download

Active: Yes

Channel database groups

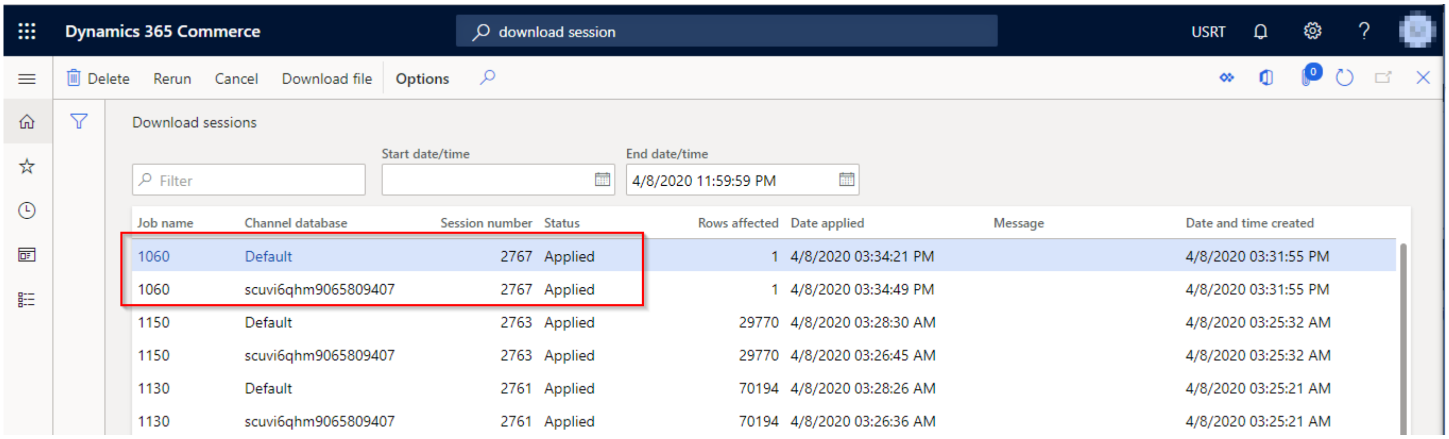
Scheduler jobs 1060

Do you want to run job 1060?

1000	Currency
1010	Customers
1020	Prices and discounts
1030	Info code information
1040	Products
1050	Loyalty information
1060	Staff
1070	Channel configuration

Ensure the job has completed before attempting to log into the POS.

- Search for the "**Download sessions**" page
- Ensure that the 1060 job is listed with status "Applied"



Download sessions

Start date/time:

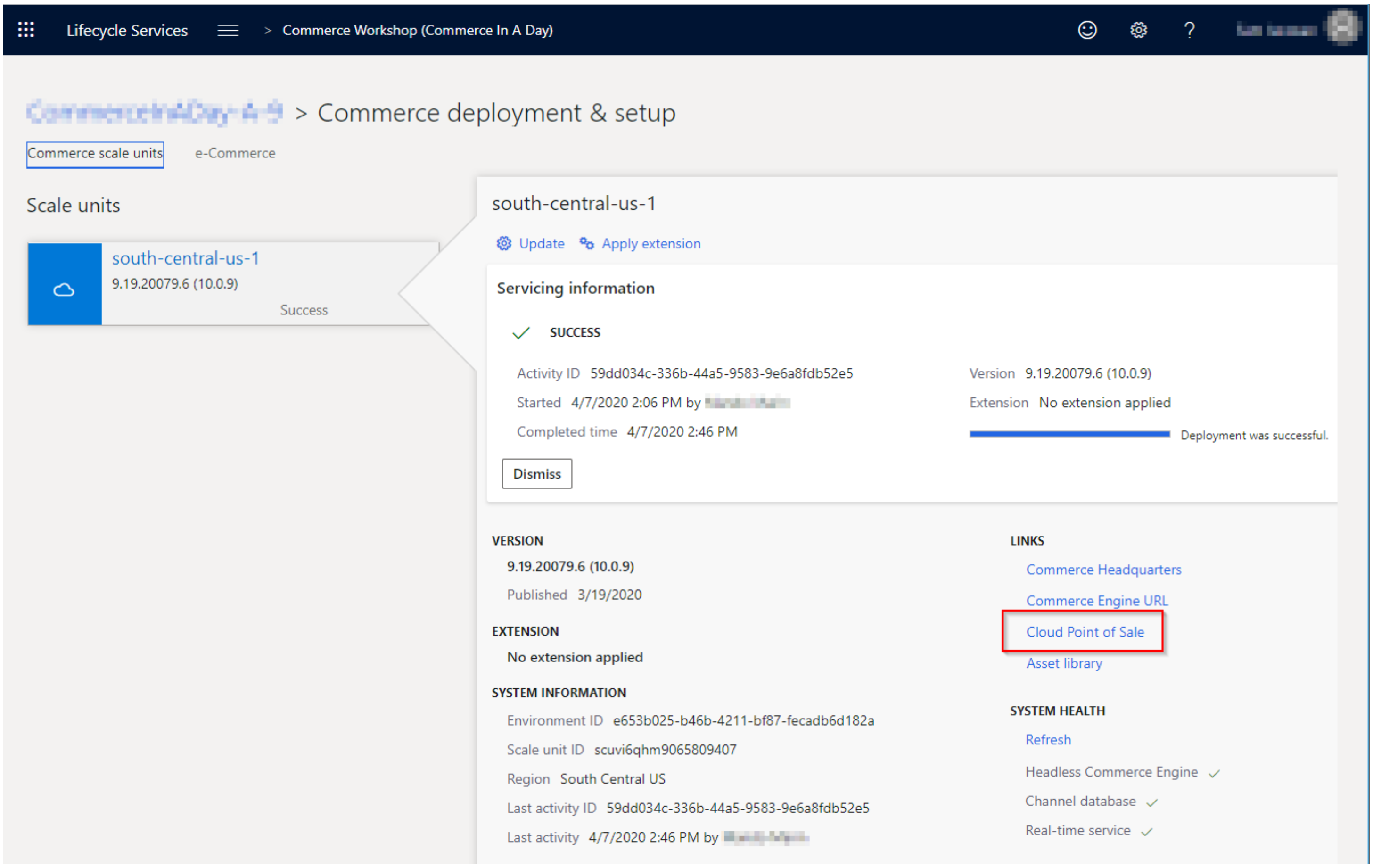
End date/time: 4/8/2020 11:59:59 PM

Job name	Channel database	Session number	Status	Rows affected	Date applied	Message	Date and time created
1060	Default	2767	Applied	1	4/8/2020 03:34:21 PM		4/8/2020 03:31:55 PM
1060	scuvi6qhm9065809407	2767	Applied	1	4/8/2020 03:34:49 PM		4/8/2020 03:31:55 PM
1150	Default	2763	Applied	29770	4/8/2020 03:28:30 AM		4/8/2020 03:25:32 AM
1150	scuvi6qhm9065809407	2763	Applied	29770	4/8/2020 03:26:45 AM		4/8/2020 03:25:32 AM
1130	Default	2761	Applied	70194	4/8/2020 03:28:26 AM		4/8/2020 03:25:21 AM
1130	scuvi6qhm9065809407	2761	Applied	70194	4/8/2020 03:26:36 AM		4/8/2020 03:25:21 AM

LOGGING INTO THE CLOUD POINT OF SALE

We are now ready to log into the point of sale. From LCS you can find the link to the point of sale as shown below.

Please note: If you cannot log onto the POS with Chrome please use Edge.



[redacted] > Commerce deployment & setup

Scale units

south-central-us-1
9.19.20079.6 (10.0.9)
Success

south-central-us-1

Servicing information

✓ SUCCESS

Activity ID 59dd034c-336b-44a5-9583-9e6a8fdb52e5
Started 4/7/2020 2:06 PM by [redacted]
Completed time 4/7/2020 2:46 PM

Version 9.19.20079.6 (10.0.9)
Extension No extension applied
Deployment was successful.

VERSION
9.19.20079.6 (10.0.9)
Published 3/19/2020

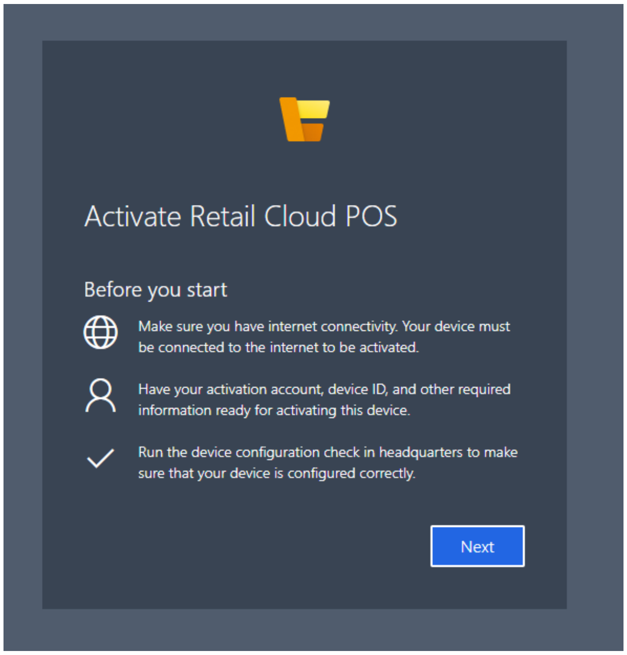
EXTENSION
No extension applied

SYSTEM INFORMATION
Environment ID e653b025-b46b-4211-bf87-fecadb6d182a
Scale unit ID scuvi6qhm9065809407
Region South Central US
Last activity ID 59dd034c-336b-44a5-9583-9e6a8fdb52e5
Last activity 4/7/2020 2:46 PM by [redacted]

LINKS
Commerce Headquarters
Commerce Engine URL
Cloud Point of Sale
Asset library

SYSTEM HEALTH
Refresh
Headless Commerce Engine ✓
Channel database ✓
Real-time service ✓

Once you click on the “Cloud Point of Sale” link, you should see a similar dialog box as shown below, click the “Next” button.



Activate Retail Cloud POS

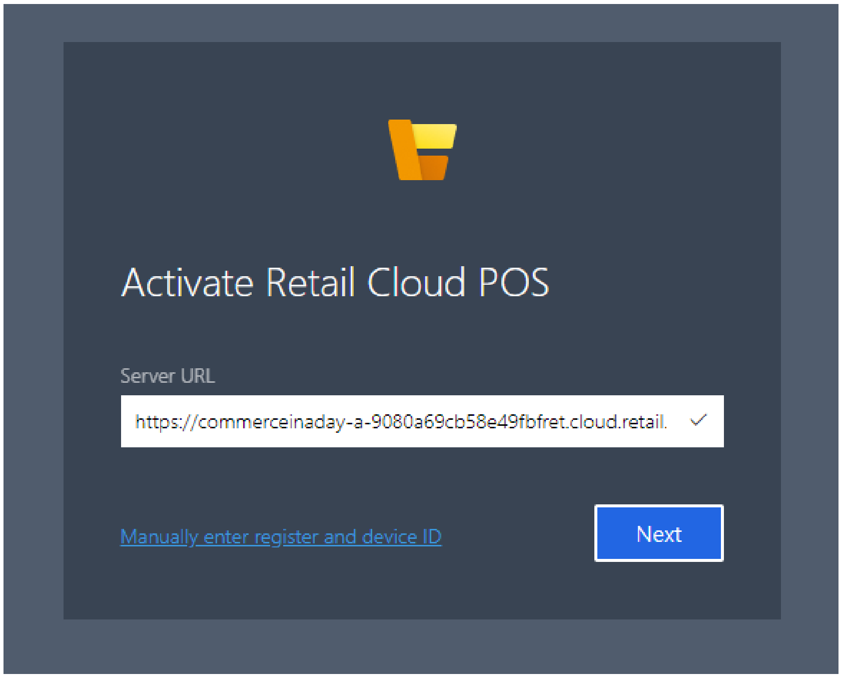
Before you start

Make sure you have internet connectivity. Your device must be connected to the internet to be activated.

Have your activation account, device ID, and other required information ready for activating this device.

Run the device configuration check in headquarters to make sure that your device is configured correctly.

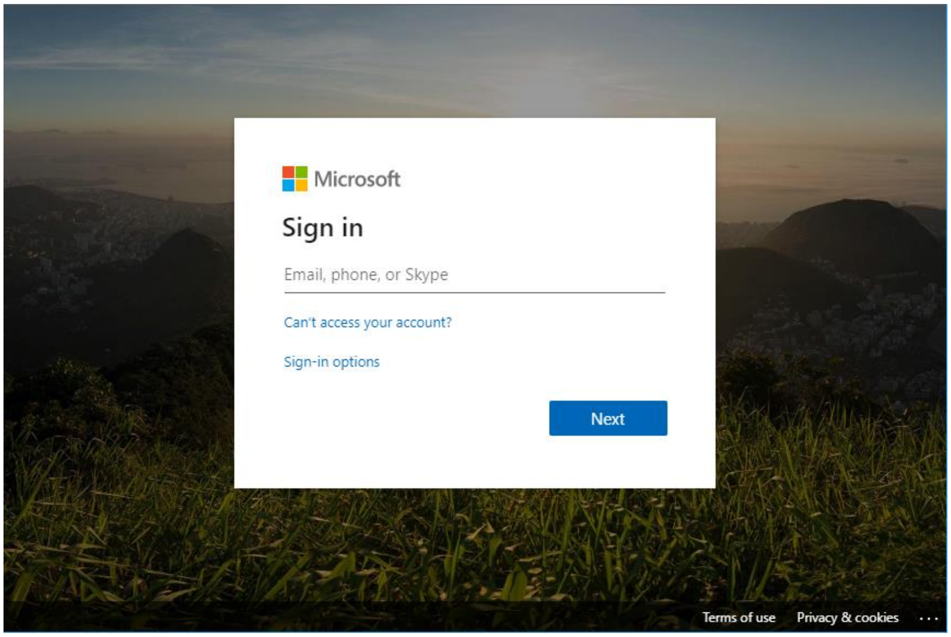
Click the “Next” button when you see the dialog box below.



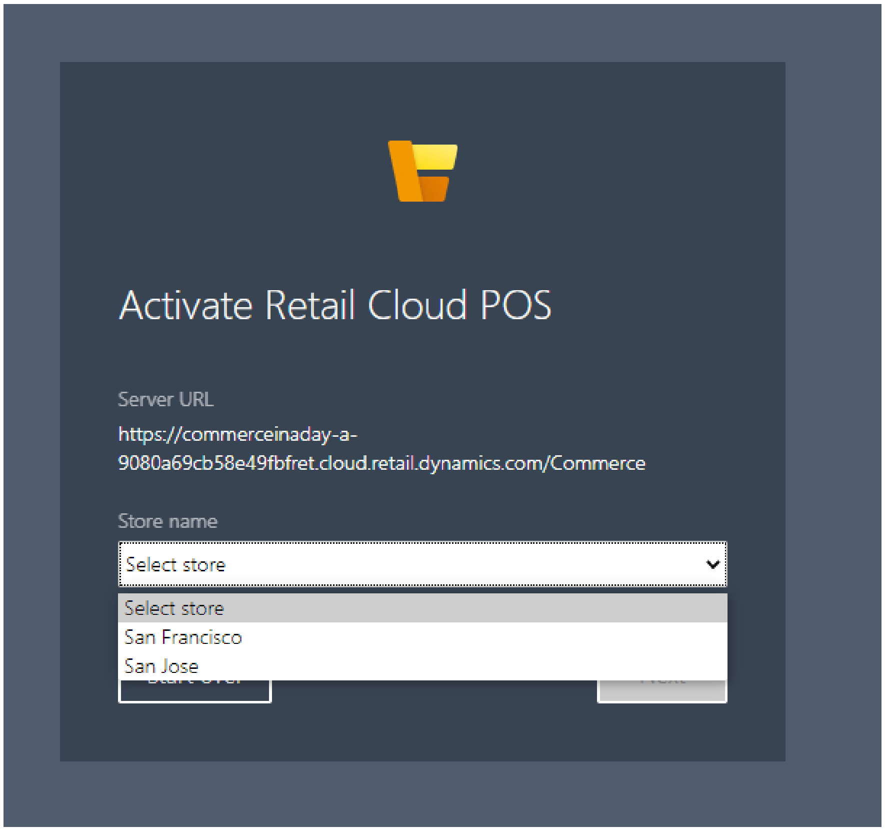
Activate Retail Cloud POS

Server URL: https://commerceinaday-a-9080a69cb58e49fbfret.cloud.retail.

You can now sign in with your AD account that was configured against the 000713 user in previous section.


Microsoft
Sign in
Email, phone, or Skype
Can't access your account?
Sign-in options
Next
Terms of use
Privacy & cookies
. . .

Next, you will be asked to select a store. Since this AD account was associated with a specific user the stores that the user has access to are listed. Select San Francisco followed by clicking the "Next" button.



Activate Retail Cloud POS

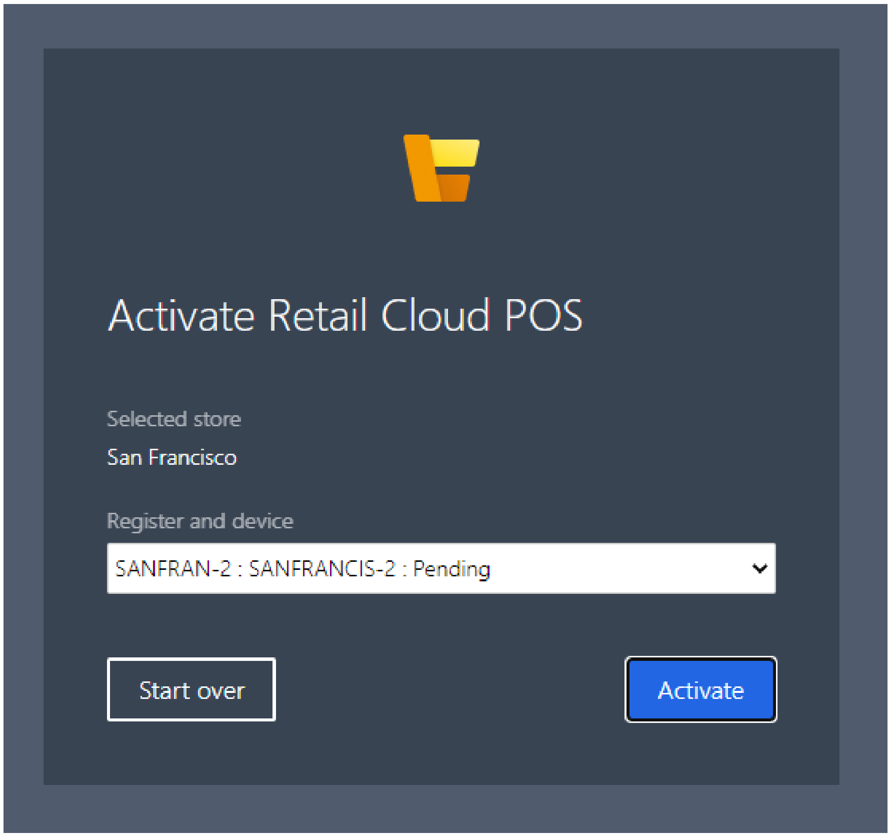
Server URL

https://commerceinaday-a-9080a69cb58e49fbfret.cloud.retail.dynamics.com/Commerce

Store name

Select store

Finally, you will be asked which register and device to log into. You can pick any of them, followed by the "Activate" button.

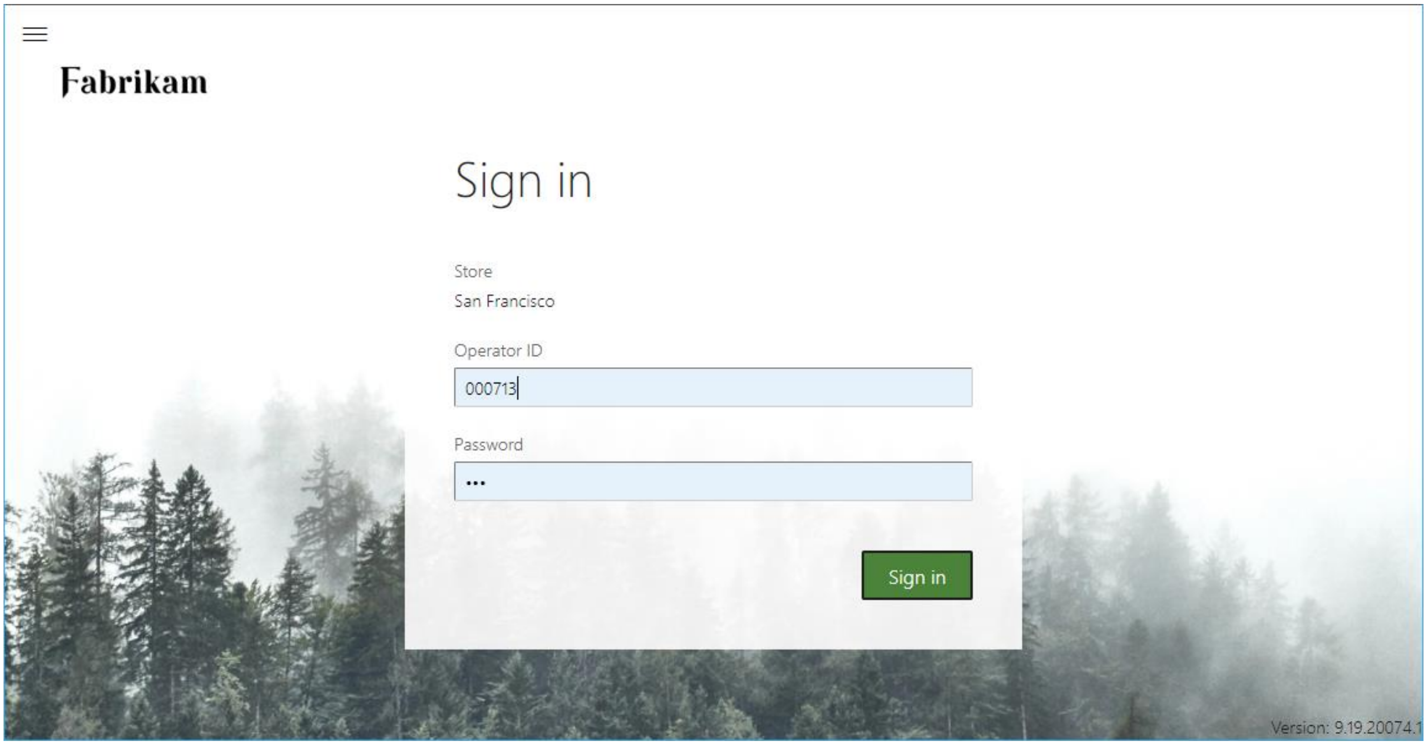


Activate Retail Cloud POS

Selected store
San Francisco

Register and device
SANFRAN-2 : SANFRANCIS-2 : Pending

You can now sign into the POS with the 000713 account, the default password set up in demo data is “123” but you can change this in the Headquarters app if desired. Notice how it is branded as Fabrikam, we won’t be covering rebranding the POS in this workshop.



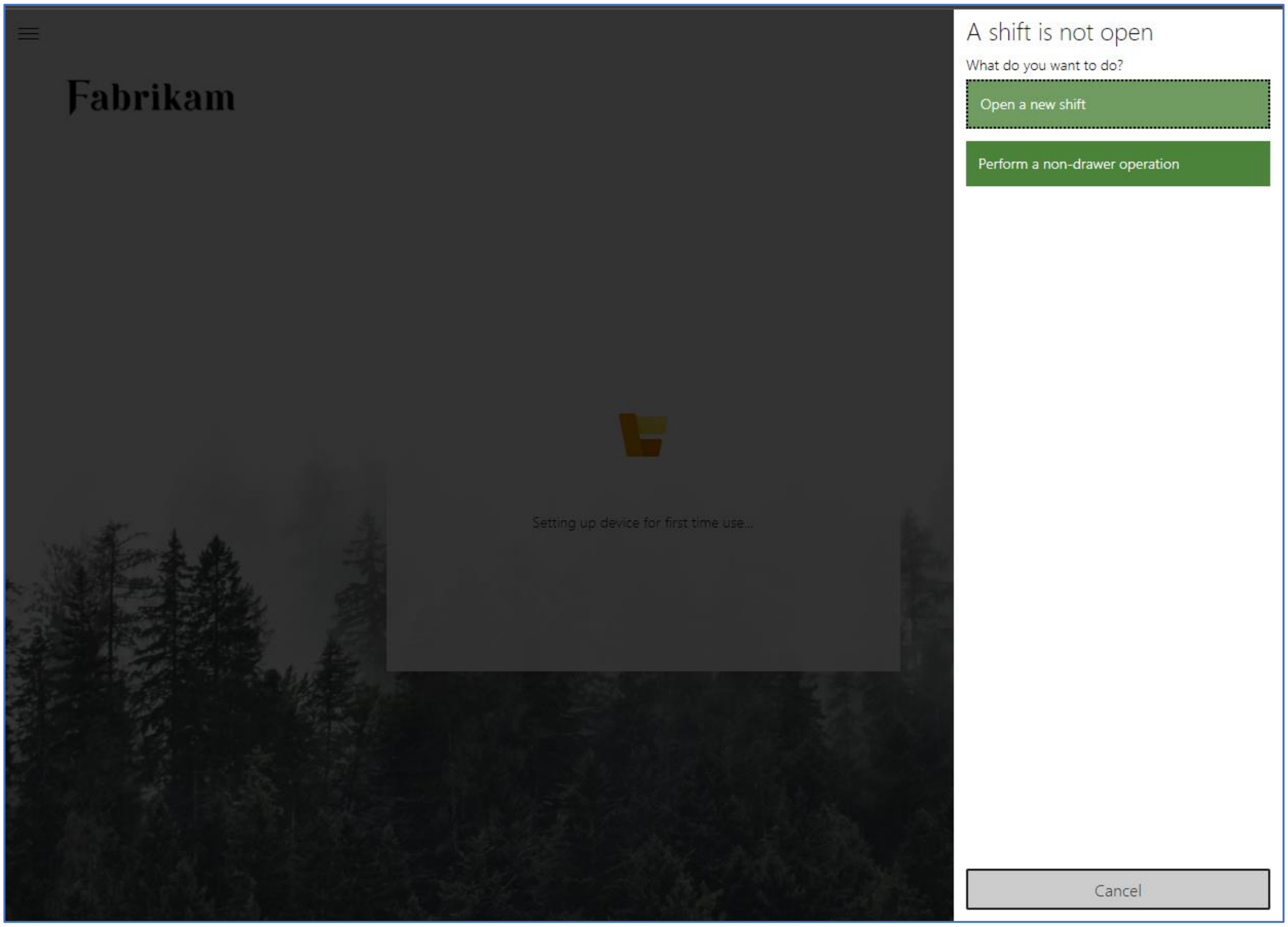
Sign in

Store
San Francisco

Operator ID
000713

Password
•••

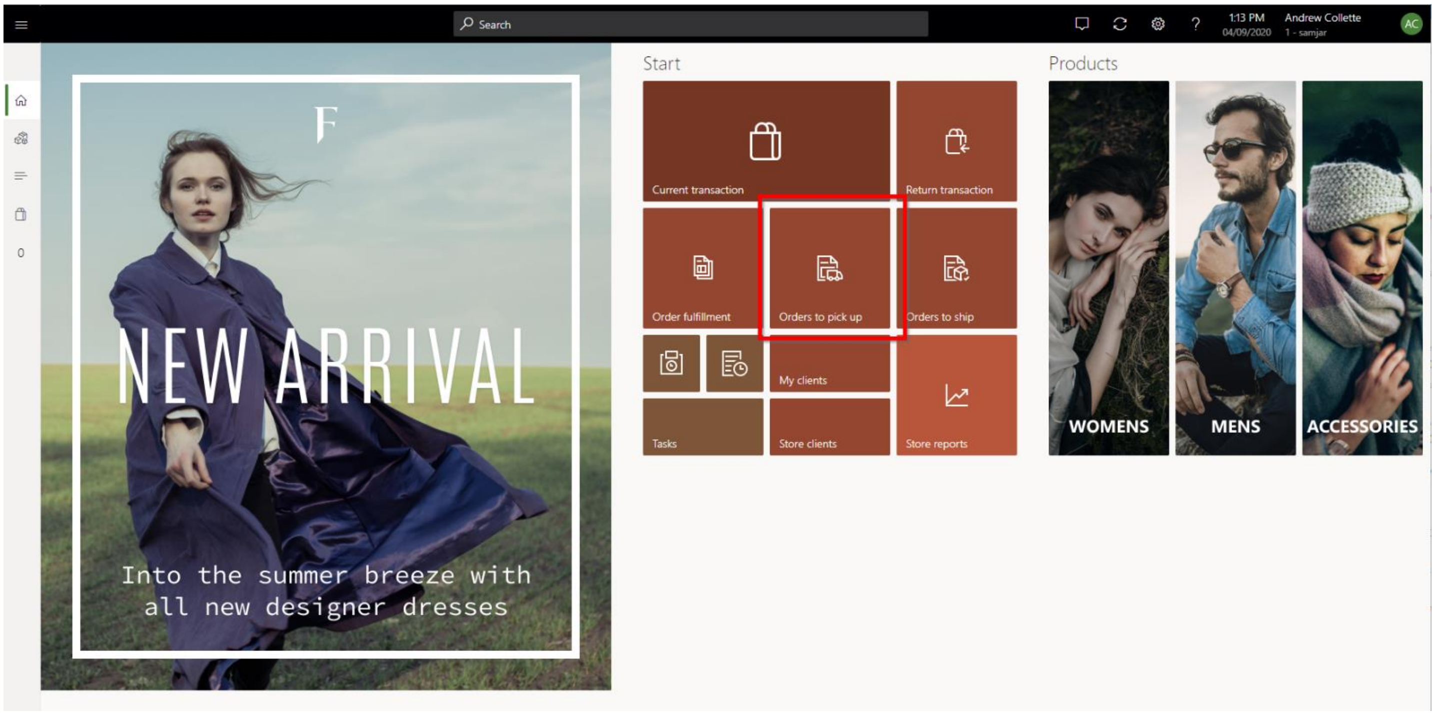
You can select “Open a new shift” and then pick “Main drawer”.


A shift is not open

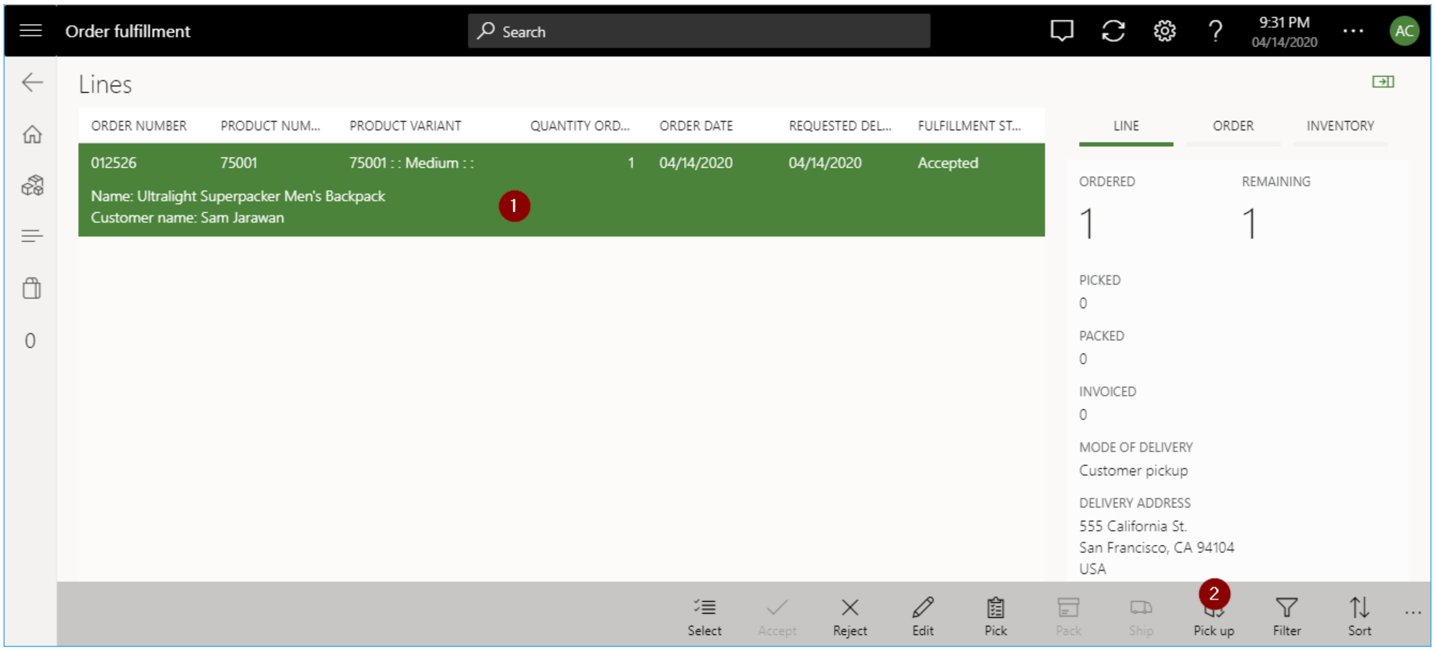
What do you want to do?

Fabrikam
Setting up device for first time use...
Open a new shift
Perform a non-drawer operation
Cancel

The POS is now ready to use, select “Orders to pick up”:



Select the order to pick up (1) and then click the “Pick up” button (2) at the bottom.



Lines

ORDER NUMBER	PRODUCT NUM...	PRODUCT VARIANT	QUANTITY ORD...	ORDER DATE	REQUESTED DEL...	FULFILLMENT ST...
012526	75001	75001 : : Medium : :	1	04/14/2020	04/14/2020	Accepted

Name: Ultralight Superpacker Men's Backpack
Customer name: Sam Jarawan

ORDERED
1

REMAINING
1

PICKED
0

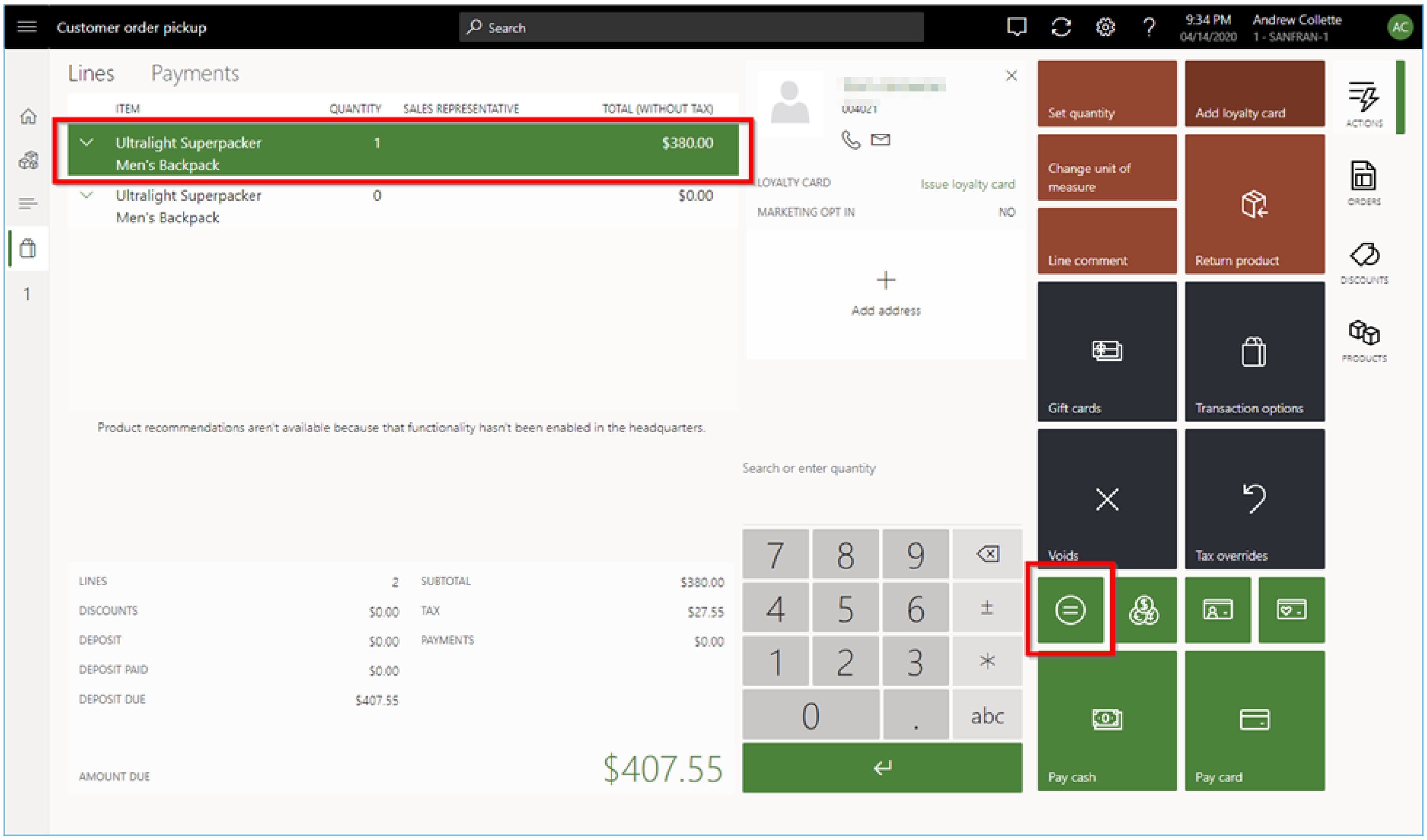
PACKED
0

INVOICED
0

MODE OF DELIVERY
Customer pickup

DELIVERY ADDRESS
555 California St.
San Francisco, CA 94104
USA

Select the item to be picked up and select the “Payment methods button” to complete the transaction.



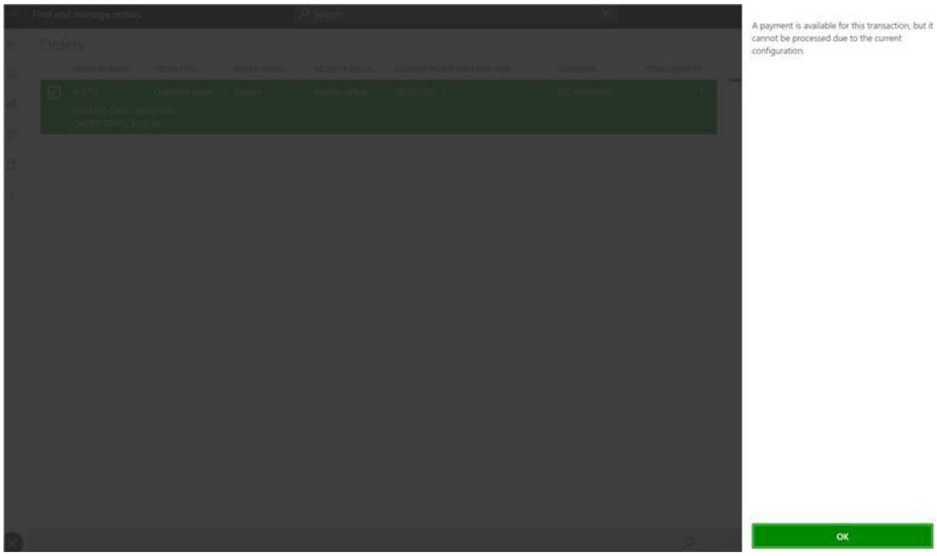
ITEM	QUANTITY	SALES REPRESENTATIVE	TOTAL (WITHOUT TAX)
Ultralight Superpacker Men's Backpack	1		$380.00
Ultralight Superpacker Men's Backpack	0		$0.00

Product recommendations aren't available because that functionality hasn't been enabled in the headquarters.

LINES	2	SUBTOTAL	$380.00
DISCOUNTS	$0.00	TAX	$27.55
DEPOSIT	$0.00	PAYMENTS	$0.00
DEPOSIT PAID	$0.00		
DEPOSIT DUE	$407.55		
AMOUNT DUE			$407.55

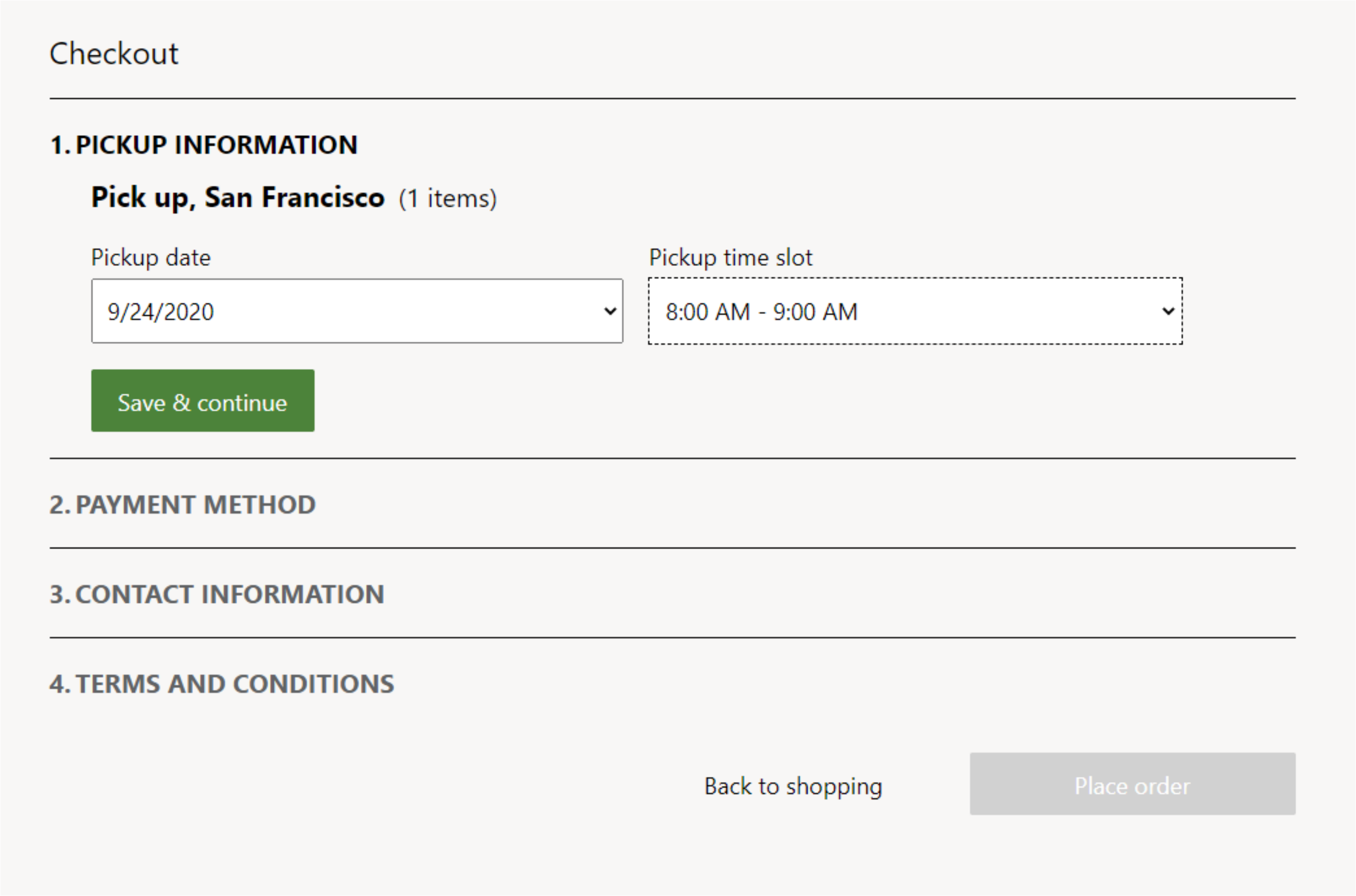
This completes the full End 2 End transaction.

Note: If you run into issues at the POS and receive an error message like below you need to change the setting on the register you’re using in HQ for “Card not present processing” to retail server.


A payment is available for this transaction, but it cannot be processed due to the current configuration.

UNIT 2 OPTIONAL – TIME SLOTS FOR CUSTOMER PICKUPS

The time slot feature gives retailers a way to define a time slot for items that the customer pickup delivery mode is turned on for. Time slots let retailers define the days and times when orders can be picked up from a store. Retailers can also define the number of orders that can be picked up during a given period. In this way, retailers can limit the number of orders that can be picked up on a given day and at a given time



Checkout

1. PICKUP INFORMATION

Pick up, San Francisco (1 items)

Pickup date: 9/24/2020

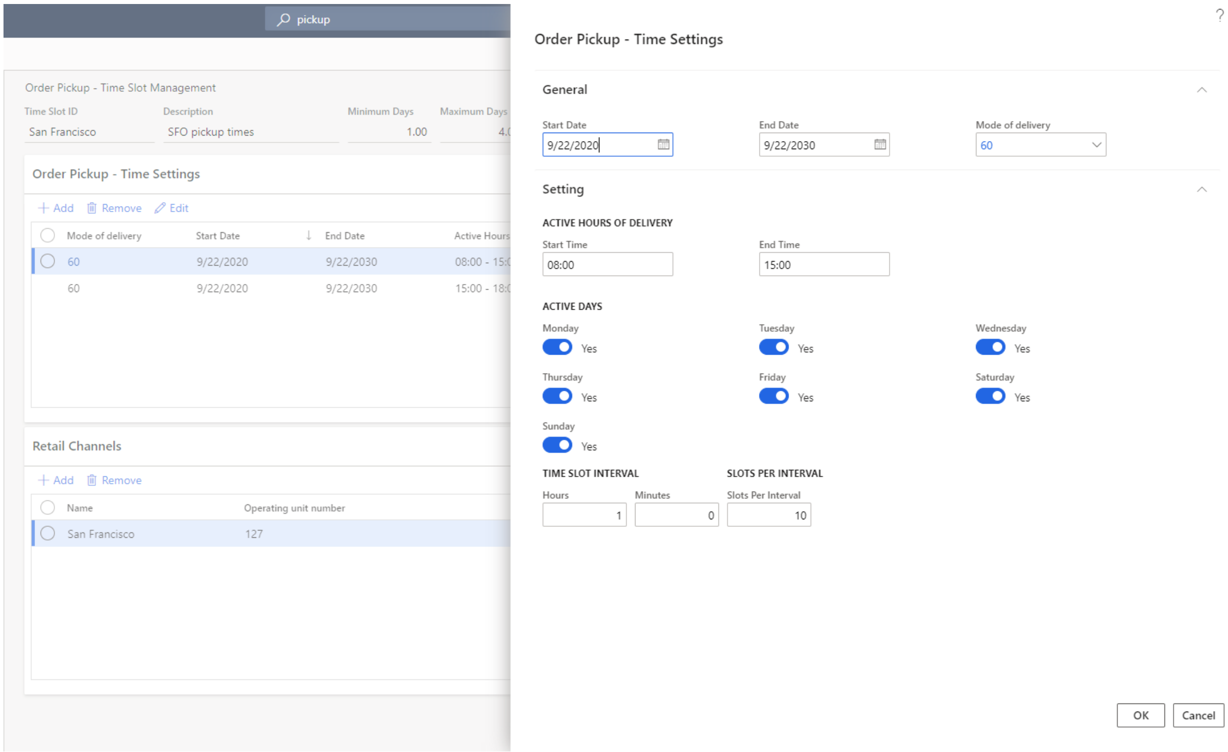
Pickup time slot: 8:00 AM - 9:00 AM

To configure the time slot feature in Commerce headquarters, follow these steps.

1. Go to **Commerce** > **Channel setup** > **Store pickup time slot**.
2. Select **New** to create a new time slot template. To use an existing template, select the template in the left pane.
3. Enter values in the **Time Slot ID** and **Description** fields.
4. On the **Order Pickup - Time Settings** FastTab, select **Add**.
5. In the **Order Pickup - Time Settings** dialog box, define the date range, mode of delivery, active hours of delivery, active days, time slot interval, slots per interval, and other settings.

   If time slots will be static for the foreseeable future, set the **End Date** field to **Never**.

Note: You can create multiple templates, but only one template can be associated with a single channel or store.



Order Pickup - Time Slot Management

Time Slot ID	Description	Minimum Days	Maximum Days
San Francisco	SFO pickup times	1.00	4.0

Order Pickup - Time Settings

Mode of delivery	Start Date	End Date	Active Hours
60	9/22/2020	9/22/2030	08:00 - 15:0
60	9/22/2020	9/22/2030	15:00 - 18:0

Retail Channels

Name	Operating unit number
San Francisco	127

Order Pickup - Time Settings

General

Start Date: 9/22/2020
End Date: 9/22/2030
Mode of delivery: 60

Setting

ACTIVE HOURS OF DELIVERY

Start Time: 08:00
End Time: 15:00

ACTIVE DAYS

Monday: Yes
Tuesday: Yes
Wednesday: Yes
Thursday: Yes
Friday: Yes
Saturday: Yes
Sunday: Yes

TIME SLOT INTERVAL

Hours: 1
Minutes: 0

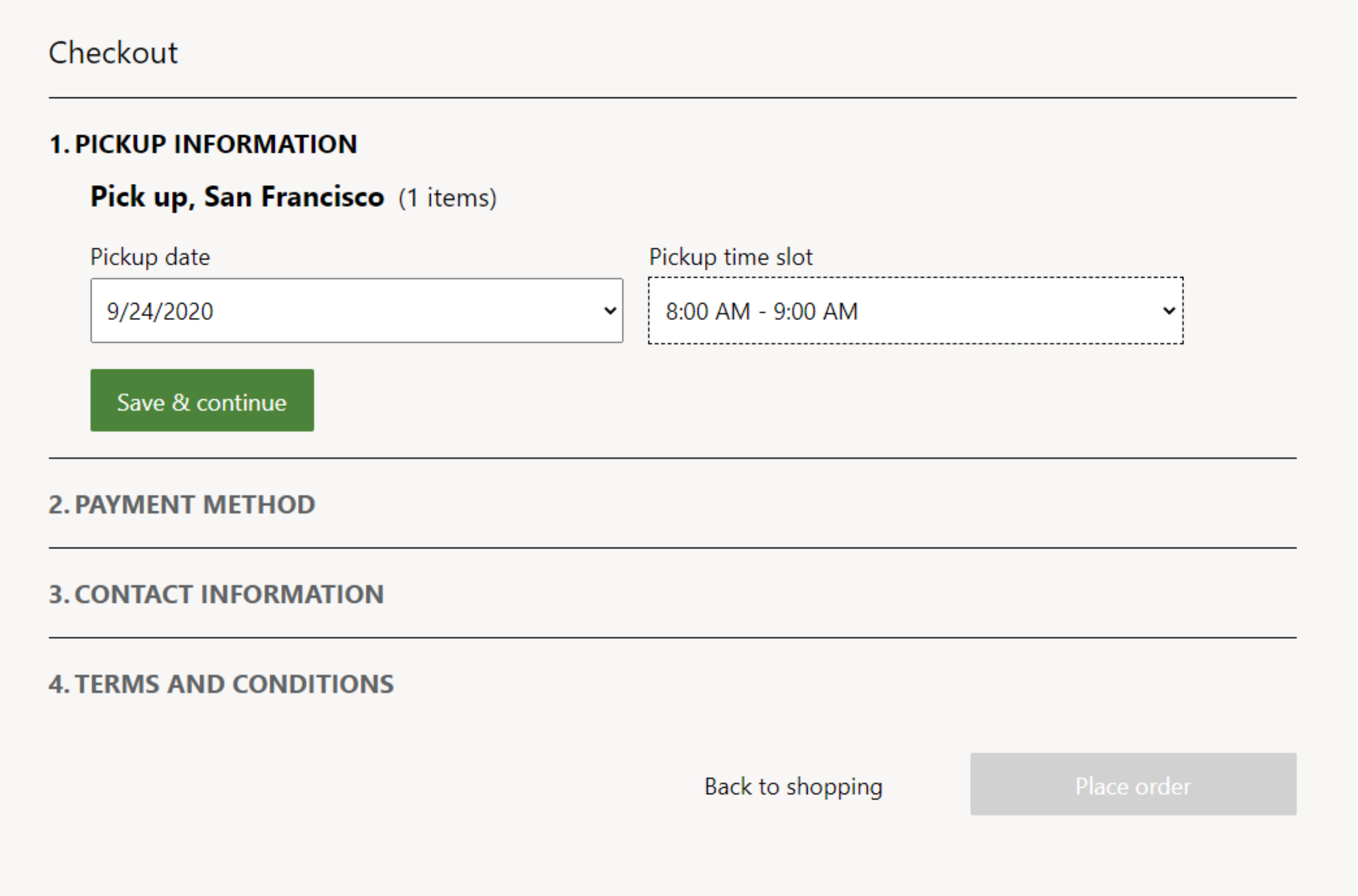
SLOTS PER INTERVAL

Slots Per Interval: 10

6. When you've finished, select **OK**.
7. If the time slots in a day will vary, create additional entries on the **Order Pickup - Time Settings** FastTab to ensure that the dates and times don't overlap.
8. On the **Retail Channels** FastTab, select **Add** to associate the time slot template with the stores or channels where it will be used.
9. In the **Choose organization nodes** dialog box, use the arrow buttons to select (or clear the selection of) the stores, regions, and organizations that the template should be associated with.
10. When you've finished, select **OK**.
11. On the **Distribution schedule** page, run the **1070** and **1135** jobs to sync the data to the channels.

PICKUP INFORMATION MODULE

Now you have setup time slots in Commerce headquarters for customer pickups you need to add the pickup information module to the checkout page.



Checkout

1. PICKUP INFORMATION

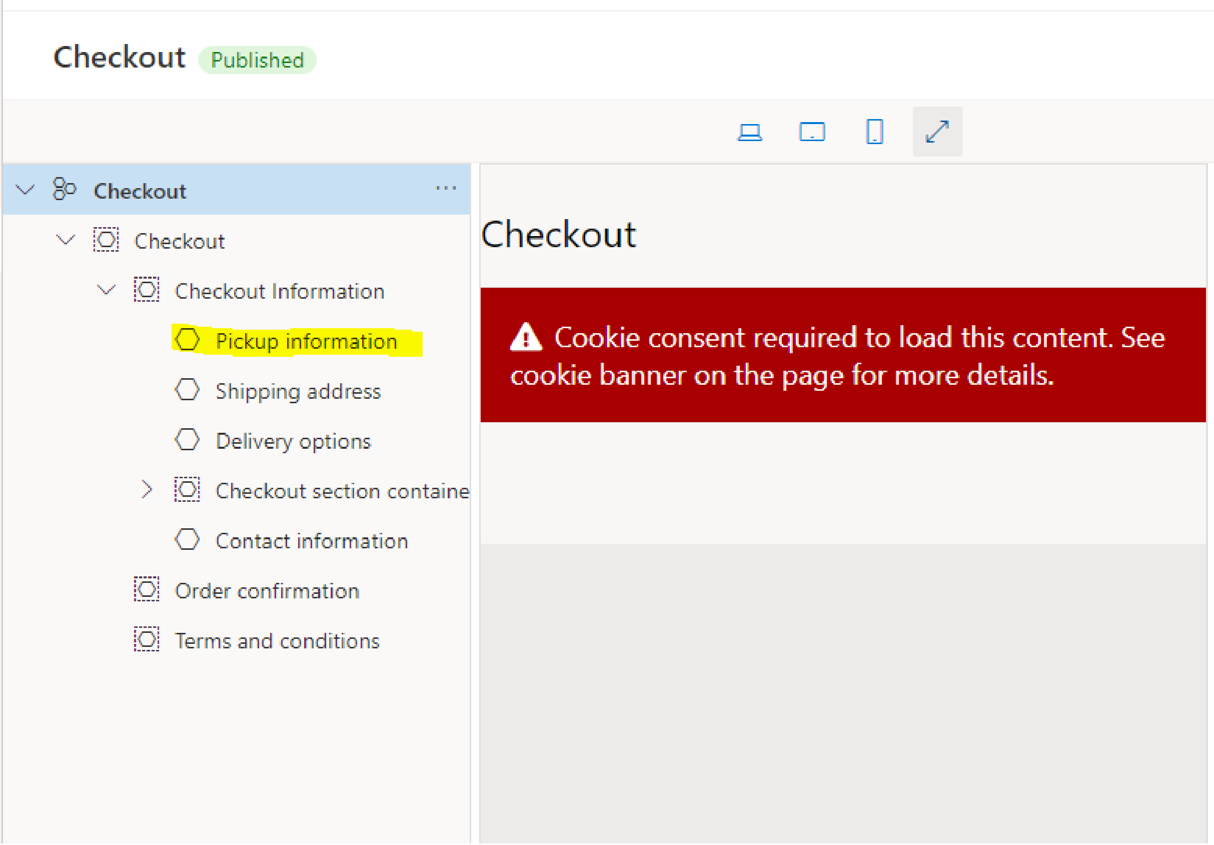
Pick up, San Francisco (1 items)

Pickup date: 9/24/2020

Pickup time slot: 8:00 AM - 9:00 AM

1. Go to Fragments and edit the “Checkout” fragment.
2. Add the “Pickup information module” as shown below.
3. Once added, save and Publish.

Note: Information about the selected time slot can be viewed in the order confirmation module and the order details module. Both these modules have a **Show timeslot information** property. Before they can show the selected time slot during the order process, this property must be set to **True.**



Checkout Published

Checkout

- Checkout
  - Checkout Information
    - Pickup information
    - Shipping address
    - Delivery options
    - Checkout section containe
    - Contact information
  - Order confirmation
  - Terms and conditions

Checkout

⚠ Cookie consent required to load this content. See cookie banner on the page for more details.

UNIT 1B E-COMMERCE BUSINESS TO BUSINESS (B2B)

Before we can use the B2B functionality we need to make sure the B2B eCommerce options have been enabled in Feature management.

The B2B e-commerce capabilities feature in Commerce headquarters enables organizations to onboard business partners and define administrator users. This feature also enables administrators to create and manage business partner users and teams, and to assign specific roles to them. Finally, it enables business partner users to create order templates and use existing orders to reorder products.

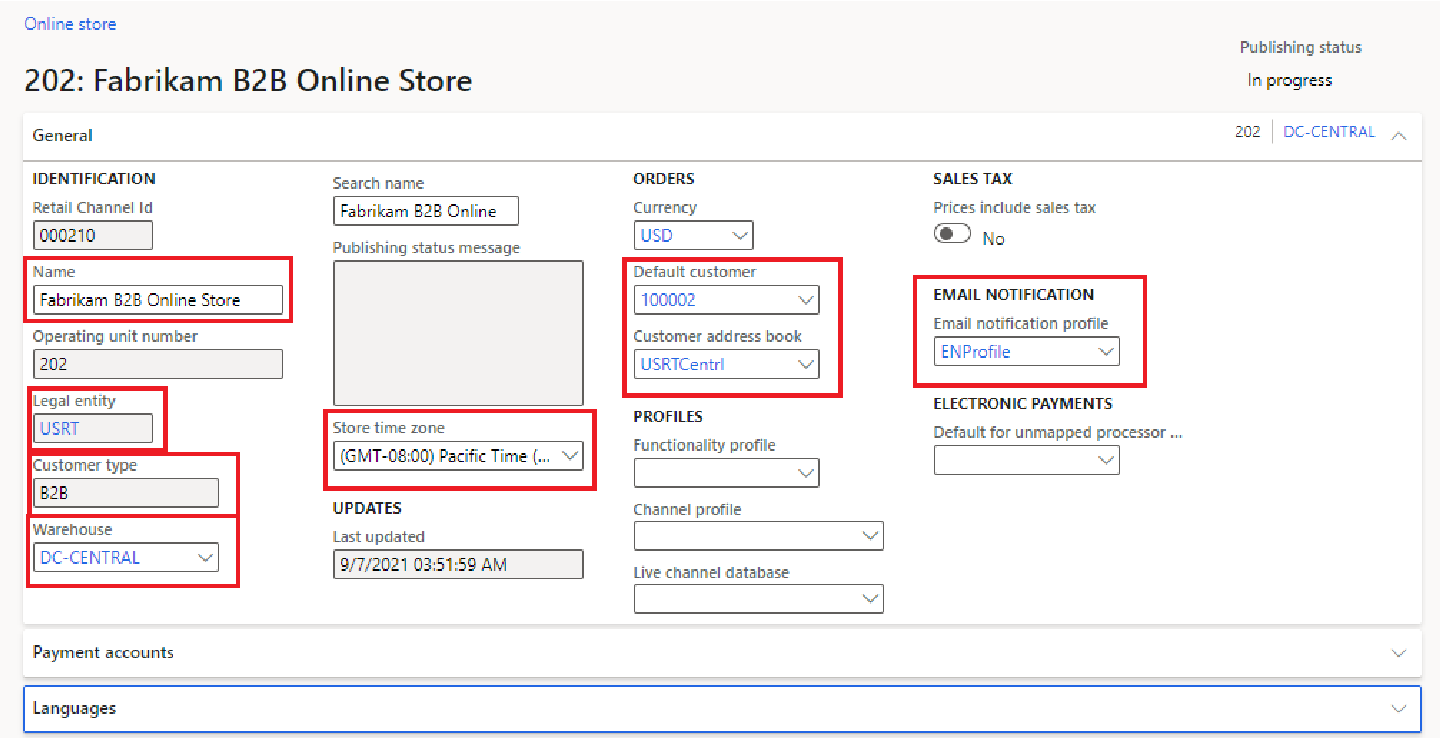
To turn on the B2B e-commerce capabilities feature in Commerce headquarters, follow these steps.

1. Go to **Workspaces > Feature Management**.
2. On the **All** tab, filter on the **Module** field by using the term **Retail and commerce**.
3. Find and select the feature that is named **Enable the use of B2B eCommerce capabilities**, and then select **Enable now**.

B2B ONLINE STORE

Configure the following items for the new channel:

- **Name** – Provide the name for the store (example: "Fabrikam B2B Online Store or Adventure Works B2B")
- **Legal entity** – Select the legal entity the store will belong to (this will be "**USRT**" in the evaluation environment)
- **Customer Type** – Make sure this is set to **B2B**
- **Warehouse** – Select the warehouse that will serve the online store (**DC-CENTRAL** can be used in the evaluation environment)
    - For basic warehouse setup see the online Warehouse set up document.
- **Store time zone** – Select the default time zone used for the store, this may map to the corporation's headquarters time zone
- **Default customer** – This customer may need to be pre-configured, all sales orders through the site for anonymous users will be tied to this account in AX (**100002** can be used in the evaluation environment)
    - For more info see the online Create a default customer doc.
    - **Note**: for any new e-commerce customer account signups the default properties are copied from this account
    - **Note**: If you are creating a new default customer, ensure that the default customer is of a type "Person" and not "Organization"
- **Customer address book** – Set the address book, this should be pre-configured before creating the channel (**USRTCentrl** can be used from within the evaluation environment)
    - For more info see the online Create a default customer doc which covers creating an address book.
    - **Note**: The default address book must be associated with the default customer provided above
- **Email notifications** – Select the email notification profile, this may need to be pre-created before creating the channel (**ENProfile** can be used if you are using the evaluation environment)
    - For more information see the online Set up an email notification profile document.
- Select the "**Save**" button.



Online store

Publishing status
In progress

202: Fabrikam B2B Online Store

General — 202 | DC-CENTRAL

IDENTIFICATION

Retail Channel Id: 000210

Name: Fabrikam B2B Online Store

Operating unit number: 202

Legal entity: USRT

Customer type: B2B

Warehouse: DC-CENTRAL

Search name: Fabrikam B2B Online

Publishing status message:

Store time zone: (GMT-08:00) Pacific Time (...

UPDATES

Last updated: 9/7/2021 03:51:59 AM

ORDERS

Currency: USD

Default customer: 100002

Customer address book: USRTCentrl

PROFILES

Functionality profile:

Channel profile:

Live channel database:

SALES TAX

Prices include sales tax: No

EMAIL NOTIFICATION

Email notification profile: ENProfile

ELECTRONIC PAYMENTS

Default for unmapped processor ...:

Payment accounts

Languages

SETUP PAYMENT ACCOUNTS

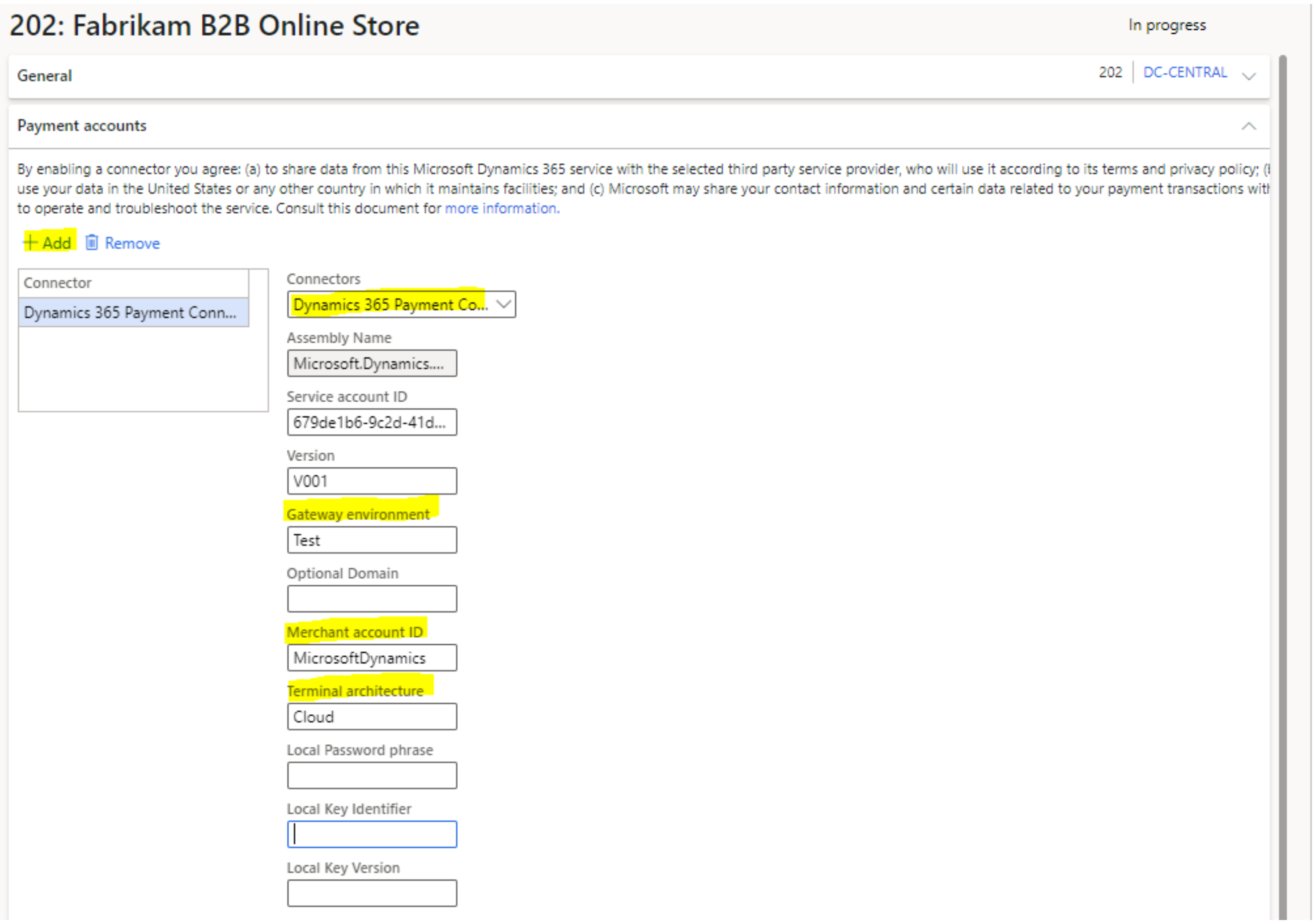
We will now setup the Adyen payment account needed to accept payments on the e-Commerce site. This section will guide you through setting up a payment connector for the Online site.

- Expand the "**Payment accounts**" section and select "**Add**" to add a payment connector.

Online channels are designed to be used out of the box with the **Dynamics 365 Payment Connector for Adyen.**

Configuration Steps **(Note: The Cloud API key will need to be supplied by your instructor for this training)**. If you plan to set this up in your own environment you will need an Adyen test account.

- Change "**Gateway Environment**" to '**Test'**
- Set the "**Merchant account ID**"
- Set "**Terminal Architecture**" to "**Cloud**"
- Change "**Allow saving payment information in e-Commerce**" to "**True"**
- All other values can be left with the defaults
- Click "**Save**"



202: Fabrikam B2B Online Store

In progress

General

202 | DC-CENTRAL

Payment accounts

By enabling a connector you agree: (a) to share data from this Microsoft Dynamics 365 service with the selected third party service provider, who will use it according to its terms and privacy policy; (
use your data in the United States or any other country in which it maintains facilities; and (c) Microsoft may share your contact information and certain data related to your payment transactions wit
to operate and troubleshoot the service. Consult this document for more information.

+ Add   Remove

Connector
Dynamics 365 Payment Conn...

Connectors: Dynamics 365 Payment Co...

Assembly Name: Microsoft.Dynamics....

Service account ID: 679de1b6-9c2d-41d...

Version: V001

Gateway environment: Test

Optional Domain:

Merchant account ID: MicrosoftDynamics

Terminal architecture: Cloud

Local Password phrase:

Local Key Identifier:

Local Key Version:

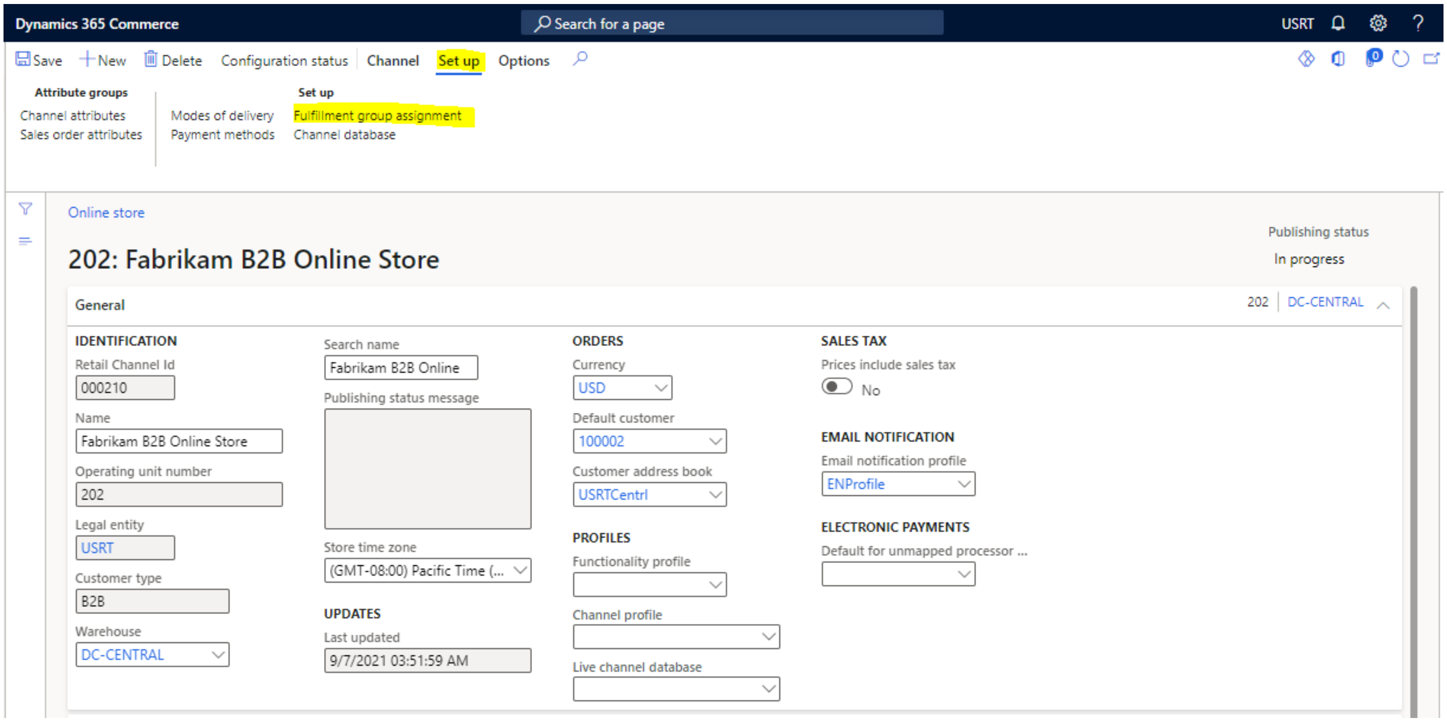
SETUP LANGUAGES

If you need to support multiple languages on your e-Commerce site, you first need to configure additional languages on the online store channel. For this workshop we will use en-us only, see appendix on how to add additional languages.

FULFILLMENT GROUP ASSIGNMENTS

In order for BOPIS (buy online, pickup in store) scenarios to work, you'll need to associate the channel with stores. Demo data already has a group of stores created called "All stores" that can be used here to ensure all stores retail stores show up when attempting the BOPIS scenario on the e-Commerce site. This section will show you how to set this up.

From within the selected online store, select the "**Setup**" tab at the top followed by the "**Fulfillment group assignment**" link.



202: Fabrikam B2B Online Store

Publishing status: In progress

General — 202 | DC-CENTRAL

IDENTIFICATION

Retail Channel Id: 000210

Name: Fabrikam B2B Online Store

Operating unit number: 202

Legal entity: USRT

Customer type: B2B

Warehouse: DC-CENTRAL

Search name: Fabrikam B2B Online

Publishing status message:

Store time zone: (GMT-08:00) Pacific Time (...

UPDATES

Last updated: 9/7/2021 03:51:59 AM

ORDERS

Currency: USD

Default customer: 100002

Customer address book: USRTCentrl

PROFILES

Functionality profile:

Channel profile:

Live channel database:

SALES TAX

Prices include sales tax: No

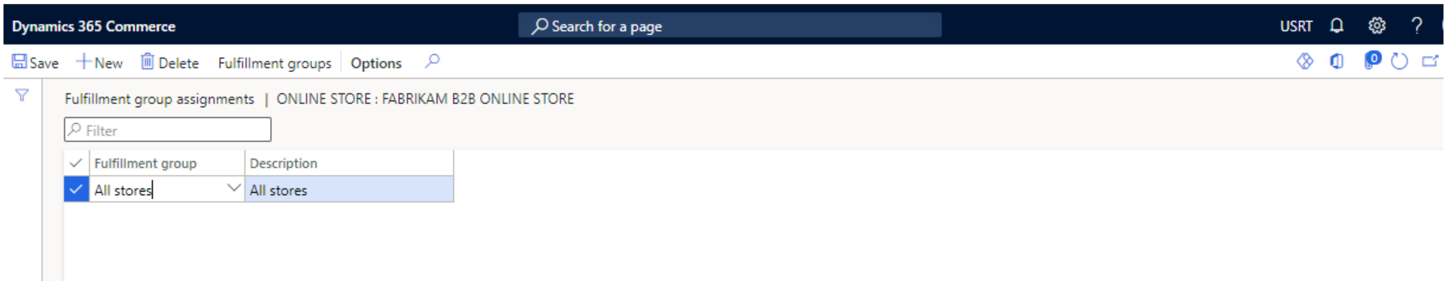
EMAIL NOTIFICATION

Email notification profile: ENProfile

ELECTRONIC PAYMENTS

Default for unmapped processor ...:

Add "**All stores**" and close the form to get back to the previous channel page. You can also pick explicit stores here if desired.



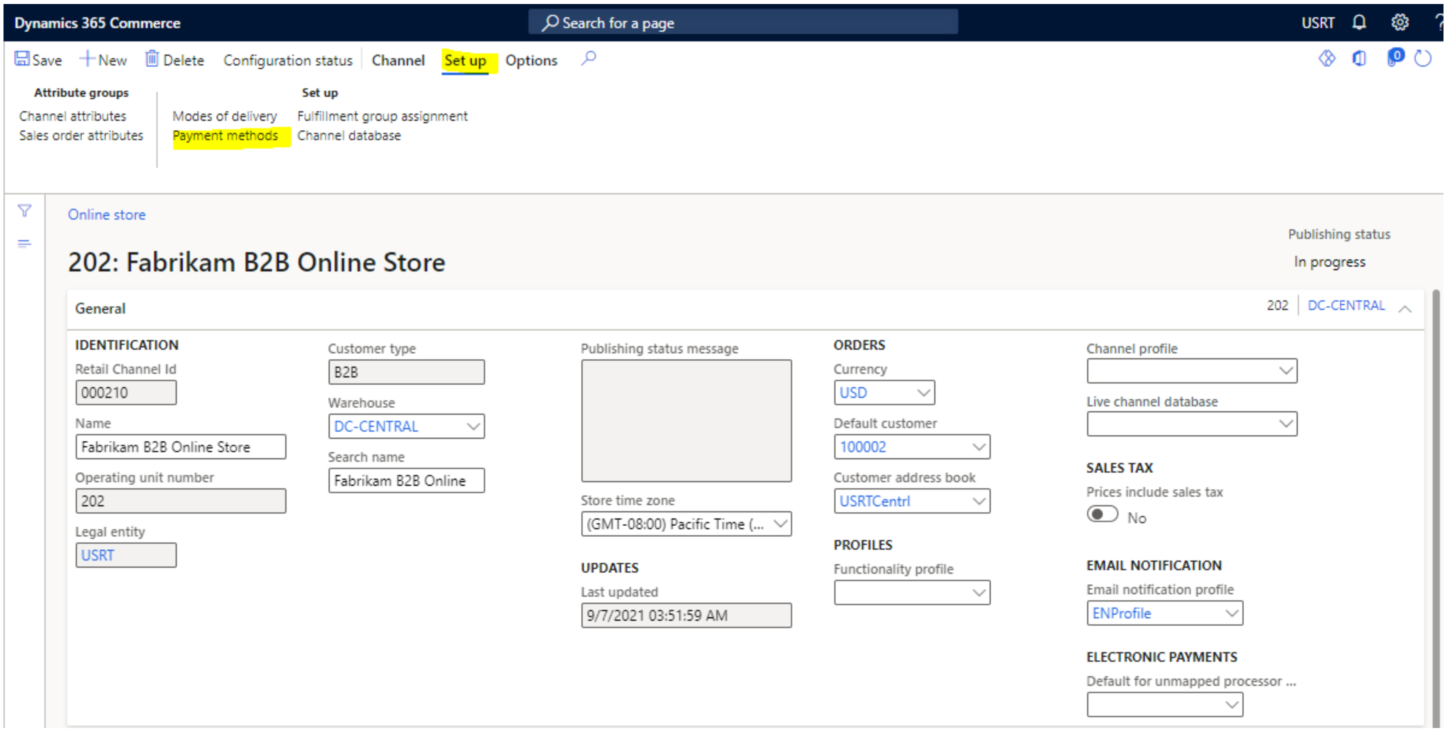
Fulfillment group assignments | ONLINE STORE : FABRIKAM B2B ONLINE STORE

✓	Fulfillment group	Description
✓	All stores	All stores

SETUP PAYMENT METHODS

Each channel can accept various payment methods. In this section we will configure the online store channel to accept credit cards and because B2B can accept on account payments we will set this up as well.

From within the selected online store channel, select the “**Payment method**” button under the “**Setup**” tab.



202: Fabrikam B2B Online Store

Publishing status: In progress

IDENTIFICATION

Retail Channel Id: 000210

Name: Fabrikam B2B Online Store

Operating unit number: 202

Legal entity: USRT

Customer type: B2B

Warehouse: DC-CENTRAL

Search name: Fabrikam B2B Online

Publishing status message:

Store time zone: (GMT-08:00) Pacific Time (...

UPDATES

Last updated: 9/7/2021 03:51:59 AM

ORDERS

Currency: USD

Default customer: 100002

Customer address book: USRTCentrl

PROFILES

Functionality profile:

Channel profile:

Live channel database:

SALES TAX

Prices include sales tax: No

EMAIL NOTIFICATION

Email notification profile: ENProfile

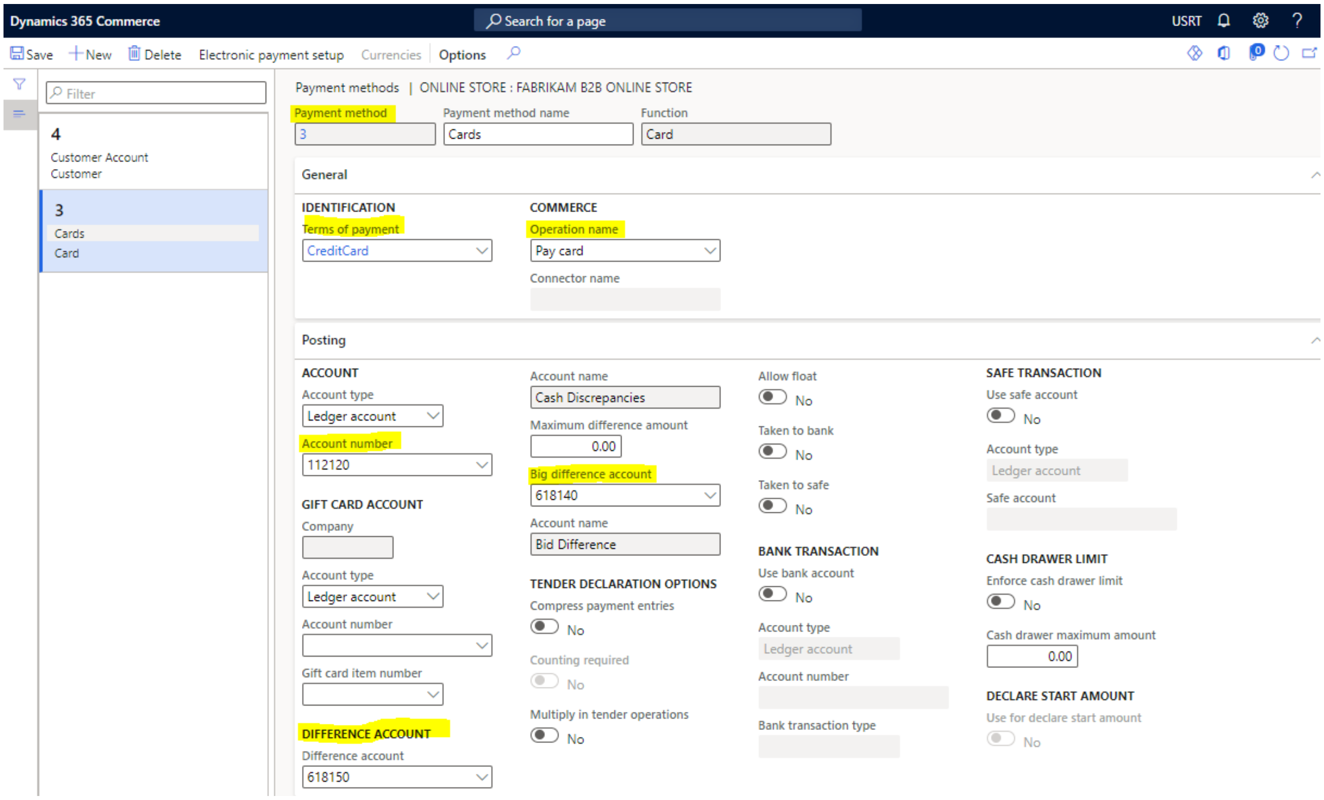
ELECTRONIC PAYMENTS

Default for unmapped processor ...:

“CREDIT CARDS”

We’ll now add another card type. Select “New” again.

- Set “Payment method” to **3**
- In the “General” section set
  - Terms of payment: “**CreditCard**”
  - Operation name “**Pay card**”
- In the Posting section set
  - In the “ACCOUNT” section set Account number to **112120** (or other if not using demo data)
  - Difference account: **618150** (or other if not using demo data)
  - Big difference account: **618140**
- Click **Save**



4
Customer Account
Customer

3
Cards
Card

Payment methods | ONLINE STORE : FABRIKAM B2B ONLINE STORE

Payment method: 3
Payment method name: Cards
Function: Card

General

IDENTIFICATION
Terms of payment: CreditCard

COMMERCE
Operation name: Pay card
Connector name:

Posting

ACCOUNT
Account type: Ledger account
Account number: 112120

GIFT CARD ACCOUNT
Company:
Account type: Ledger account
Account number:
Gift card item number:

DIFFERENCE ACCOUNT
Difference account: 618150

Account name: Cash Discrepancies
Maximum difference amount: 0.00
Big difference account: 618140
Account name: Bid Difference

TENDER DECLARATION OPTIONS
Compress payment entries: No
Counting required: No
Multiply in tender operations: No

Allow float: No
Taken to bank: No
Taken to safe: No

BANK TRANSACTION
Use bank account: No
Account type: Ledger account
Account number:
Bank transaction type:

SAFE TRANSACTION
Use safe account: No
Account type: Ledger account
Safe account:

CASH DRAWER LIMIT
Enforce cash drawer limit: No
Cash drawer maximum amount: 0.00

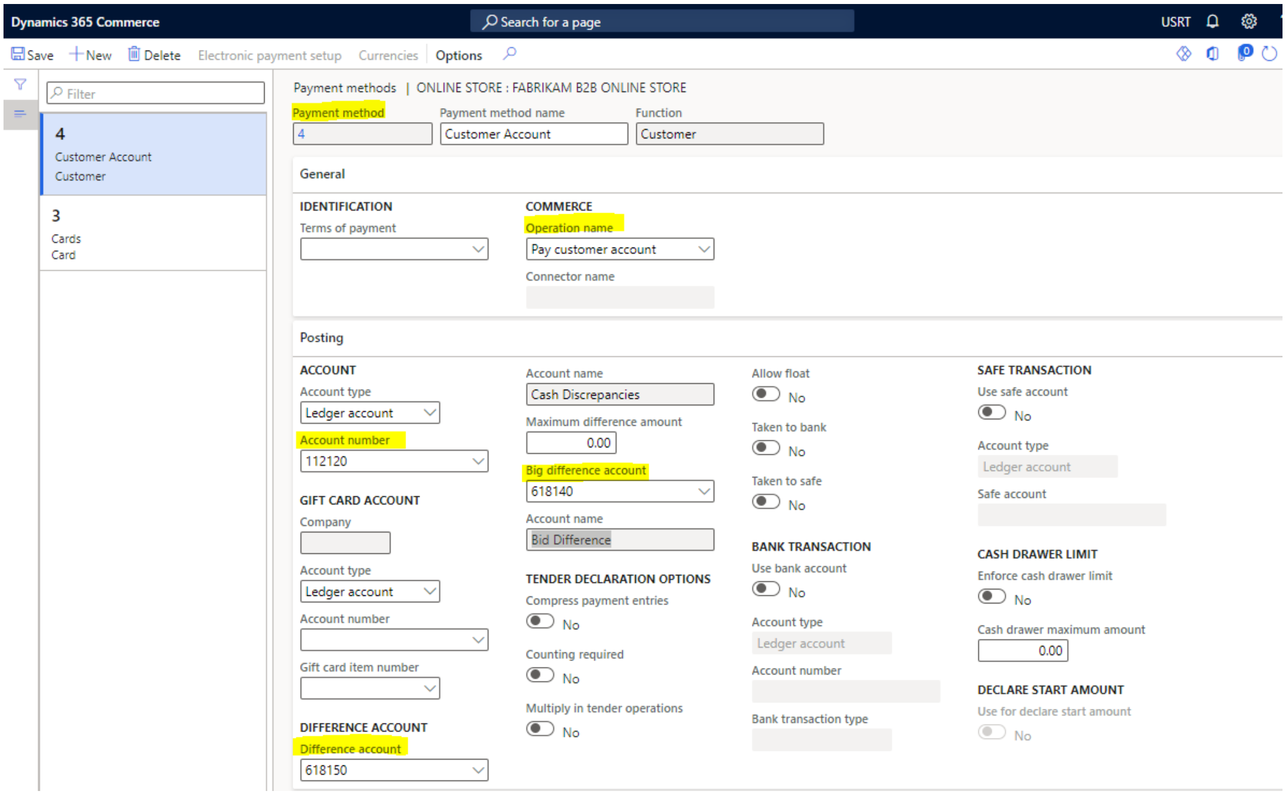
DECLARE START AMOUNT
Use for declare start amount: No

- Select “**Electronic payment setup**” and add cards to support, you should only need to set the “**Card ID**” for each, example add any of the following: AMEXPRESS, EUROCARD, MAESTRO, MASTER, VISA, VISAELEC (used for debit). Please include “Visa” to ensure the visa test card. Close window when completed.

“CUSTOMER ACCOUNT”

Select “New” again.

- Set “Payment method” to **4**
- In the “General” section set
  - Terms of payment - Blank
  - Operation name “**Pay customer account**”
- In the Posting section set
  - In the “ACCOUNT” section set Account number to **112120** (or other if not using demo data)
  - Difference account: **618150** (or other if not using demo data)
  - Big difference account: **618140**
- Click **Save**



4
Customer Account
Customer

3
Cards
Card

Payment methods | ONLINE STORE : FABRIKAM B2B ONLINE STORE

Payment method: 4
Payment method name: Customer Account
Function: Customer

General

IDENTIFICATION
Terms of payment:

COMMERCE
Operation name: Pay customer account
Connector name:

Posting

ACCOUNT
Account type: Ledger account
Account number: 112120

GIFT CARD ACCOUNT
Company:
Account type: Ledger account
Account number:
Gift card item number:

DIFFERENCE ACCOUNT
Difference account: 618150

Account name: Cash Discrepancies
Maximum difference amount: 0.00
Big difference account: 618140
Account name: Bid Difference

TENDER DECLARATION OPTIONS
Compress payment entries: No
Counting required: No
Multiply in tender operations: No

Allow float: No
Taken to bank: No
Taken to safe: No

BANK TRANSACTION
Use bank account: No
Account type: Ledger account
Account number:
Bank transaction type:

SAFE TRANSACTION
Use safe account: No
Account type: Ledger account
Safe account:

CASH DRAWER LIMIT
Enforce cash drawer limit: No
Cash drawer maximum amount: 0.00

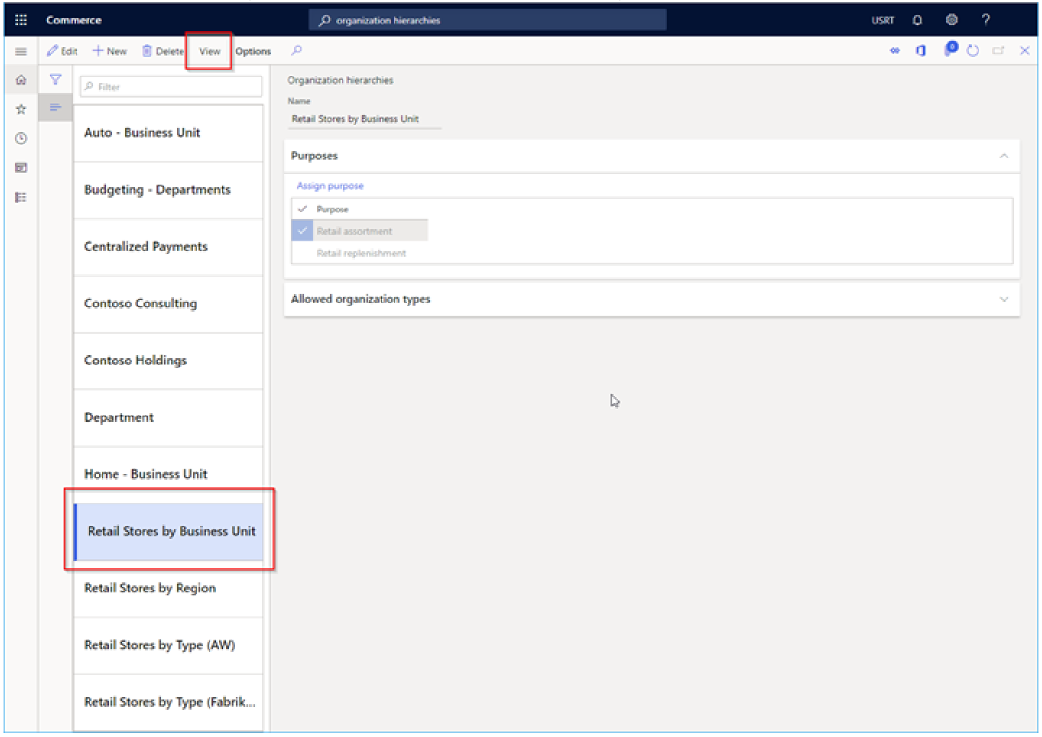
DECLARE START AMOUNT
Use for declare start amount: No

ADD CHANNEL TO ORG HIERARCHY

Business units can be used to organize channels for simpler configurations. For example, instead of configuring modes of delivery for each store independently, or assorting products per channel, they can be done by org structure units.

In this section we will add our new channel to an existing organization hierarchy business unit that is included as part of the demo data. You may want to create a new “Business Unit” for your channel and below steps can be used as a general guide.

- Search for the “**Organization hierarchies”** page
  - Select "**Retail Stores by Business Unit**" and "**View**" on top



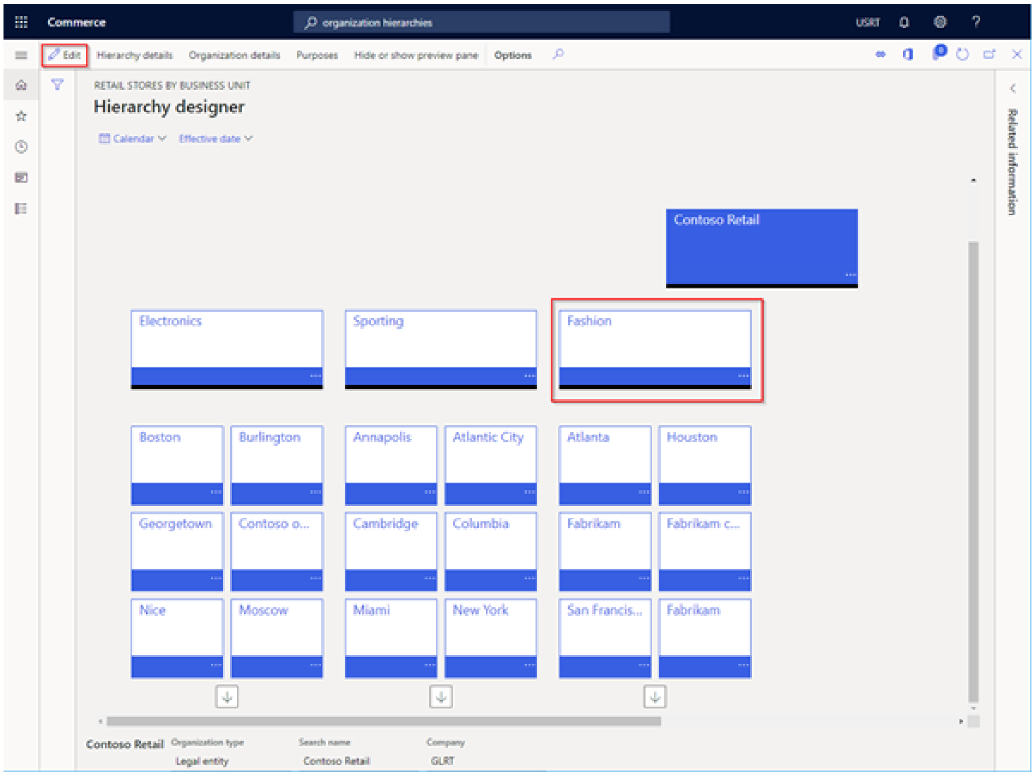
Auto - Business Unit
Budgeting - Departments
Centralized Payments
Contoso Consulting
Contoso Holdings
Department
Home - Business Unit
Retail Stores by Business Unit
Retail Stores by Region
Retail Stores by Type (AW)
Retail Stores by Type (Fabrik...

Organization hierarchies
Name
Retail Stores by Business Unit

Purposes
Assign purpose
Purpose
Retail assortment
Retail replenishment

Allowed organization types

  - Select “**Edit**” then the "**Fashion**" by hitting the “…” on fashion



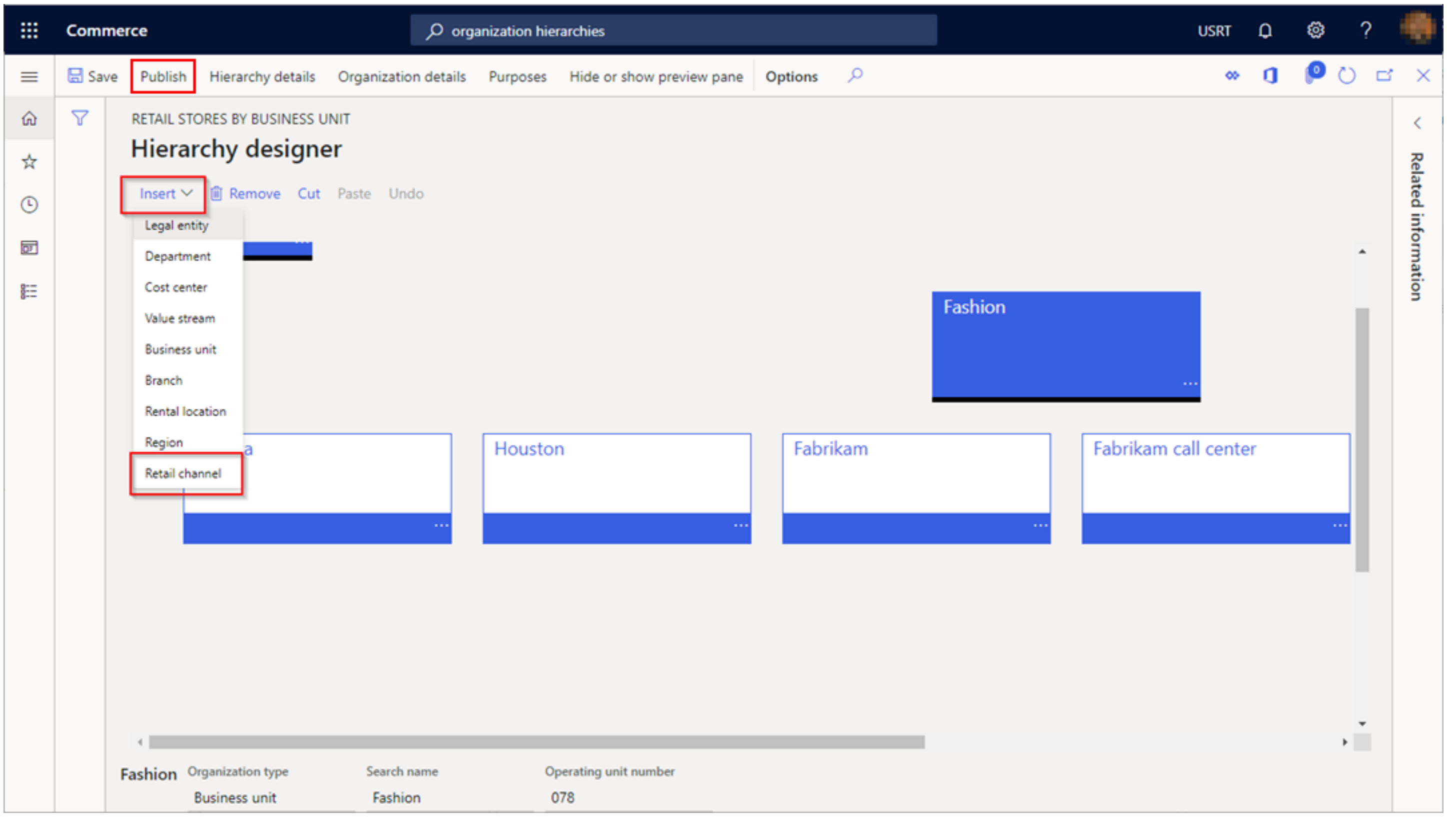
RETAIL STORES BY BUSINESS UNIT

Hierarchy designer

Contoso Retail

Organization type	Search name	Company
Legal entity	Contoso Retail	GLRT

- Select "**Insert**" then “**Retail channel**” and pick the new channel to be added
- Note: You may need to select the arrow to see the new channel added as you scroll to the right.
- Save, **publish**, select an older date (example: 1st of last month since only 1 publish a day is allowed, if another publish is required you can pick a newer date)



RETAIL STORES BY BUSINESS UNIT

Hierarchy designer

Fashion

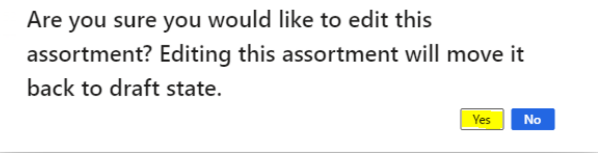
Organization type	Search name	Operating unit number
Business unit	Fashion	078

“ASSORTMENTS”

As we have already learnt how to setup assortments during our B2C training we will not be setting up a new one during this exercise, we will use the same assortment group as the “Fabrikam extended online store.

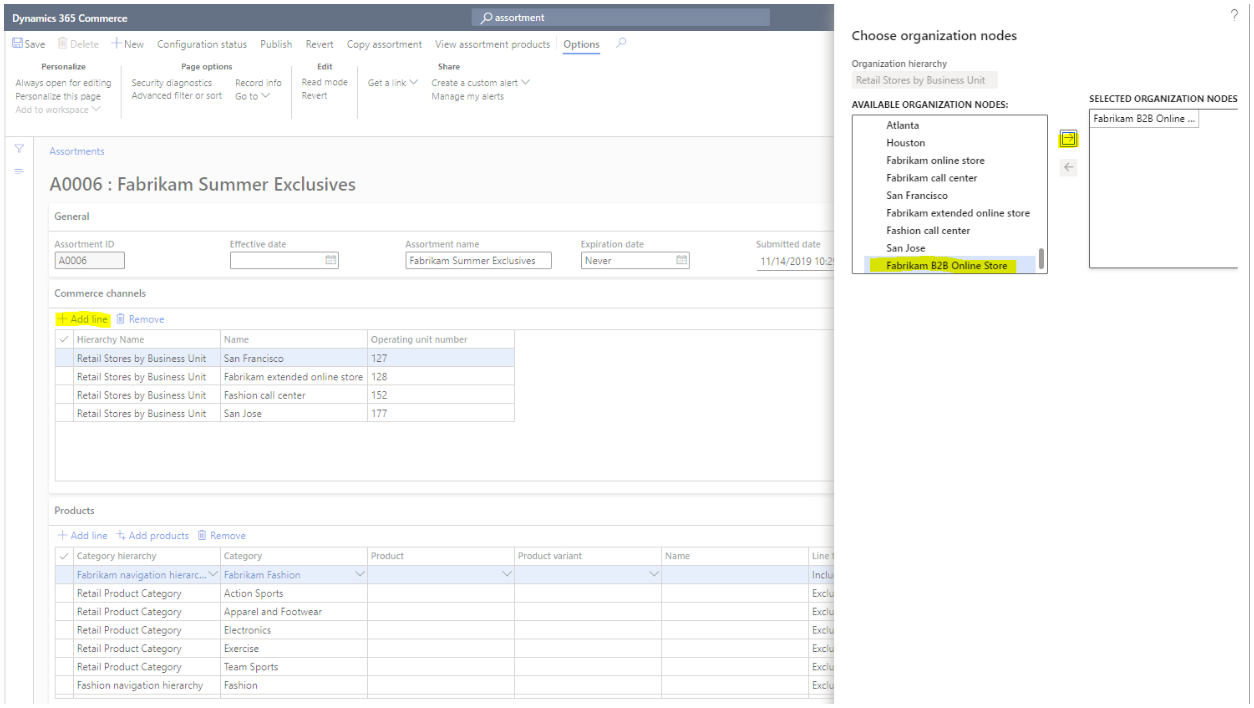
In the search bar type “**assortments**”, go to “Fabrikam Summer Exclusives”

Click “Edit”, you will receive a warning, click yes



Are you sure you would like to edit this assortment? Editing this assortment will move it back to draft state.

Add the “Fabrikam B2B Online store” and then click save and Publish.



A0006 : Fabrikam Summer Exclusives

General

Assortment ID: A0006
Effective date:
Assortment name: Fabrikam Summer Exclusives
Expiration date: Never
Submitted date: 11/14/2019 10:2

Commerce channels

Hierarchy Name	Name	Operating unit number
Retail Stores by Business Unit	San Francisco	127
Retail Stores by Business Unit	Fabrikam extended online store	128
Retail Stores by Business Unit	Fashion call center	152
Retail Stores by Business Unit	San Jose	177

Products

Category hierarchy	Category	Product	Product variant	Name	Line t
Fabrikam navigation hierarc...	Fabrikam Fashion				Inclu
Retail Product Category	Action Sports				Exclu
Retail Product Category	Apparel and Footwear				Exclu
Retail Product Category	Electronics				Exclu
Retail Product Category	Exercise				Exclu
Retail Product Category	Team Sports				Exclu
Fashion navigation hierarchy	Fashion				Exclu

Choose organization nodes

Organization hierarchy: Retail Stores by Business Unit

AVAILABLE ORGANIZATION NODES:

- Atlanta
- Houston
- Fabrikam online store
- Fabrikam call center
- San Francisco
- Fabrikam extended online store
- Fashion call center
- San Jose
- Fabrikam B2B Online Store

SELECTED ORGANIZATION NODES

- Fabrikam B2B Online ...

SETUP MODES OF DELIVERY

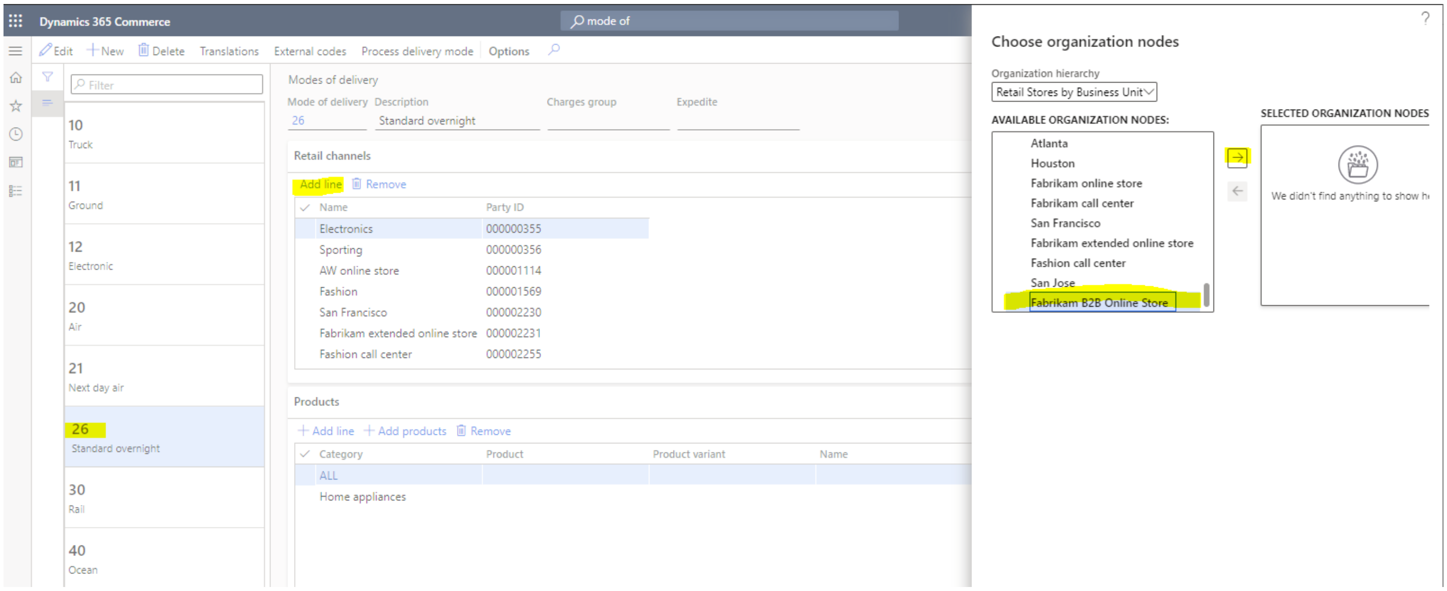
Modes of delivery are used at cart checkout when asking what delivery mode to use. We need to setup the modes of delivery available to the new online store channel.

Search for “**modes of delivery**” configuration in Headquarters.

Next, we will add this channel to the following modes of delivery.

- “26 - Standard overnight”
- “60 – In Store pick up”
- “61 -Curbside pick up”
- “99 - Standard”

Select each one and under “**Retail channels**”, click the “**Add line**” link then add the new channel:



10	Truck
11	Ground
12	Electronic
20	Air
21	Next day air
26	Standard overnight
30	Rail
40	Ocean

Modes of delivery

Mode of delivery	Description	Charges group	Expedite
26	Standard overnight		

Retail channels

Name	Party ID
Electronics	000000355
Sporting	000000356
AW online store	000001114
Fashion	000001569
San Francisco	000002230
Fabrikam extended online store	000002231
Fashion call center	000002255

Products

Category	Product	Product variant	Name
ALL			
Home appliances			

Choose organization nodes

Organization hierarchy: Retail Stores by Business Unit

AVAILABLE ORGANIZATION NODES:

- Atlanta
- Houston
- Fabrikam online store
- Fabrikam call center
- San Francisco
- Fabrikam extended online store
- Fashion call center
- San Jose
- Fabrikam B2B Online Store

SELECTED ORGANIZATION NODES

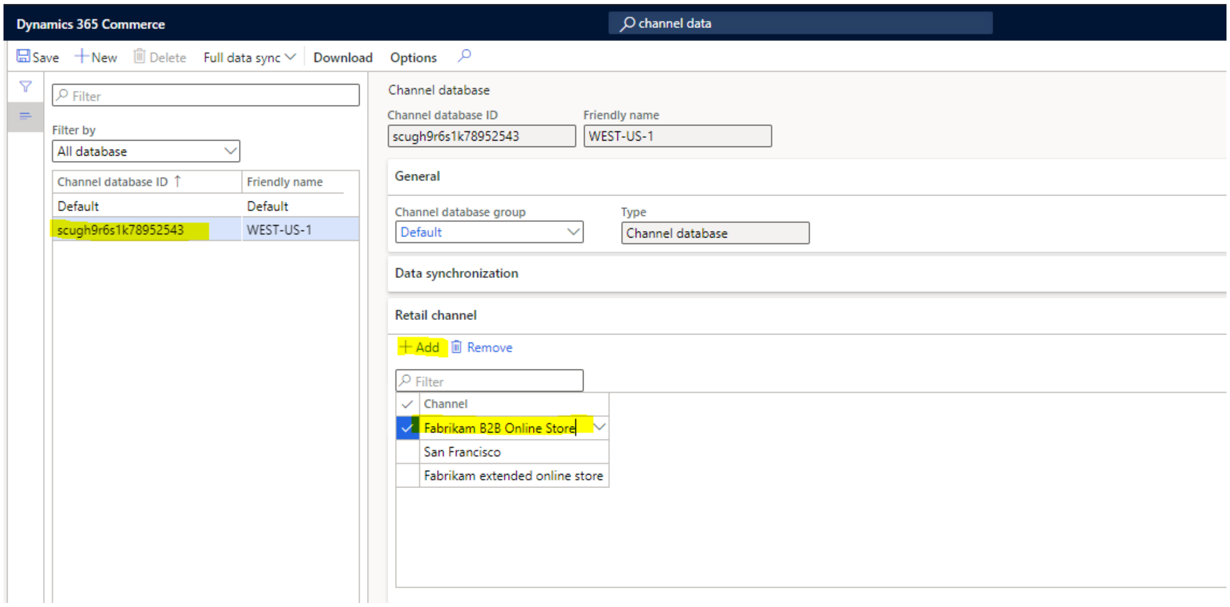
We didn't find anything to show h

After any changes you must run “Process delivery modes”, run now.

CONFIGURE CHANNEL TO BE PART OF THE APPROPRIATE CHANNEL DATABASE

Now that the channel creation and configuration are complete, we need to ensure we pick the correct channel database. By default, the deployments will contain a scale unit that we should select.

- Search for “**Channel database**” and load the page
- Select the scale unit channel database (not the “Default”) as shown in the image below.
- Select “**Edit**” and add the new channel then click save



Channel database ID	Friendly name
Default	Default
scugh9r6s1k78952543	WEST-US-1

Channel database

Channel database ID: scugh9r6s1k78952543

Friendly name: WEST-US-1

General

Channel database group: Default

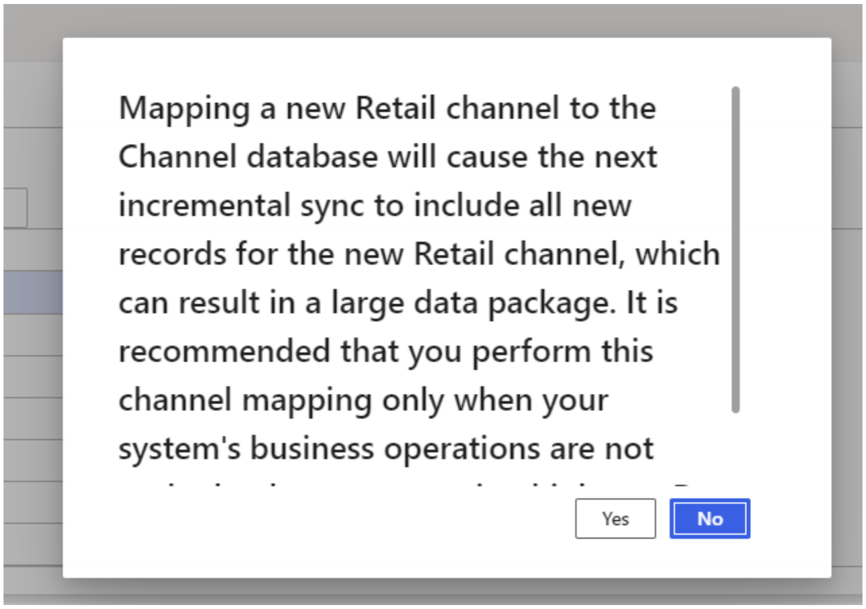
Type: Channel database

Data synchronization

Retail channel

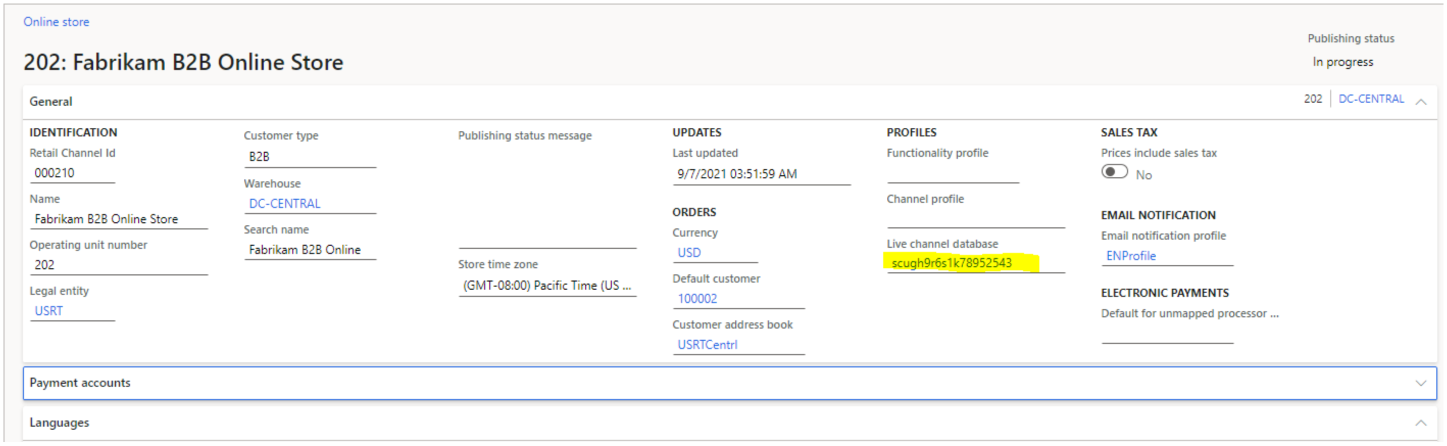
Channel
Fabrikam B2B Online Store
San Francisco
Fabrikam extended online store

Note: you should get a warning message as below, stating this could take some time, select “Yes”


Mapping a new Retail channel to the Channel database will cause the next incremental sync to include all new records for the new Retail channel, which can result in a large data package. It is recommended that you perform this channel mapping only when your system's business operations are not

- Note we will later run the “Full data sync” from here but not yet…

If you now go back to the online store channel you should see the channel database listed as shown below:



Online store

202: Fabrikam B2B Online Store

Publishing status
In progress

General — 202 | DC-CENTRAL

IDENTIFICATION
Retail Channel Id: 000210
Name: Fabrikam B2B Online Store
Operating unit number: 202
Legal entity: USRT

Customer type: B2B
Warehouse: DC-CENTRAL
Search name: Fabrikam B2B Online

Publishing status message:
Store time zone: (GMT-08:00) Pacific Time (US ...

UPDATES
Last updated: 9/7/2021 03:51:59 AM

ORDERS
Currency: USD
Default customer: 100002
Customer address book: USRTCentrl

PROFILES
Functionality profile:
Channel profile:
Live channel database: scugh9r6s1k78952543

SALES TAX
Prices include sales tax: No

EMAIL NOTIFICATION
Email notification profile: ENProfile

ELECTRONIC PAYMENTS
Default for unmapped processor ...:

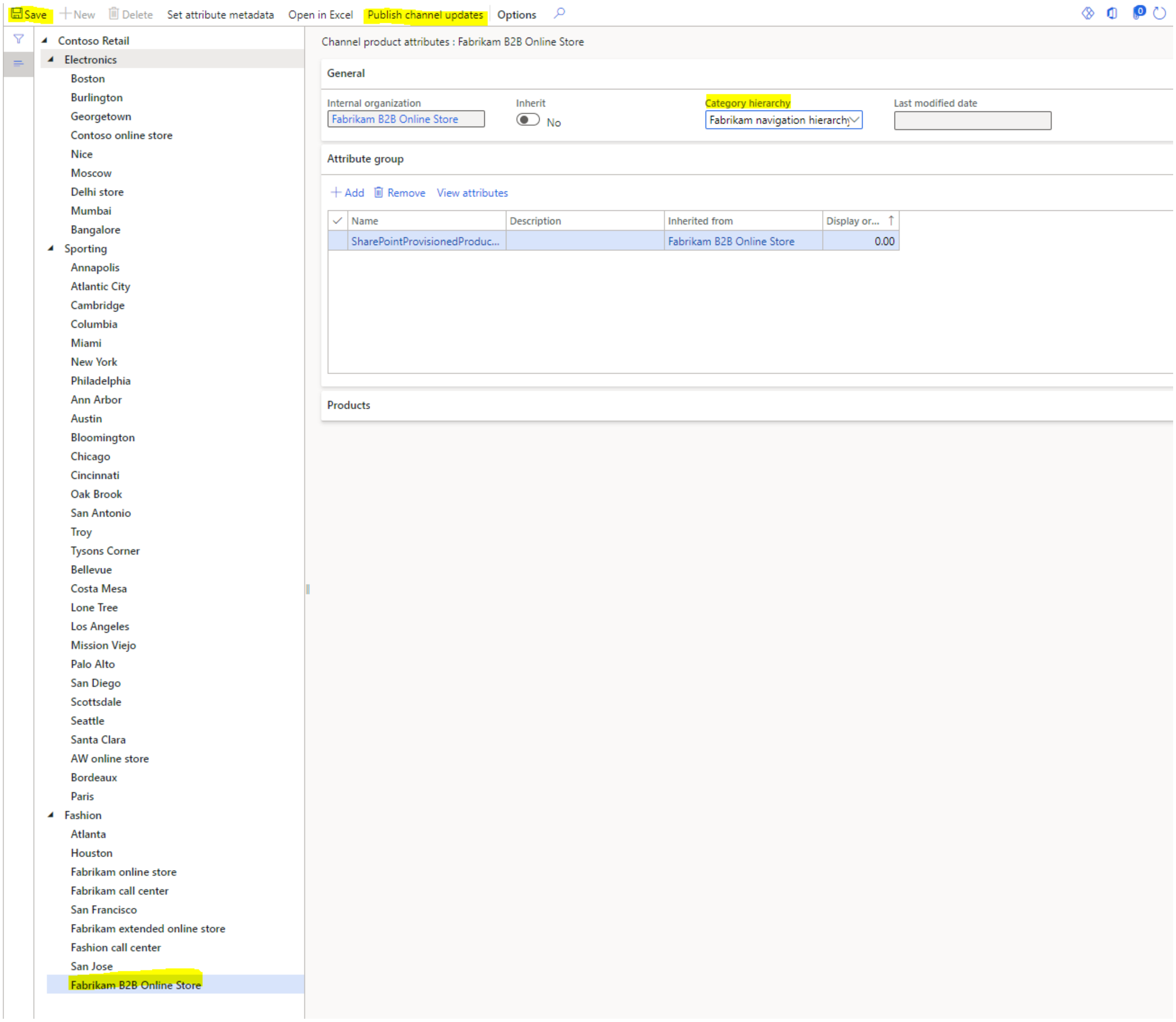
“CHANNEL NAVIGATION”

We now need to setup our channel navigation and since we are following the Fabrikam option we will set the navigation hierarchy to be the same as the “Fabrikam extended online store”.

Search “**Channel categories and product attributes**”

Select “Fabrikam B2B Online Store, then edit and change category hierarchy to “Fabrikam navigation hierarchy”.

Click save and then Publish channel updates.



- Contoso Retail
  - Electronics
    - Boston
    - Burlington
    - Georgetown
    - Contoso online store
    - Nice
    - Moscow
    - Delhi store
    - Mumbai
    - Bangalore
  - Sporting
    - Annapolis
    - Atlantic City
    - Cambridge
    - Columbia
    - Miami
    - New York
    - Philadelphia
    - Ann Arbor
    - Austin
    - Bloomington
    - Chicago
    - Cincinnati
    - Oak Brook
    - San Antonio
    - Troy
    - Tysons Corner
    - Bellevue
    - Costa Mesa
    - Lone Tree
    - Los Angeles
    - Mission Viejo
    - Palo Alto
    - San Diego
    - Scottsdale
    - Seattle
    - Santa Clara
    - AW online store
    - Bordeaux
    - Paris
  - Fashion
    - Atlanta
    - Houston
    - Fabrikam online store
    - Fabrikam call center
    - San Francisco
    - Fabrikam extended online store
    - Fashion call center
    - San Jose
    - Fabrikam B2B Online Store

Channel product attributes : Fabrikam B2B Online Store

General

Internal organization: Fabrikam B2B Online Store
Inherit: No
Category hierarchy: Fabrikam navigation hierarchy
Last modified date:

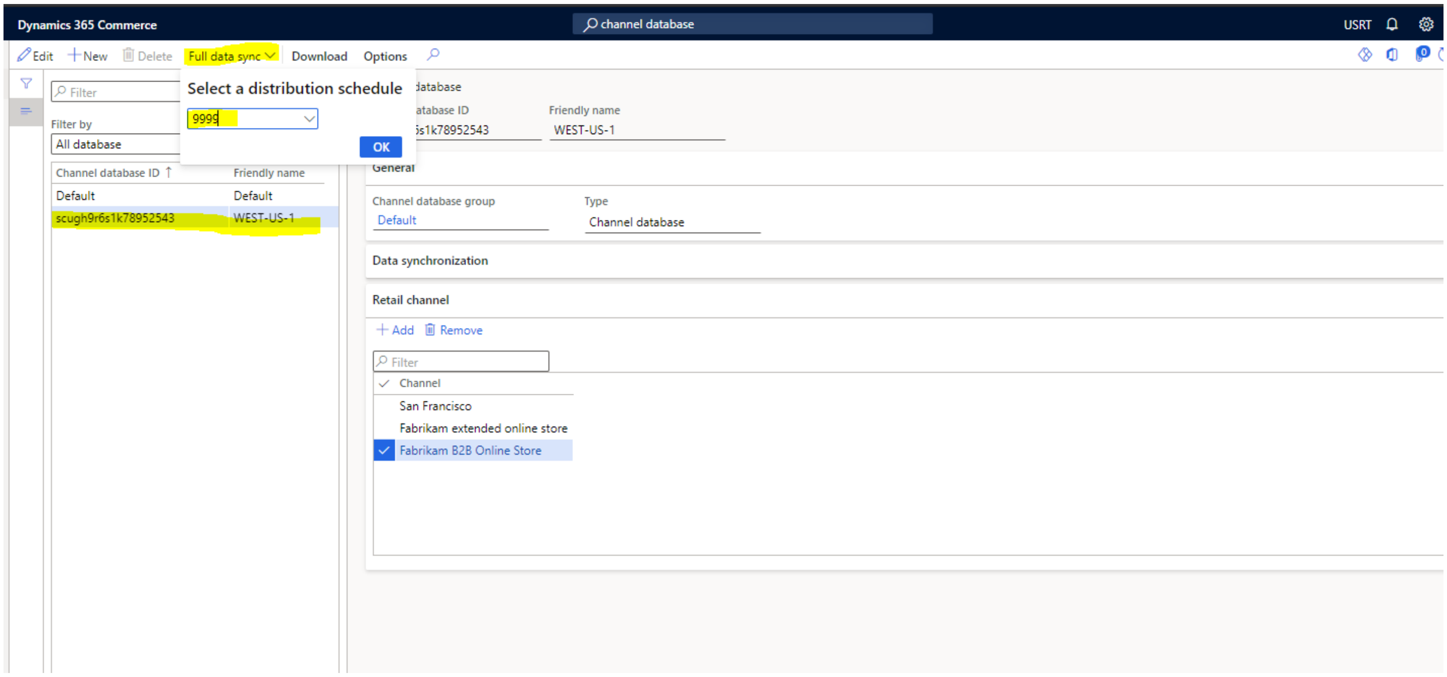
Attribute group

Name	Description	Inherited from	Display or...
SharePointProvisionedProduc...		Fabrikam B2B Online Store	0.00

Products

To sync all jobs to ensure all the data gets sync'd to the scale unit - search for the "**Channel Database**" page.

- **Ensure** the RSCU channel database is selected and select "**Full data sync**" drop down and type in "**9999**" to run all jobs. Press "**OK**" on the next screen



Select a distribution schedule: 9999

Filter by: All database

Channel database ID	Friendly name
Default	Default
scugh9r6s1k78952543	WEST-US-1

database

atabase ID: 6s1k78952543

Friendly name: WEST-US-1

Channel database group: Default

Type: Channel database

Channel
San Francisco
Fabrikam extended online store
✓ Fabrikam B2B Online Store

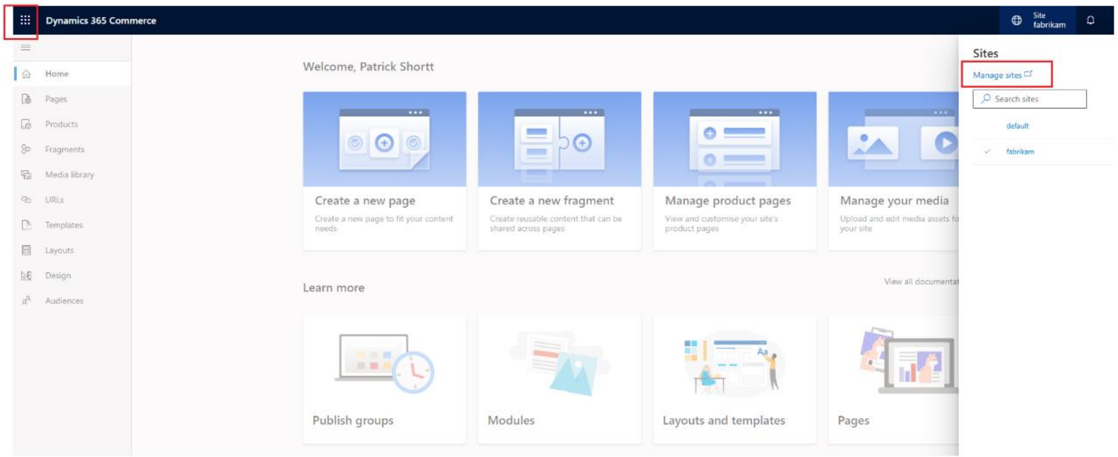
- This could take 10-15 minutes to complete (coffee break time!). It could take up to another 15 minutes for the Azure search indexing before products will be viewable on the e-Commerce site.
- When the dialog goes away type "**download sessions**" in the search box and refresh until all Status items say "Applied"

UNIT 2B BUSINESS TO BUSINESS (B2B) SITE SETUP

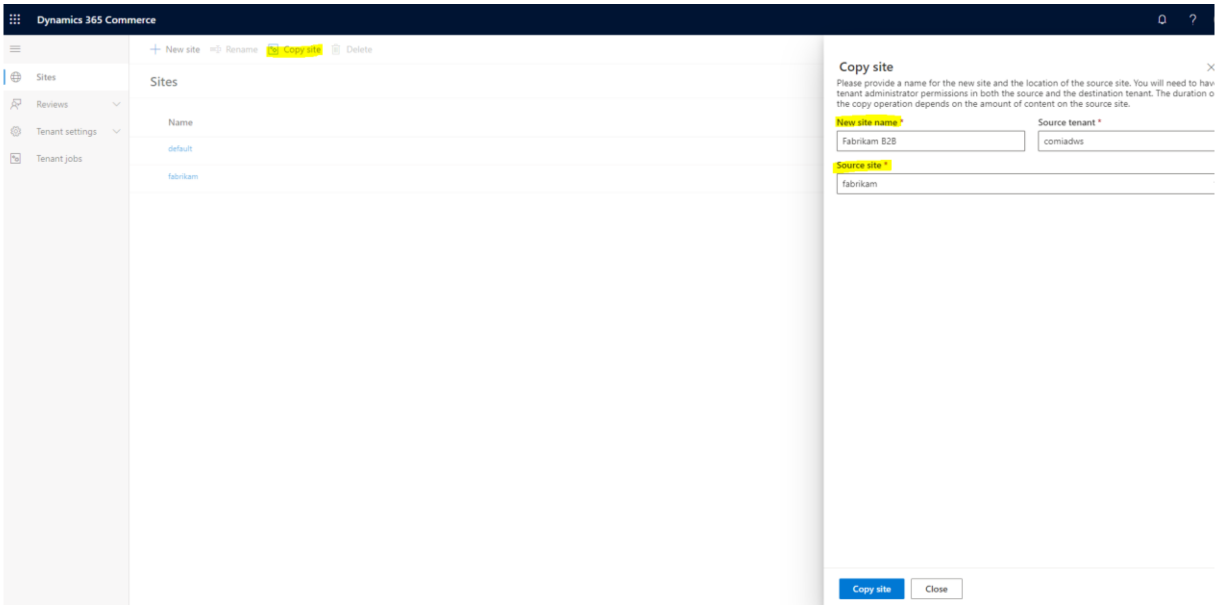
Business-to-business (B2B) e-commerce sites provide some key capabilities that optimize the workflow for a B2B user. In this unit we will be creating the pages required for a B2B site, however before doing this we need to do a site copy of our B2C site so we don't have to create all the templates, pages URL's etc from scratch.

SETUP YOUR B2B E-COMMERCE SITE

Go to the site builder tool, "Manage sites" or "Home"



"Copy Site". Give your new site a name and select the source site which is your B2C site.



Copy site

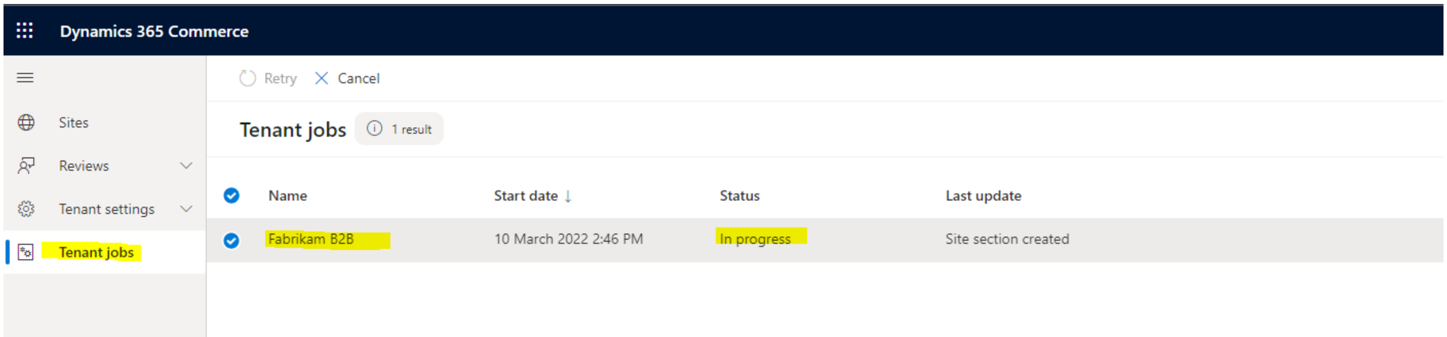
Please provide a name for the new site and the location of the source site. You will need to hav tenant administrator permissions in both the source and the destination tenant. The duration o the copy operation depends on the amount of content on the source site.

New site name *: Fabrikam B2B

Source tenant *: comiadws

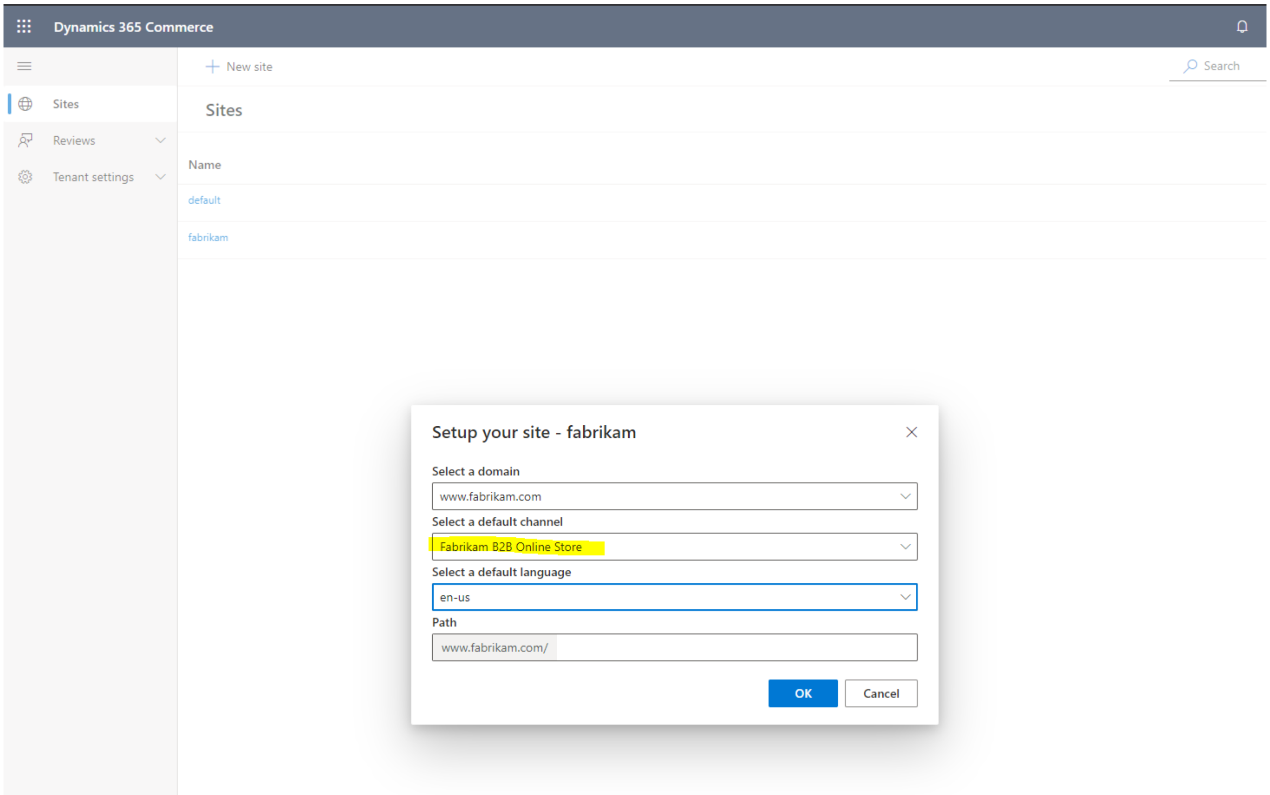
Source site *: fabrikam

Click on Tenant jobs to see the progress, the job will take around 10+ mins to complete.



Name	Start date ↓	Status	Last update
Fabrikam B2B	10 March 2022 2:46 PM	In progress	Site section created

Once the site has been copied you need to setup the site information, similar to the setup below. Remember you need to select the new B2B online store you have just created in the "select a default channel" and give it a path name. For example: www.fabrikam.com/B2B



Setup your site - fabrikam

Select a domain: www.fabrikam.com

Select a default channel: Fabrikam B2B Online Store

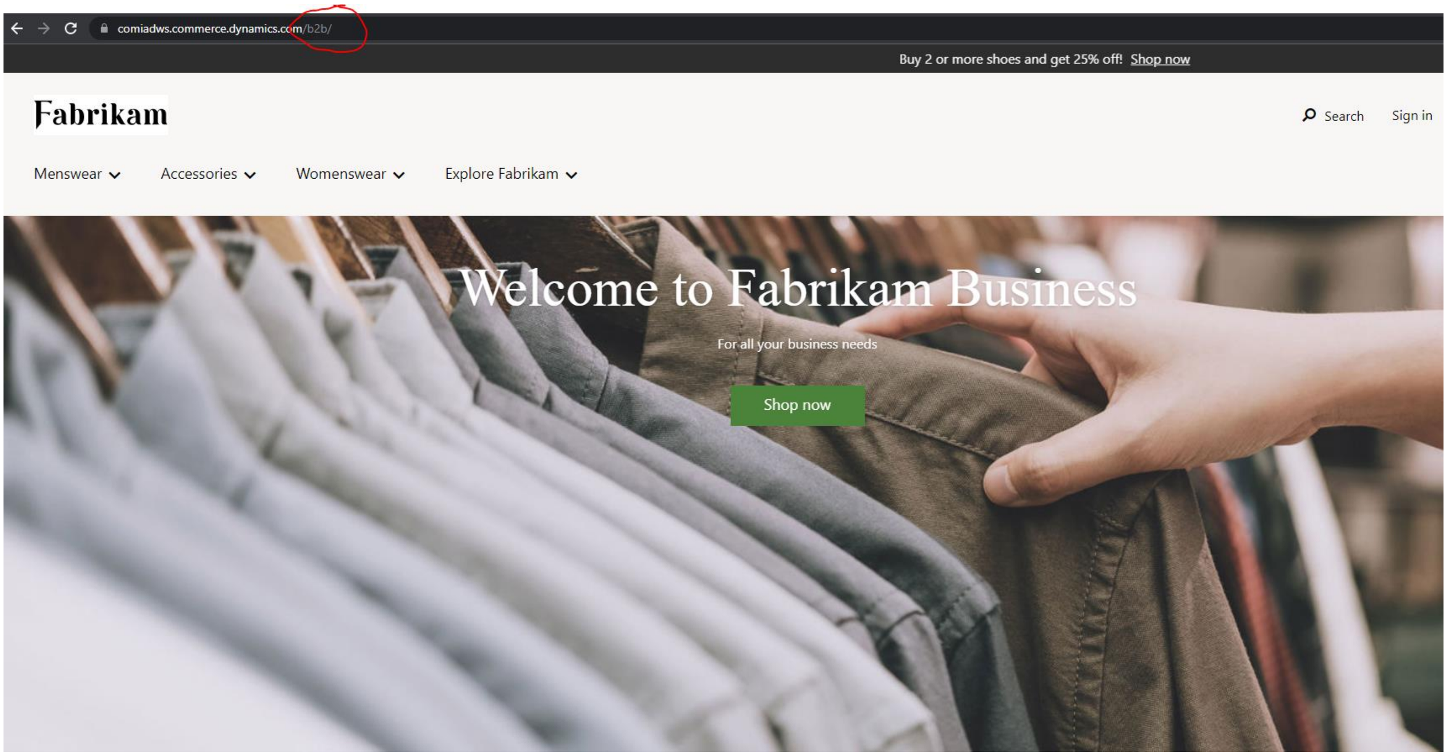
Select a default language: en-us

Path: www.fabrikam.com/

Note: You should now have a fully functional site, with pages, products, fragments, media library, URLs and Templates populated based on the site you just copied.

As you already know how to create a B2C site and change the look and feel you might want to change your homepage first so you don't mix up your B2C site and B2B site.

Example:



comiadws.commerce.dynamics.com/b2b/

Buy 2 or more shoes and get 25% off! Shop now

Welcome to Fabrikam Business

For all your business needs

BUSINESS PARTNER REQUEST PAGE

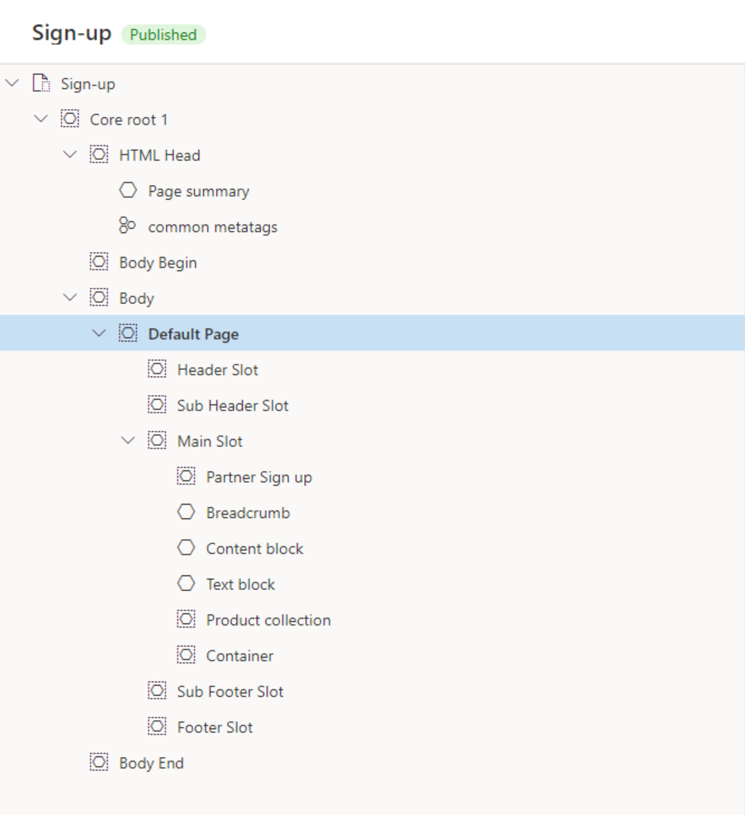
One of the first pages which need to setup for our B2B site is the business partner sign-up pages, this will allow potential business partners to sign up to become a business partner.

The **Partner sign up** module on a business partner request page is used to initiate user requests to become business partners. This module lets you collect the user information that is required for the sign-up process. Additionally, the **Business account address** module is used to capture the user's address.

To set up and configure the business partner request page in site builder, follow these steps.

1. Create a template that is named **Business Partner Sign-up**. This template should include the following modules:
    - Partner sign up
    - Breadcrumb
    - Container
    - Content block
    - Text block
    - Product collection

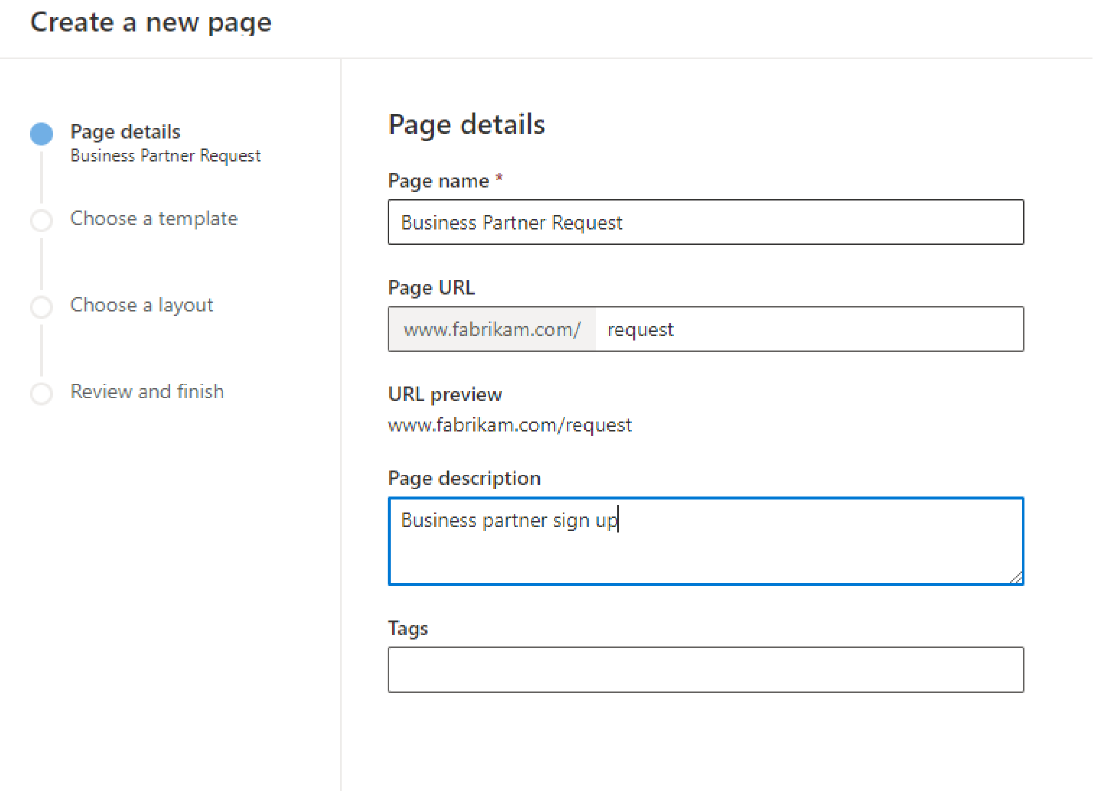
Note: Please make sure you add the “Page summary & common metatags”.



Sign-up Published

- Sign-up
  - Core root 1
    - HTML Head
      - Page summary
      - common metatags
    - Body Begin
    - Body
      - **Default Page**
        - Header Slot
        - Sub Header Slot
        - Main Slot
          - Partner Sign up
          - Breadcrumb
          - Content block
          - Text block
          - Product collection
          - Container
        - Sub Footer Slot
        - Footer Slot
    - Body End

2. Go to Pages and use the **Sign-up** template to create a page that is named **Business Partner Request**.



Create a new page

- Page details
  Business Partner Request
- Choose a template
- Choose a layout
- Review and finish

Page details

Page name *: Business Partner Request

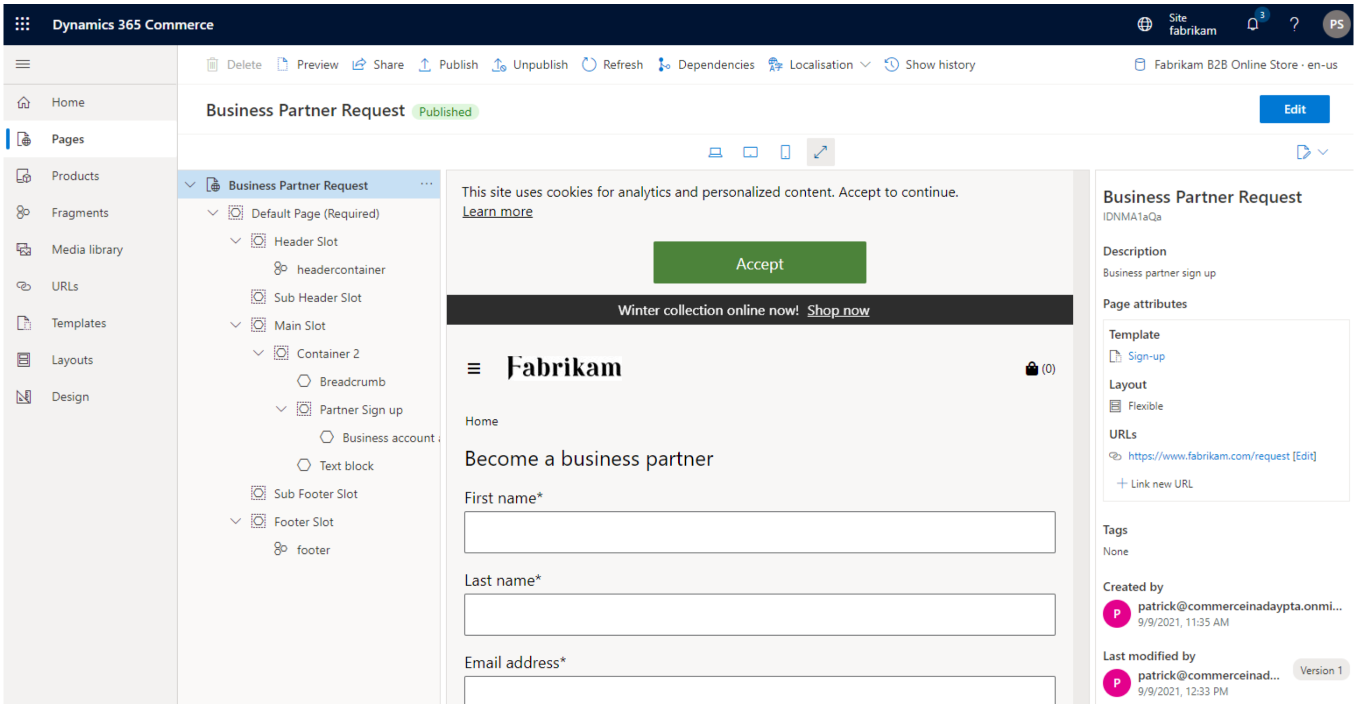
Page URL: www.fabrikam.com/ request

URL preview: www.fabrikam.com/request

Page description: Business partner sign up

Tags:

3. In the **Header** slot, add the header fragment that is preconfigured with the site header.
4. In the **Footer** slot, add the footer fragment that is preconfigured with the site footer.
5. In the **Main** slot, add a **Container** module. In the module properties pane, set the **Width** value to **Fill Container**.
6. In the **Container** slot, add a **Breadcrumb** module. In the module properties pane, under **Links**, configure a link to the home page, and enter **Home** as the link text.
7. In the **Container** slot, add a **Partner sign up** module below the **Breadcrumb** module. In the module properties pane, under **Heading**, enter **Become a business partner**.
8. In the **Partner sign up** slot, add a **Business account address** module.
9. In the **Container** slot, add a **Text block** module below the **Partner sign up** module. Here, you can define some terms and conditions for the sign-up process.
10. Select **Save**, and then select **Preview** to preview the page.
11. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.
12. Don’t forget to publish the URL for the page.



Business Partner Request Published

This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Winter collection online now! Shop now

Become a business partner

First name*

Last name*

Email address*

Business Partner Request
IDNMA1aQa

Description
Business partner sign up

Page attributes

Template
Sign-up

Layout
Flexible

URLs
https://www.fabrikam.com/request [Edit]
+ Link new URL

Tags
None

Created by
patrick@commerceinadaypta.onmi...
9/9/2021, 11:35 AM

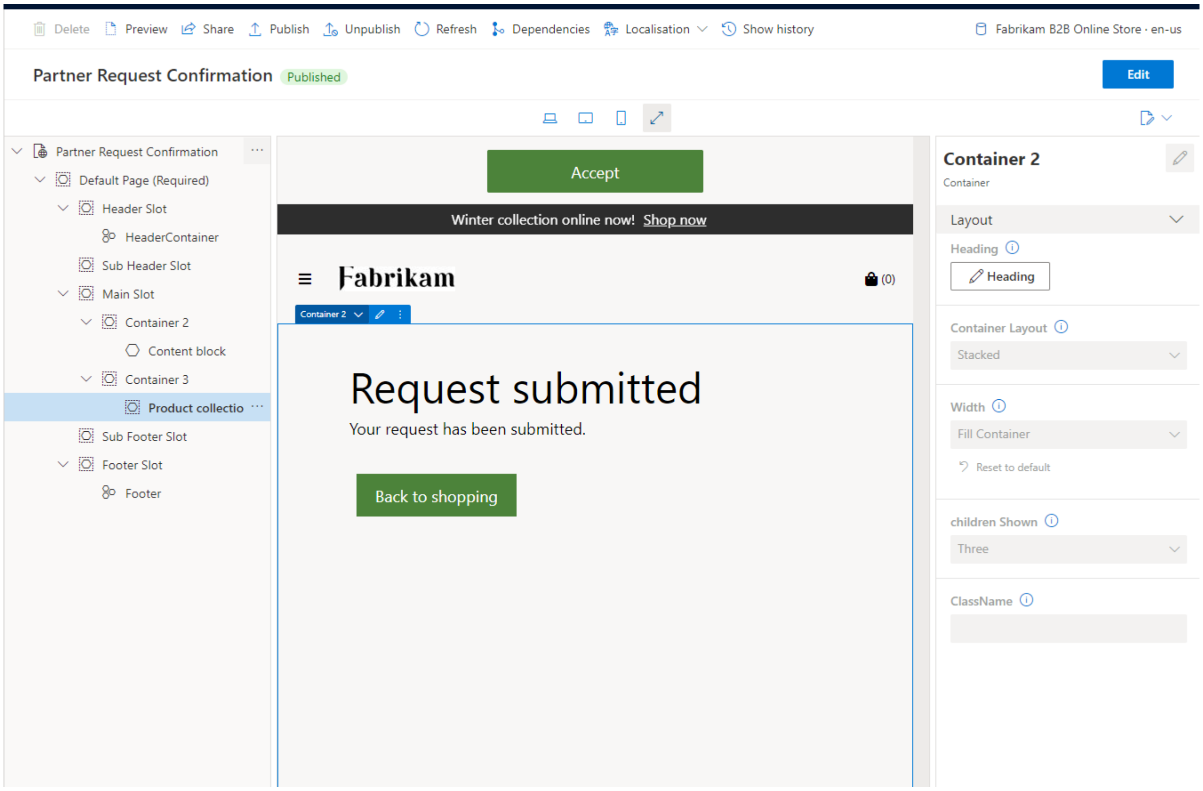
Last modified by
patrick@commerceinad... Version 1
9/9/2021, 12:33 PM

BUSINESS PARTNER REQUEST CONFIRMATION PAGE

After a business partner request is submitted, a confirmation page should be shown to the user to acknowledge the submission.

To set up and configure the request confirmation page in site builder, follow these steps.

1. Use the **Business Partner Signup** template that you created earlier to create a page that is named **Partner Request Confirmation**.
2. In the **Header** slot, add the header fragment that is preconfigured with the site header.
3. In the **Footer** slot, add the footer fragment that is preconfigured with the site footer.
4. In the **Main** slot, add a **Container** module. In the module properties pane, set the **Width** value to **Fill Container**.
5. In the **Container** slot, add a **Content block** module. In the module properties pane, under **Heading**, enter **Request submitted**. In the **Rich Text** field, enter **Your request has been submitted**. Under **Links**, configure a link to the home page, and enter **Back to shopping** as the link text.
6. Optional - Add another **Container** module, and add a **Product Collection** module to it.
7. Configure the **Product Collection** module with the recommendation or category list that you want to showcase on the page. This can be setup later.
8. Select **Save**, and then select **Preview** to preview the page.
9. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.
10. Publish the URL for the page.



Request submitted

Your request has been submitted.

Heading: Heading

Container Layout: Stacked

Width: Fill Container

children Shown: Three

ClassName:

To add a link to the request confirmation page in site builder, follow these steps.

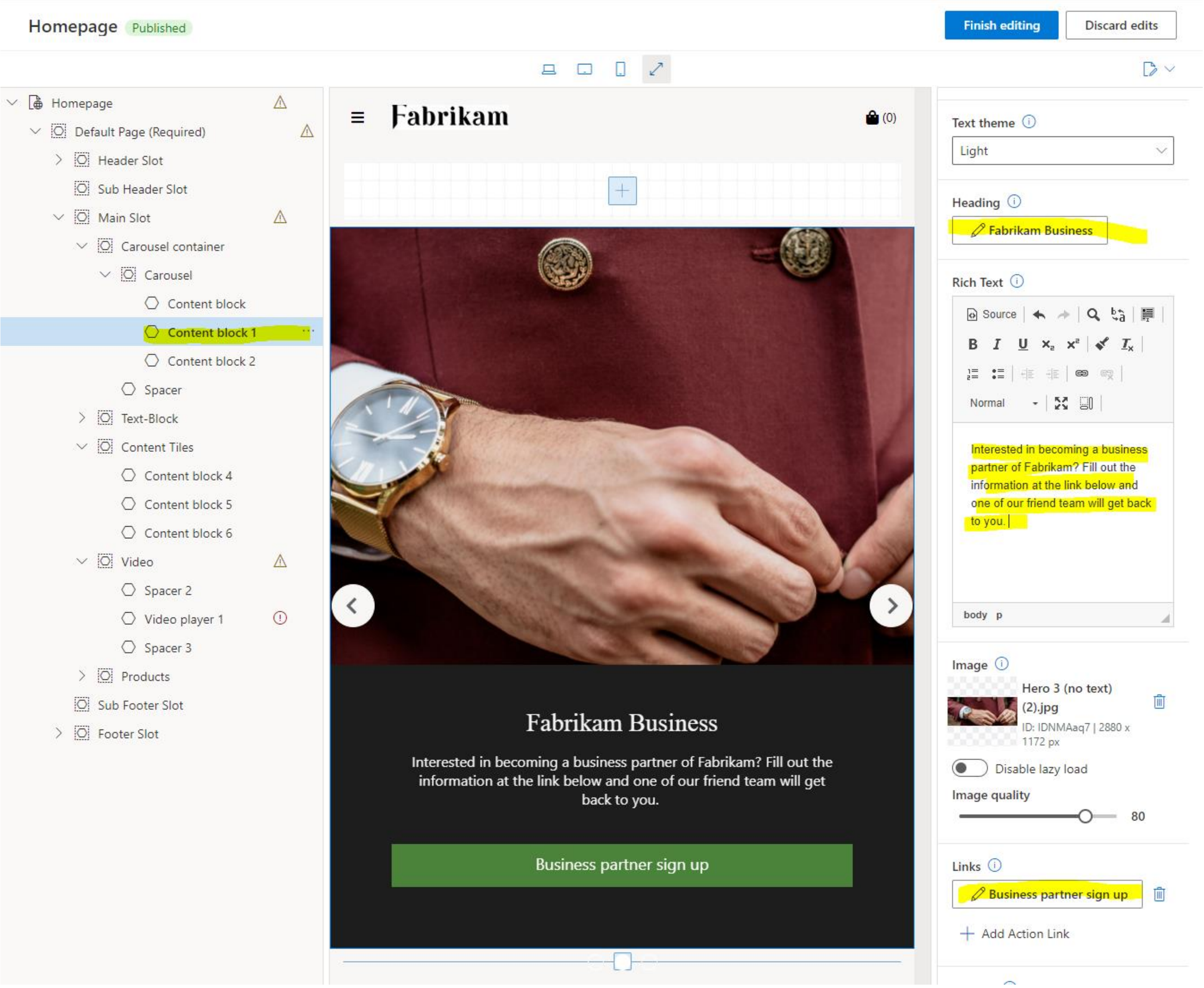
1. Go to the **Business Partner Request** page that you created earlier and select **Edit**.
2. Select the **Partner sign up** module slot. In the properties pane, under **Link to the Sign-Up Confirmation page**, configure the link to the business partner request page that you created earlier.
3. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.

BUSINESS PARTNER REQUEST LINK TO THE HOME PAGE

After the business partner request sign-up and confirmation pages are created, you must make the sign-up page accessible through a link on the home page. You can complete this task by adding the link to any **Content block** module in the on the home page.

To add a business partner request link to the home page in site builder, follow these steps.

1. Go to the home page for your site, and select **Edit**.
2. Select **Content block 1** module slot in the carousel container. In the module properties pane, under **Links**, configure a link to the business partner request page that you created earlier, and enter **Business partner sign up** or similar text as the link text.
3. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.



Fabrikam Business

Interested in becoming a business partner of Fabrikam? Fill out the information at the link below and one of our friend team will get back to you.

Business partner sign up

Text theme: Light

Heading: Fabrikam Business

Rich Text: Interested in becoming a business partner of Fabrikam? Fill out the information at the link below and one of our friend team will get back to you.

Image: Hero 3 (no text) (2).jpg
ID: IDNMAaq7 | 2880 x 1172 px

Image quality: 80

Links: Business partner sign up

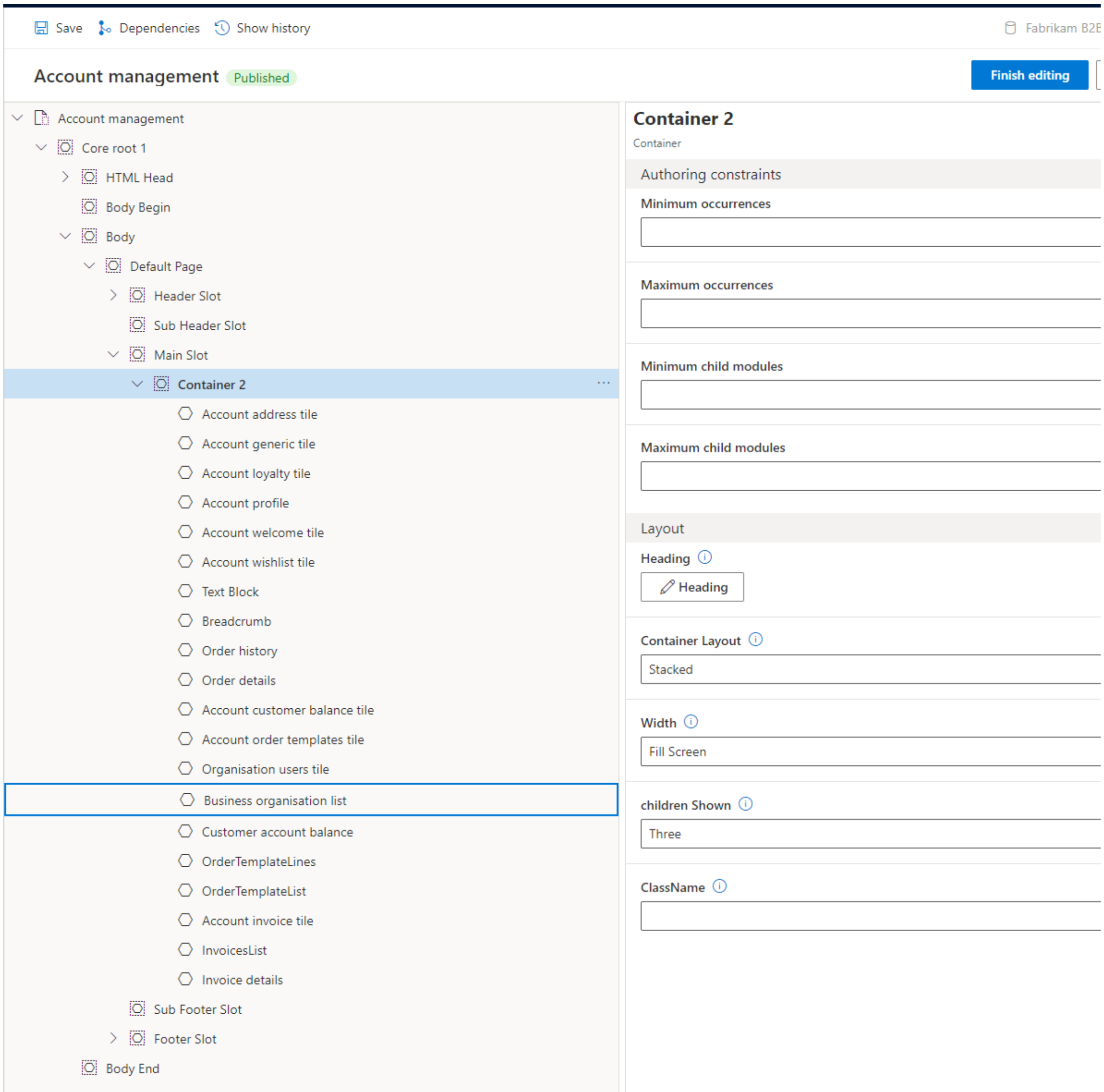
ACCOUNT MANAGEMENT LANDING PAGE

The account management landing page includes all the account management information that is required for both B2B and B2C e-commerce sites. Only signed-in users can view this page.

We will use the existing account management template that has been already created during the first run experience and just add the missing options.

1. Go to the template named **Account management** and make sure all the options below are part of the template. The ones in blue need to be added, starting from "Account customer balance tile".
    - Header
    - Footer
    - Breadcrumb
    - Account welcome tile
    - Account generic tile
    - Account Address tile
    - Account wishlist tile
    - Account address tile
    - Account loyalty tile
    - Account customer balance tile
    - Account order templates tile
    - Organization users
    - Business organization list
    - Customer account balance
    - Order template lines
    - Order template list
    - Account invoice tile
    - Invoices list
    - Invoice details

Once finished, save and publish.



Account management Published

- Account management
  - Core root 1
    - HTML Head
    - Body Begin
    - Body
      - Default Page
        - Header Slot
        - Sub Header Slot
        - Main Slot
          - Container 2
            - Account address tile
            - Account generic tile
            - Account loyalty tile
            - Account profile
            - Account welcome tile
            - Account wishlist tile
            - Text Block
            - Breadcrumb
            - Order history
            - Order details
            - Account customer balance tile
            - Account order templates tile
            - Organisation users tile
            - Business organisation list
            - Customer account balance
            - OrderTemplateLines
            - OrderTemplateList
            - Account invoice tile
            - InvoicesList
            - Invoice details
        - Sub Footer Slot
        - Footer Slot
    - Body End

Container 2

Container

Authoring constraints

Minimum occurrences

Maximum occurrences

Minimum child modules

Maximum child modules

Layout

Heading

Heading

Container Layout: Stacked

Width: Fill Screen

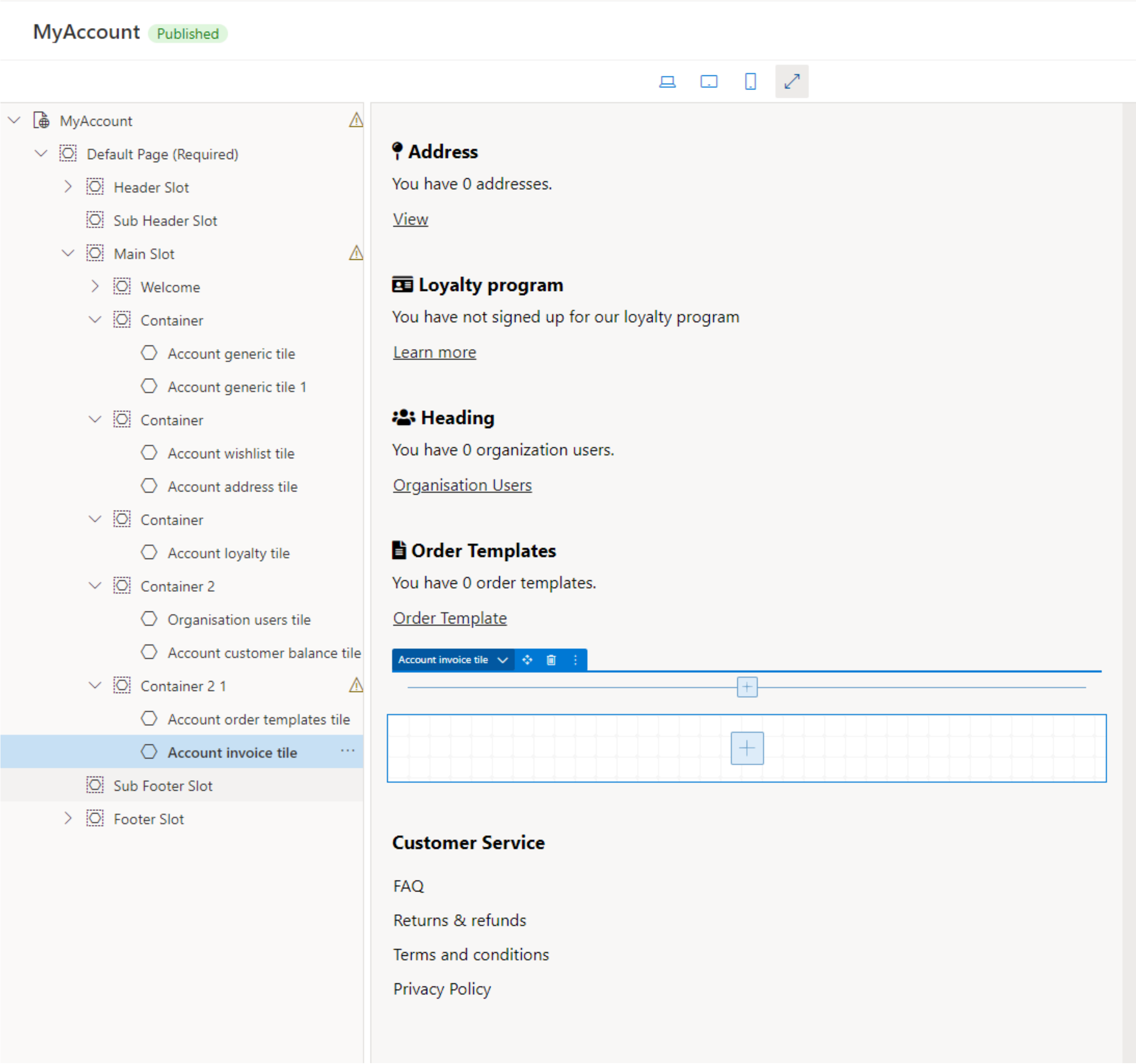
children Shown: Three

ClassName

2. Go to Pages and find **My Account page**.
3. In the **Main** slot, add a **Container** module. In the module properties pane, set the **Width** value to **Fill Container**.
4. In the **Main** slot, add another **Container** module. In the module properties pane, set the **Width** value to **Fill Container**. Set the **Children Shown** value to **Two**.
5. In the **Container** slot, add an **Organization users** module. In the module properties pane, under **Heading**, enter **Organization Users**.
6. In the **Container** slot, add an **Account customer balance tile** module. In the module properties pane, under **Heading**, enter **Account credit**.
7. In the **Main** slot, add another **Container** module. In the module properties pane, set the **Width** value to **Fill Container**. Set the **Children Shown** value to **Two**.
8. In the **Container** module, add an **Account order templates tile** module. In the module properties pane, under **Heading**, enter **Order Templates**.
9. In the **Container** module, add an **Account invoice tile** module. In the module properties pane, under **Heading**, enter **My Invoices**.

10. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.

Important note: Some of the sections that you added won't appear on the "what you see is what you get" (WYSIWIG) canvas in site builder, because they require a signed-in B2B account.



Address

You have 0 addresses.

View

Loyalty program

You have not signed up for our loyalty program

Learn more

Heading

You have 0 organization users.

Organisation Users

Order Templates

You have 0 order templates.

Order Template

Customer Service

FAQ

Returns & refunds

Terms and conditions

Privacy Policy

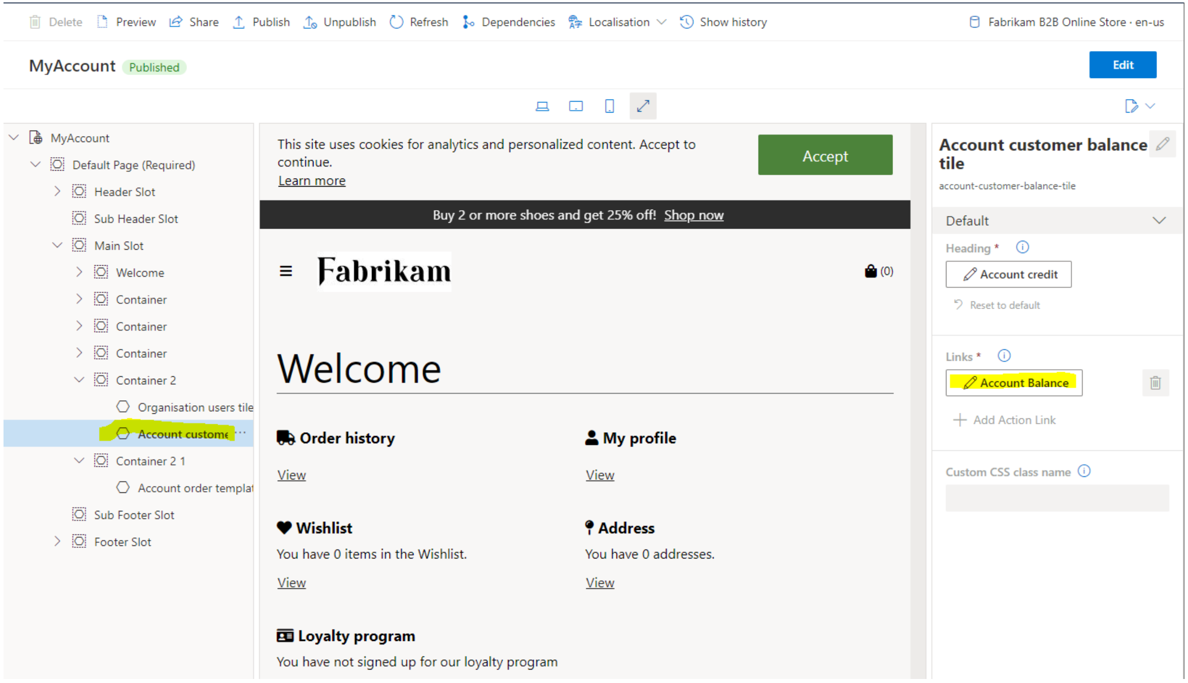
CUSTOMER BALANCE PAGES

Customer accounts can be used to pay for orders, the available balance in a customer account can be viewed from a user's account management page.

Before a signed-in B2B users can view their customer balance, you must create a customer balance page.

To create a customer balance page in site builder, follow these steps.

1. Use the **Account management** template to create a page that is named **Customer Balance**.
2. In the **Main** slot, add a **Container** module. In the module properties pane, set the **Width** value to **Fill Container**. Set the **Children Shown** value to **Two**.
3. In the **Container** slot, add a **Breadcrumb** module. In the module properties pane, under **Links**, configure a link to the account management landing page, and enter **My Account** as the link text.
4. In the **Container** slot, add a **Customer Account Balance** module. In the module properties pane, under **Heading**, enter **Account Balance**.
5. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.
6. Publish the URL for the page.
7. Go to the account management landing page (**My Account**) that you created earlier, click edit.
8. In the properties pane for the **Account customer balance tile** module, add a link to the customer balance page.



Account customer balance tile
account-customer-balance-tile

Heading * Account credit

Links * Account Balance

9. Save and publish the page.

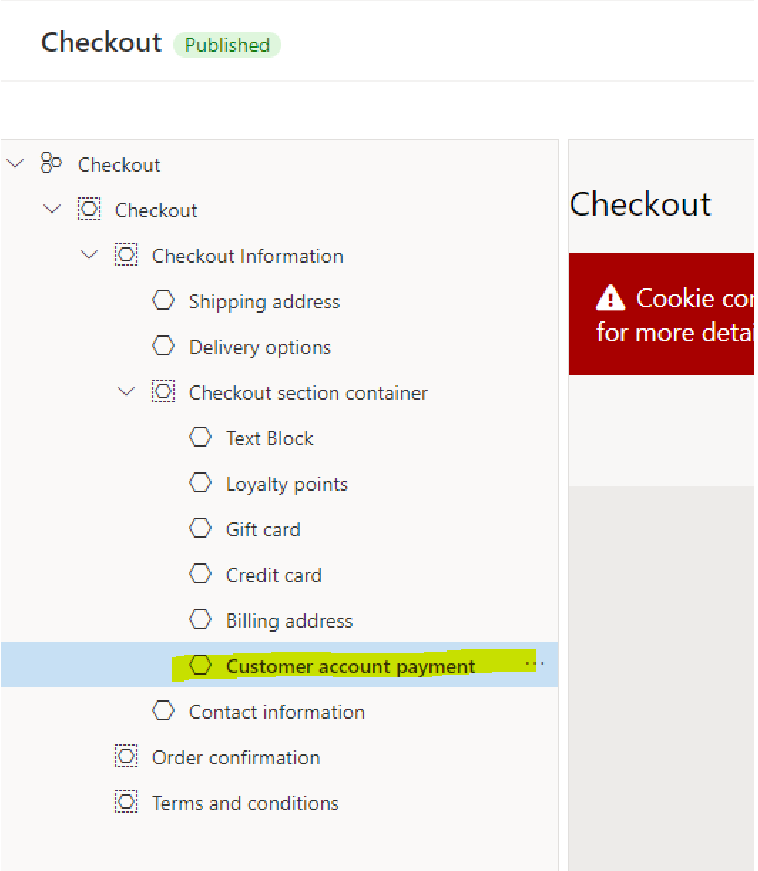
The customer balance page has now been created, and users can access it from the account management landing page.

CUSTOMER BALANCE PAGES – CONFIGURE CHECK OUT PAGE

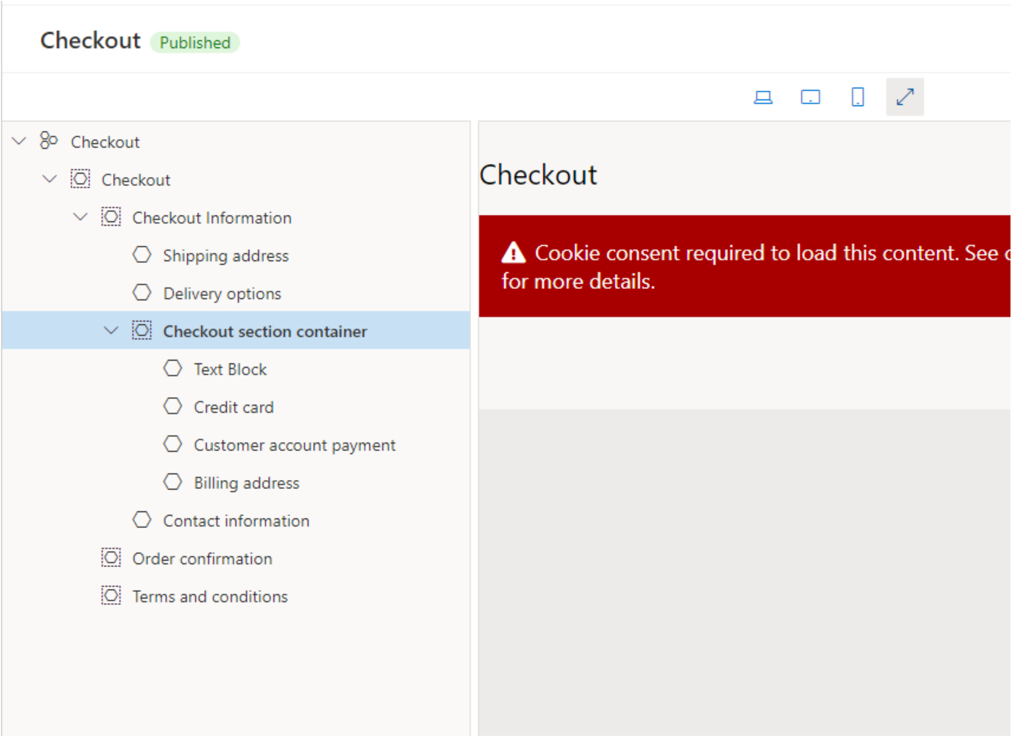
The **Customer Account Payment** module is required to enable the customer balance to be used as a form of payment. This module should be added to the checkout page as a form of payment.

The checkout page uses the “checkout” fragment so it’s the checkout fragment which needs to be modified to allow for customer account payment at checkout.

1. Go to the checkout fragment, click edit
2. In the “Checkout section container” add the customer account payment module
3. Once added, you can see it is the last in the list below billing address, is this how you would like a customer to see it during checkout, sitting below the bill address? Move it up the list to where you think it should go.



4. Since we are setting up a B2B site we have decided not to use the “Gift Card” module so delete this module so it does not show at checkout. If you also decide you don’t want to use Loyalty points in your B2B site, remove this module.
5. Save and publish



Checkout Published

- Checkout
  - Checkout
    - Checkout Information
      - Shipping address
      - Delivery options
      - Checkout section container
        - Text Block
        - Credit card
        - Customer account payment
        - Billing address
      - Contact information
    - Order confirmation
    - Terms and conditions

Checkout

⚠ Cookie consent required to load this content. See c
for more details.

In addition, at **Site Builder > Extensions**, you must make sure that the **Enable customer account payments** property is set to **Enabled for B2B customers**. Note: If this is a new site there might be some other settings you need to set before being able to save and publish the change for "**Enable customer account payments**".

ORDER TEMPLATE PAGES

Two order template pages need to be set up for a B2B e-commerce site: an order templates list page and an order template lines page.

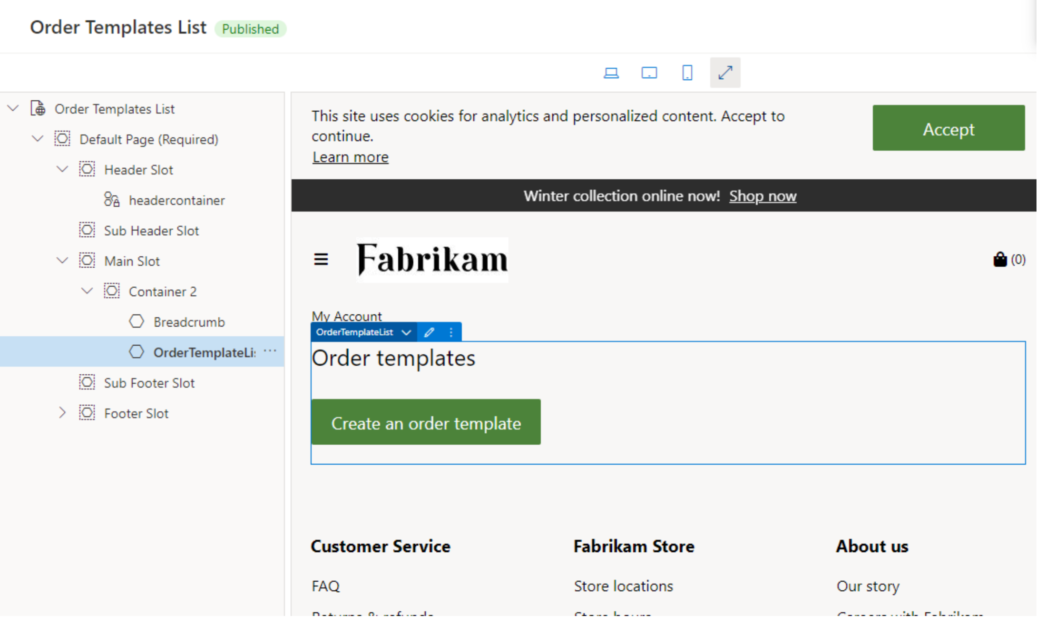
An order templates list page shows a list of all order templates that are available. It's set up by using the **Order templates list** module. The order templates list page lets you create or delete a template and add the items in a template to the cart.

An order template lines page shows the details of the order template that is selected on an order templates list page. It's set up by using the **Order templates lines** module. When a user selects the name of a template on an order templates list page, the order template lines page appears and shows the details of the template. The user can then view and edit the items in the template.

ORDER TEMPLATES LIST PAGE

To create an order templates list page in site builder, follow these steps.

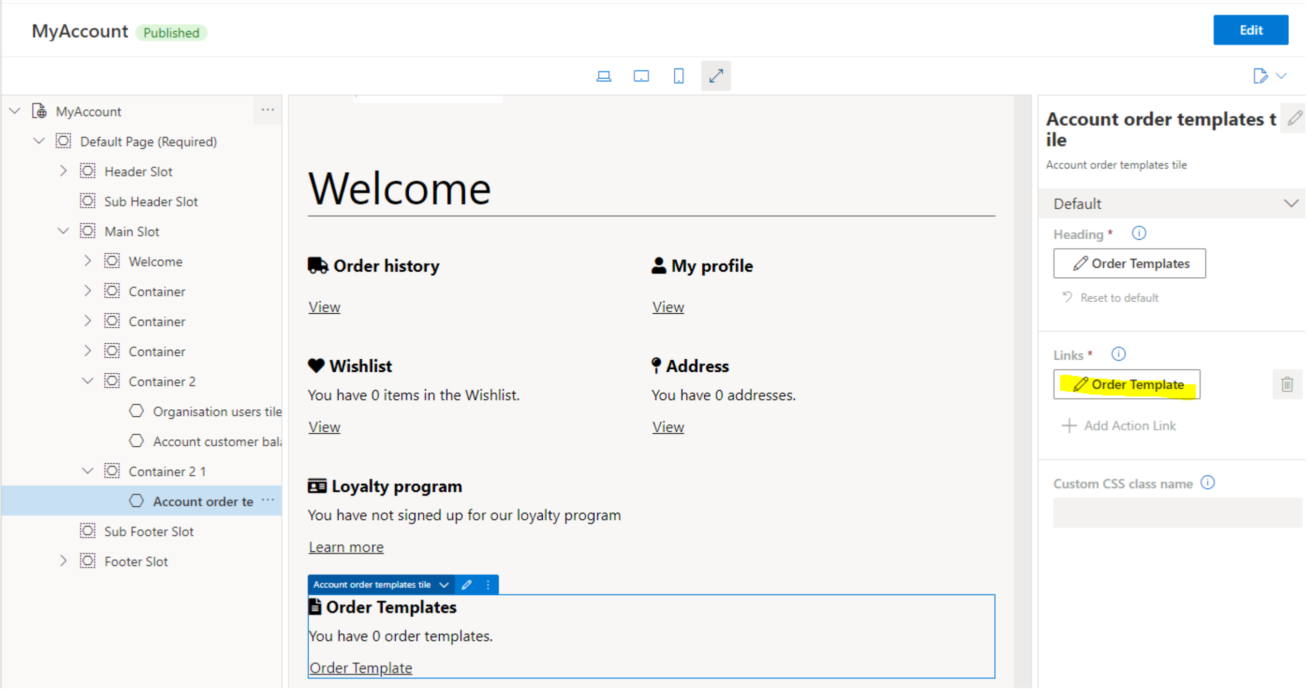
1. Use the **Account management** template that you created earlier to create a page that is named **Order templates list**.
2. In the **Main** slot, add a **Container** module. In the module properties pane, set the **Width** value to **Fill Container**.
3. In the **Container** slot, add a **Breadcrumb** module. In the module properties pane, under **Links**, configure a link to the account management landing page (**MyAccount** page), and enter **My Account** as the link text.
4. In the **Container** slot, add an **Order templates list** module and change name to "Order Templates".
5. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.



This site uses cookies for analytics and personalized content. Accept to continue.

6. Publish the URL for the page.
7. Go to the account management landing page (**My Account**) that you created earlier, click edit.
8. In the properties pane for the **Account order templates tile** module, under **Links**, configure a link to the order templates list page that you just created.

9. Save and publish the page.



Heading * Order Templates

Links * Order Template

The order templates list page has now been created, and users can access it from the account management landing page.

ORDER TEMPLATE LINES PAGE

To create an order template lines page in site builder, follow these steps.

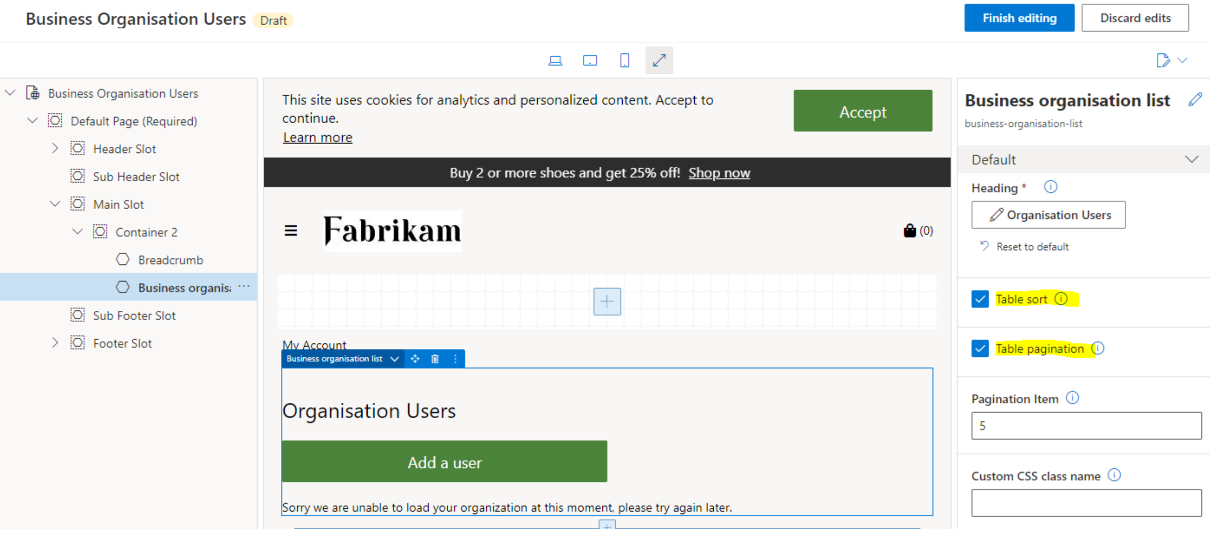
1. Use the **Account management** template that you created earlier to create a page that is named **Order template lines**.
2. In the **Main** slot, add a **Container** module. In the module properties pane, set the **Width** value to **Fill Container**.
3. In the **Container** slot, add a **Breadcrumb** module. In the module properties pane, under **Links**, configure a link to the account management landing page, and enter **My Account** as the link text and a link to my account.
4. In the **Container** slot, add an **Order template lines** module.
5. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.
6. Publish the URL for the page.

ONBOARD BUSINESS PARTNER USERS

The organization users page lets the administrator of a business partner organization onboard additional users from that organization to the B2B e-commerce site. It's set up by using the **Business organization list** module. From the organization users page, an administrator can add or remove users, define account balances, and request statements for a user.

To create an organization users page in site builder, follow these steps.

1. Use the **Account management** template that you created earlier to create a page that is named **Organization Users**.
2. In the **Main** slot, add a **Container** module. In the module properties pane, set the **Width** value to **Fill Container**.
3. In the **Container** slot, add a **Breadcrumb** module. In the module properties pane, under **Links**, configure a link to the account management landing page, and enter **My Account** as the link text with a link to My Account.
4. In the **Container** slot, add a **Business organization list** module. In the module properties pane, under **Heading**, enter **Organization Users**.
5. In the **Business organization list** module properties pane, enable the **Table Sort** and **Table pagination** properties. Set the pagination count to **5**.



This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Buy 2 or more shoes and get 25% off! Shop now

Sorry we are unable to load your organization at this moment, please try again later.

Heading *: Organisation Users

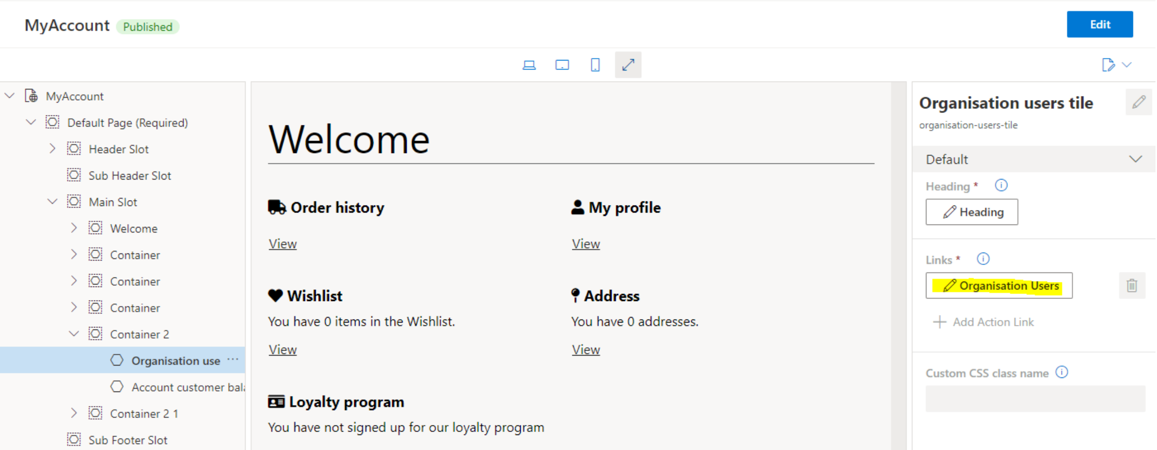
☑ Table sort

☑ Table pagination

Pagination Item: 5

Custom CSS class name:

6. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.
7. Publish the URL for the page.
8. Go to the account management landing page (**My Account**) that you created earlier, click edit.
9. In the properties pane for the **Organization users tile** module, under **Links**, configure a link to the organization users page that you just created.



Welcome

Order history

View

My profile

View

Wishlist

You have 0 items in the Wishlist.

View

Address

You have 0 addresses.

View

Loyalty program

You have not signed up for our loyalty program

Organisation users tile

organisation-users-tile

Default

Heading *: Heading

Links *: Organisation Users

Add Action Link

Custom CSS class name

10. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.

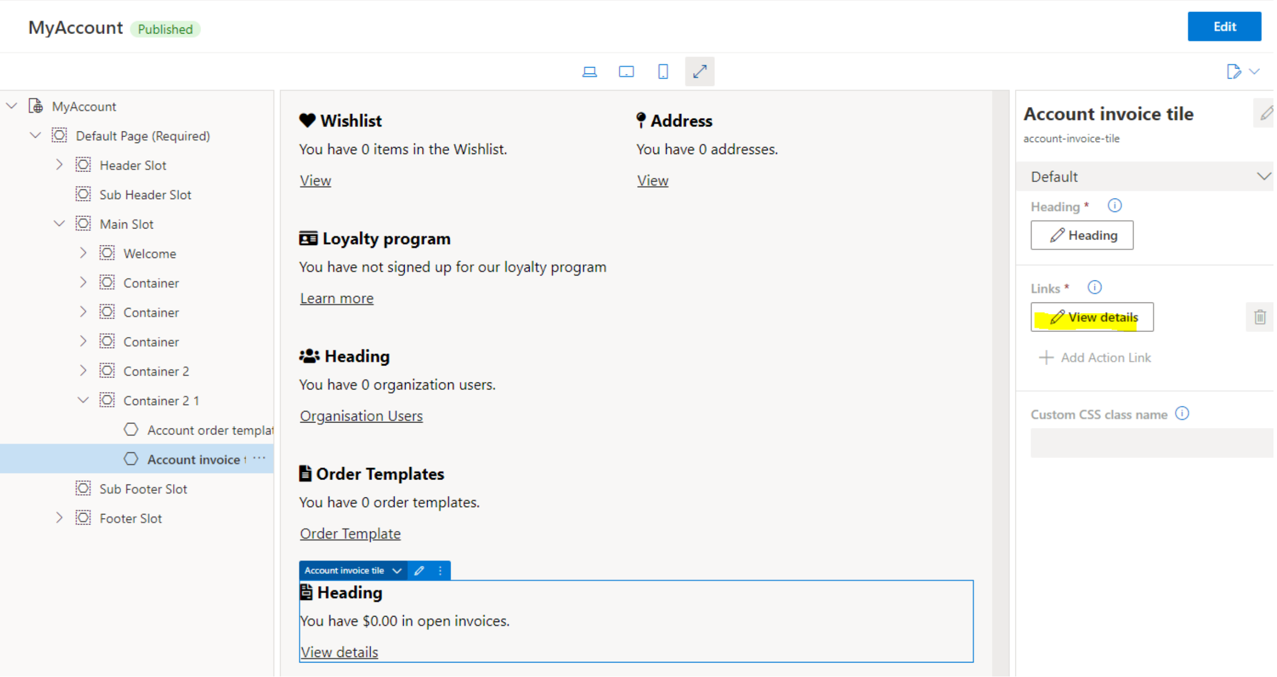
CREATE INVOICE PAGES

An invoices list page shows a list of all available invoices. It's set up by using the **InvoicesList** module. From the invoice list page, users can pay off or request invoices.

An invoice details page shows the details of the invoice that is selected on an invoices list page. It's set up by using the **Invoice details** module. When a user selects an invoice ID on an invoice list page, the invoice details page appears and shows the details of the invoice.

To create an invoices list page in site builder, follow these steps.

1. Use the **Account management** template that you created earlier to create a page that is named **Invoices List**.
2. In the **Main** slot, add a **Container** module. In the module properties pane, set the **Width** value to **Fill Container**.
3. In the **Container** slot, add a **Breadcrumb** module. In the module properties pane, under **Links**, configure a link to the account management landing page, and enter **My Account** as the link text with a link to my account.
4. In the **Container** slot, add an **InvoicesList** module. In the module properties pane, under **Heading**, enter **Invoices**.
5. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.
6. Publish the URL for the page.
7. Go to the account management landing page (**My Account**) that you created earlier, click edit.
8. In the properties pane for the **Account invoices tile** module, under **Links**, configure a link to the invoices list page that you just created and Link text “View details”.

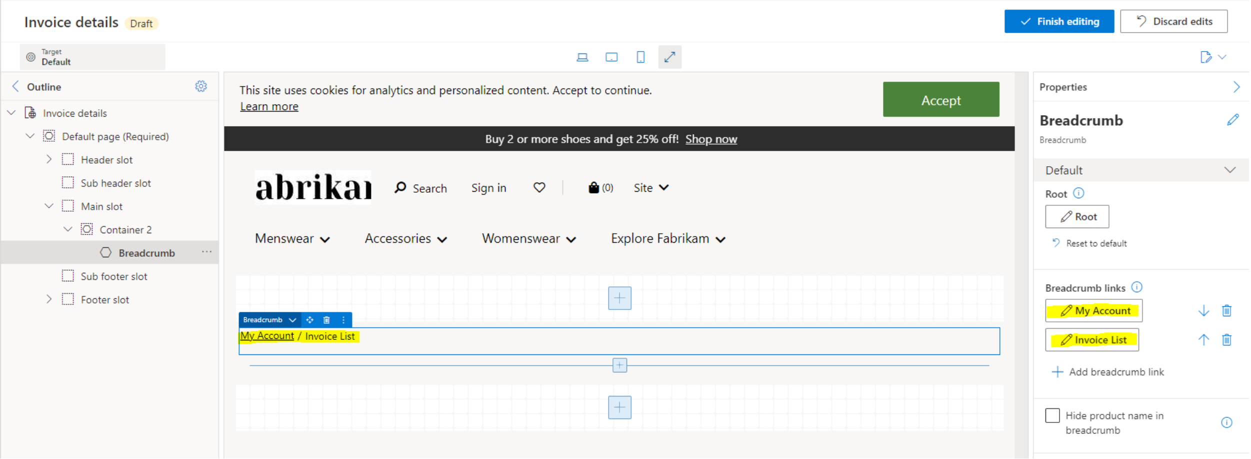


9. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.

CREATE AN INVOICE DETAILS PAGE

To create an invoice details page in site builder, follow these steps.

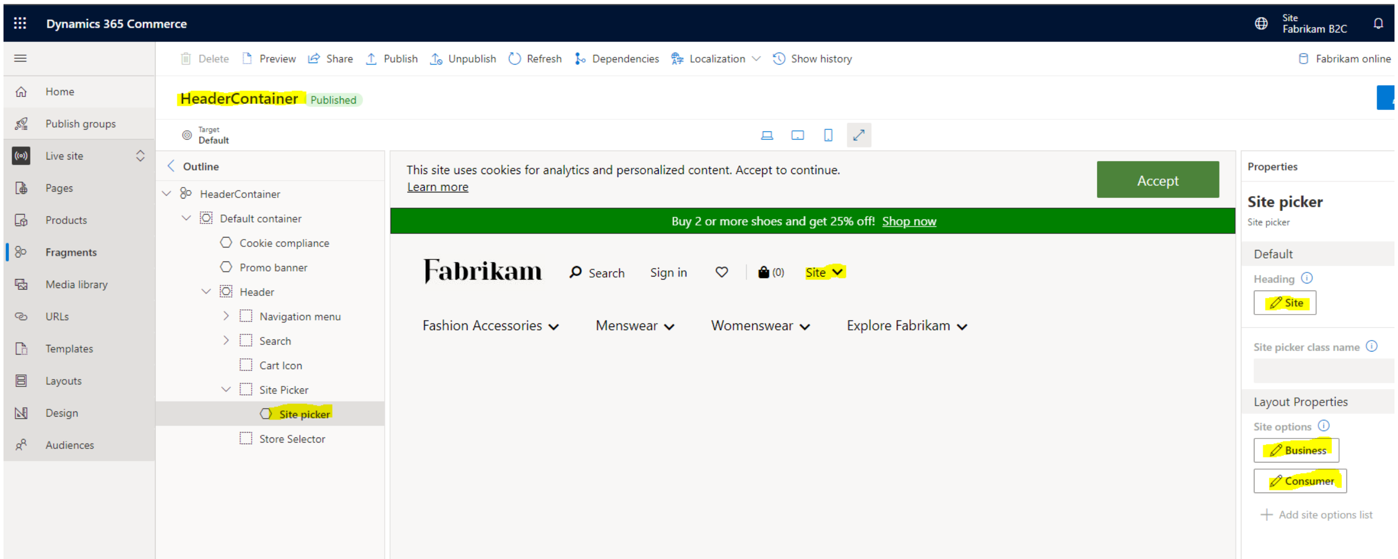
1. Use the **Account management** template that you created earlier to create a page that is named **Invoice Details**.
2. In the **Main** slot, add a **Container** module. In the module properties pane, set the **Width** value to **Fill Container**.
3. In the **Container** slot, add a **Breadcrumb** module. In the module properties pane, under **Breadcrumb Links**, configure a link to the account management landing page, and enter **My Account** as the link text. Then configure a link to the invoices list page and enter **Invoice Lists** as the link text.



4. In the **Container** slot, add an **Invoice details** module.
5. Select **Save**, select **Finish editing** to check in the page, and then select **Publish** to publish it.
6. Publish the URL for the page.

ADD A SITE PICKER MODULE TO A PAGE

The site picker module can be added to the **Site picker** slot of the header module. After a site picker module is added, you can define the module heading and site options. Generally, a header module is contained in a header fragment that can be shared across e-commerce pages for a site.



This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Buy 2 or more shoes and get 25% off! Shop now

OPTIONAL - AZURE ACTIVE DIRECTORY B2C

We now need to set up your Azure Active Directory (Azure AD) business-to-consumer (B2C) tenants for user site authentication in Dynamics 365 Commerce.

Dynamics 365 Commerce uses Azure AD B2C to support user credential and authentication flows. A user can sign up, sign in, and reset their password through these flows. The setup for B2C and B2B site for Azure AD B2C is very similar.

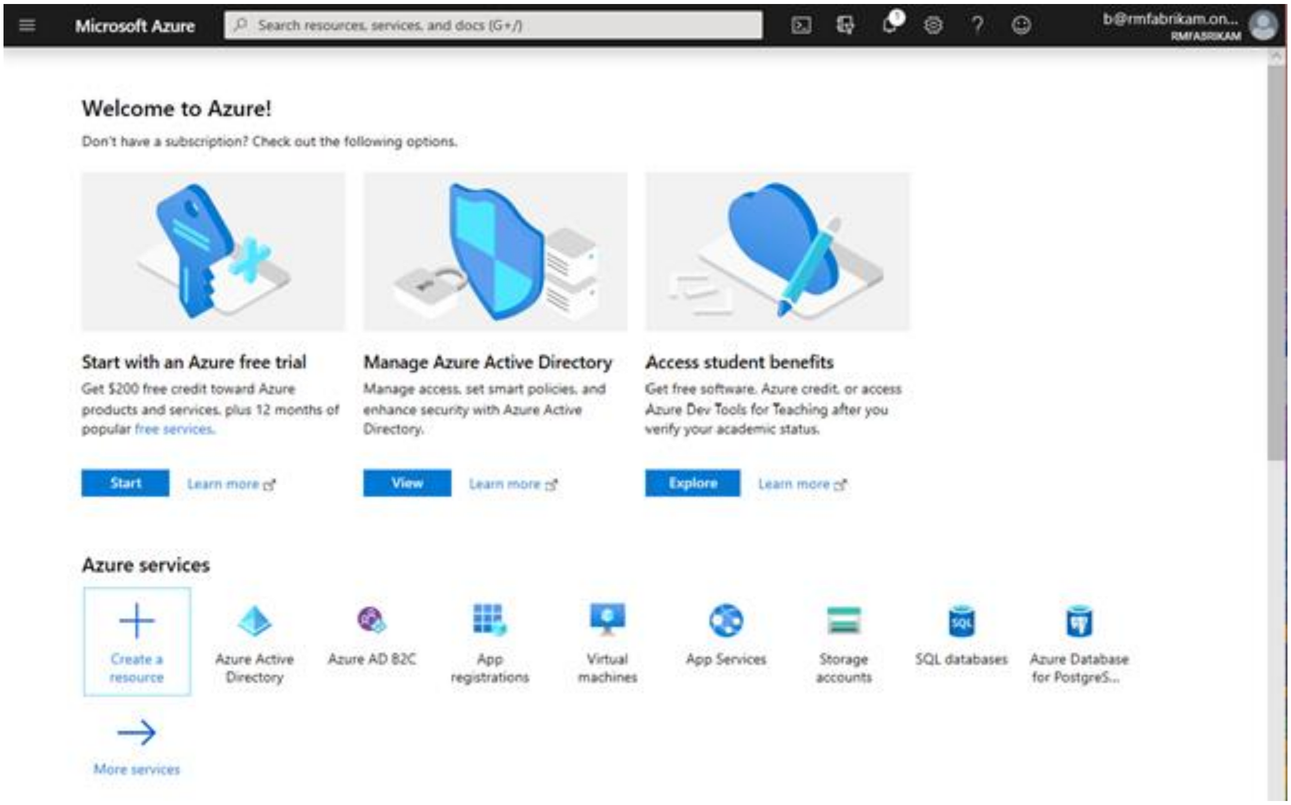
The Azure AD B2C tenant has already been setup for the workshop, the instructor will provide the additional information required to complete this task.

If you would like to use social identity providers, please see our Microsoft Docs site for more information as this will not be covered in this training.

There is a good video on our community page for Azure AD B2C setup. - (2) Unlock the Power of Dynamics 365 Commerce: Setup a B2C Tenant for e-Commerce Site Authentication | December 18, 2020 - Microsoft Dynamics Blog

SETUP AZURE AD B2C

1. Sign in to the Azure portal.
2. From the Azure portal menu, select **Create a resource**. Be sure to use the subscription and directory that will be connected with your Commerce environment.



Welcome to Azure!

Don't have a subscription? Check out the following options.

Start with an Azure free trial

Get $200 free credit toward Azure products and services, plus 12 months of popular free services.

Manage Azure Active Directory

Manage access, set smart policies, and enhance security with Azure Active Directory.

Access student benefits

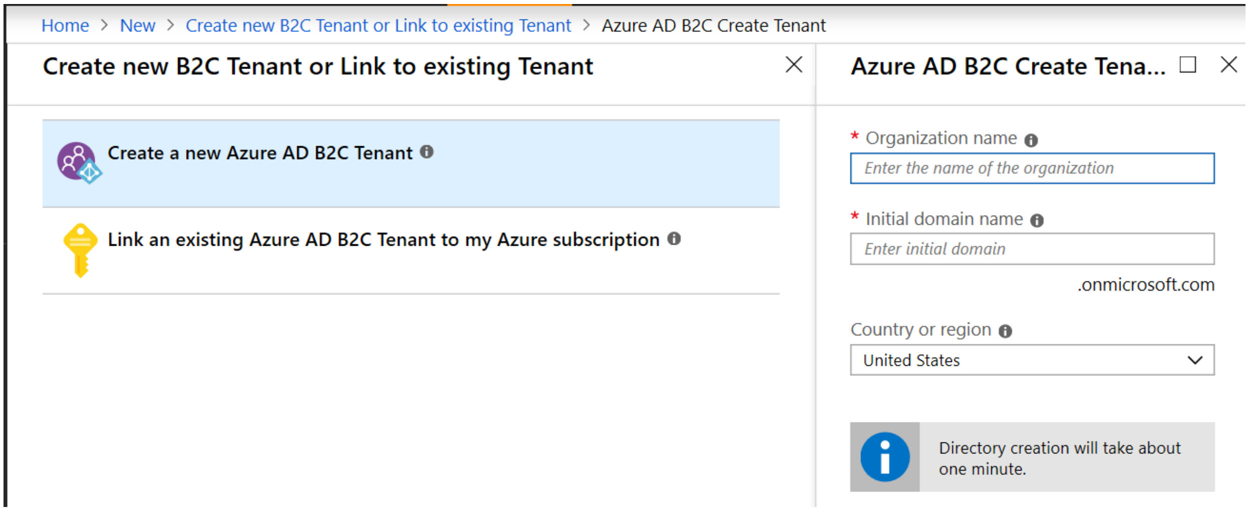
Get free software, Azure credit, or access Azure Dev Tools for Teaching after you verify your academic status.

Azure services

3. In the search box, search **Azure Active Directory B2C** and select.

4. Once on the **Create New B2C Tenant or Link to existing Tenant** page, use one of the options below that best suits your company's needs:

    - **Create a new Azure AD B2C Tenant**: Use this option to create a new Azure AD B2C tenant.
        1. Select **Create a new Azure AD B2C Tenant**.
        2. Under **Organization name**, enter the organization name.
        3. Under **Initial domain name**, enter the initial domain name.
        4. For **Country or region**, select the country or region.
        5. Select **Create** to create the tenant.


Home > New > Create new B2C Tenant or Link to existing Tenant > Azure AD B2C Create Tenant
Create new B2C Tenant or Link to existing Tenant
Create a new Azure AD B2C Tenant
Link an existing Azure AD B2C Tenant to my Azure subscription
Azure AD B2C Create Tena...

* Organization name

Enter the name of the organization

* Initial domain name

Enter initial domain

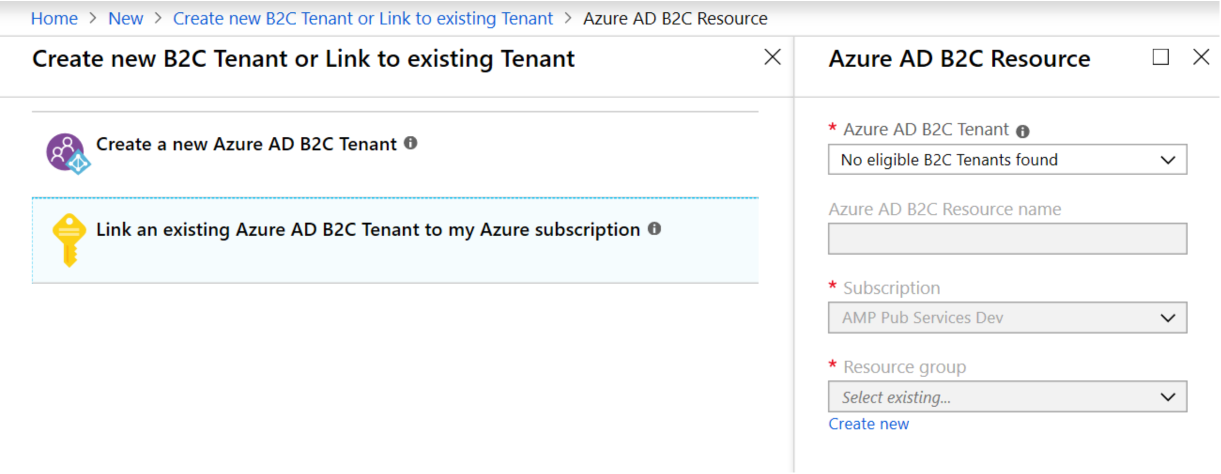
.onmicrosoft.com

Country or region

United States

Directory creation will take about one minute.

    - **Link an existing Azure AD B2C Tenant to my Azure subscription**: Use this option if you already have an Azure AD B2C tenant you want to link to.
        1. Select **Link an existing Azure AD B2C Tenant to my Azure subscription**.
        2. For **Azure AD B2C Tenant**, select the appropriate B2C tenant. If the message "No eligible B2C Tenants found" appears in the selection box, you do not have an existing eligible B2C tenant and will need to create a new one.
        3. For **Resource group**, select **Create new**. Enter a **Name** for the resource group that will contain the tenant, select the **Resource group location**, and then select **Create**.



Home > New > Create new B2C Tenant or Link to existing Tenant > Azure AD B2C Resource

Create new B2C Tenant or Link to existing Tenant

Create a new Azure AD B2C Tenant

Link an existing Azure AD B2C Tenant to my Azure subscription

Azure AD B2C Resource

* Azure AD B2C Tenant: No eligible B2C Tenants found

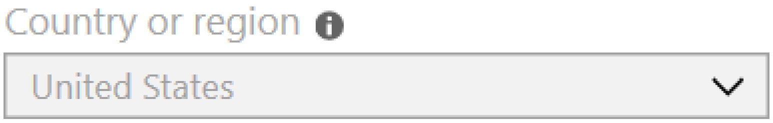
Azure AD B2C Resource name

* Subscription: AMP Pub Services Dev

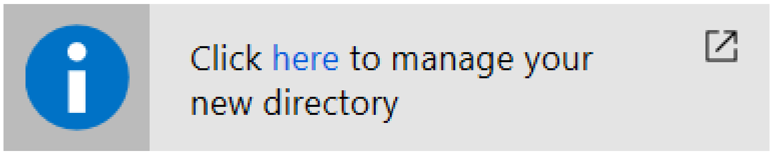
* Resource group: *Select existing...*

Create new

5. Once the new Azure AD B2C directory is created (this may take a few moments), a link to the new directory will appear on the dashboard. This link will direct you to the "Welcome to Azure Active Directory B2C" page.



Country or region: United States

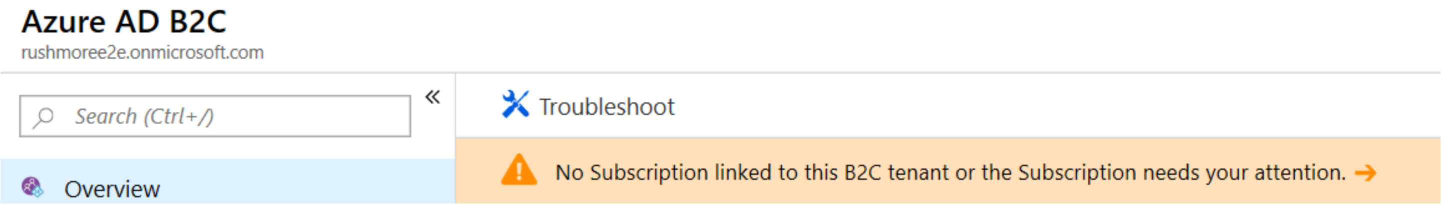


Click here to manage your new directory

Note

If you have multiple subscriptions within your Azure account or have set up the B2C tenant without linking to an active subscription, a **Troubleshoot** banner will direct you to link the tenant to a subscription. Select the troubleshooting message and follow the instructions to resolve the subscription issue.

The following image shows an example of an Azure AD B2C **Troubleshoot** banner.



Azure AD B2C

rushmoree2e.onmicrosoft.com

Search (Ctrl+/)

Overview

Troubleshoot

No Subscription linked to this B2C tenant or the Subscription needs your attention. →

CREATE THE B2C APPLICATION

Once the B2C tenant has been created, you will create a B2C application within your new Azure AD B2C tenant to interact with Commerce.

To create the B2C application, follow these steps.

1. In the Azure portal, select **App registrations**, and then select **New registration**.
2. Under **Name**, enter the name to give this Azure AD B2C application.
3. Under **Supported account types**, select **Accounts in any identity provider or organizational directory (for authenticating users with user flows)**.
4. For **Redirect URI**, enter your dedicated reply URLs as type **Web**. For information on reply URLs and how to format them, see Reply URLs below. A redirect URI/reply URL must be entered to enable redirections from Azure AD B2C back to your site when a user authenticates. The reply URL can be added during the registration process, or can be added later by selecting the **Add a Redirect URI** link from the **Overview** menu in the B2C application's **Overview** section.
5. For **Permissions**, select **Grant admin consent to openid and offline_access permissions**.
6. Select **Register**.
7. Select the newly-created application and navigate to the **API permissions** menu.
8. If a reply URL is entered, under **Implicit grant and hybrid flows** select both the **Access tokens** and **ID tokens** options to enable them for the application, and then select **Save**. If a reply URL was not entered during registration, it can also be added on this page by selecting **Add a platform**, selecting **Web**, and then entering the redirect URI of the application. The **Implicit grant and hybrid flows** section will then be available to select both the **Access tokens** and **ID tokens** options.
9. Go to the **Overview** menu of the Azure portal and copy the **Application (client) ID**. Note this ID for later setup steps (referenced later as the **Client GUID**).

For additional reference on App Registrations in Azure AD B2C, please see The new App registrations experience for Azure Active Directory B2C

REPLY URLS

Reply URLs are important as they provide an allow list of the return domains when your site calls Azure AD B2C to authenticate a user. This permits the return of the authenticated user back to the domain from which they are signing into (your site domain).

In the **Reply URL** box on the **Azure AD B2c - Applications > New application** screen, you need to add separate lines for both your site domain and (once your environment is provisioned) the Commerce-generated URL. These URLs must always use a valid URL format and must be base URLs only (no trailing forward slashes or paths). The string /_msdyn365/authresp then needs to be appended to the base URLs, as in the following examples.

- https://www.fabrikam.com/_msdyn365/authresp (The domain should match the e-commerce domain completely. If you have multiple domains, you need to add this URL for each domain.)
- https://fabrikam-prod.commerce.dynamics.com/_msdyn365/authresp

CREATE USER FLOW POLICIES

User flows are the policies Azure AD B2C uses to provide secure sign in, sign up, edit profile, and forget password user experiences. Dynamics 365 Commerce uses these flows to perform the policy actions to interact with the Azure AD B2C tenant. When a user interacts with these policies, they are redirected to the Azure AD B2C tenant to perform the actions.

Azure AD B2C provides three basic user flow types:

- Sign up and sign in
- Profile editing
- Password reset

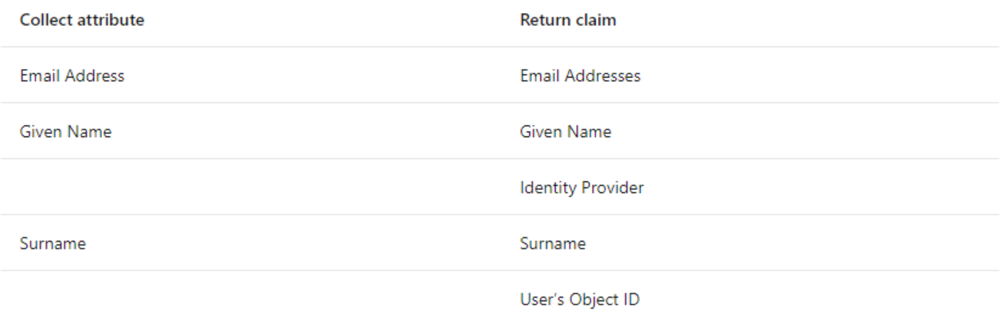
You can choose to use the default user flows provided by Azure AD, which will display a page hosted by Azure AD B2C. Alternately, you can create an HTML page to control the look and feel of these user flow experiences.

To customize the user policy pages with pages built in Dynamics 365 Commerce, see Set up custom pages for user logins. For additional information, see Customize the interface of user experiences in Azure Active Directory B2C.

CREATE A SIGN UP AND SIGN IN USER FLOW POLICY

To create a sign up and sign in user flow policy, follow these steps.

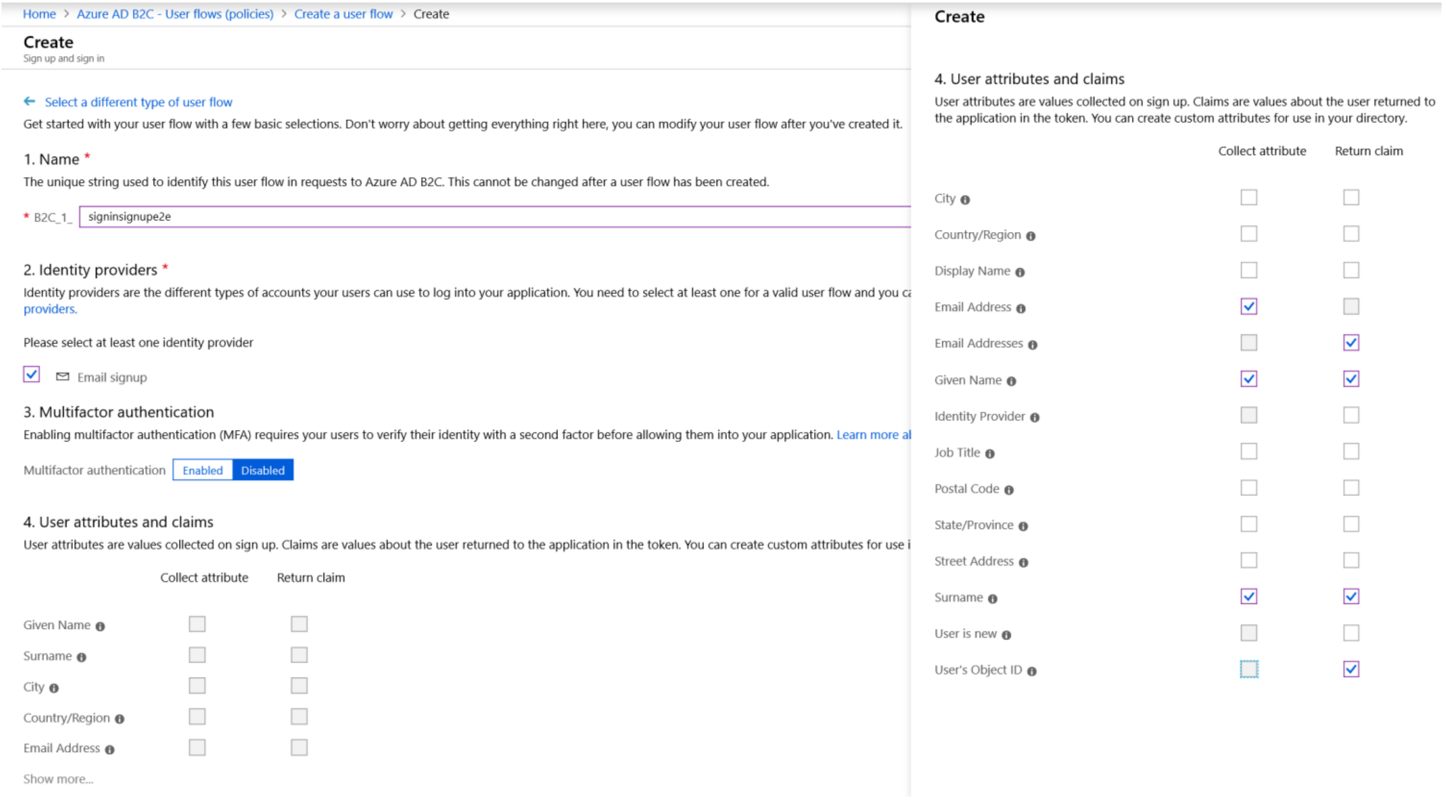
1. In the Azure portal, select **User flows (policies)** in the left navigation pane.
2. On the **Azure AD B2C – User flows (policies)** page, select **New User Flow**.
3. Select the **Sign up and sign in** policy, and then select the **Recommended** version.
4. Under **Name**, enter a policy name. This name will display afterwards with a prefix the portal assigns (for example, "B2C_1_").
5. Under **Identity providers**, in the **Local accounts** section, select **Email signup**. Email authentication is used in most common scenarios for Commerce. If you are also using social identity provider authentication, these can also be selected at this time.
6. Under **Multifactor Authentication**, select the appropriate choice for your company.
7. Under **User attributes and claims**, select options to collect attributes or return claims as appropriate. Select **Show more...** to get the full list of attributes and claims options. Commerce requires the following default options:



Collect attribute	Return claim
Email Address	Email Addresses
Given Name	Given Name
	Identity Provider
Surname	Surname
	User's Object ID

8. Select **Create**.

The following image is an example of the Azure AD B2C sign up and sign in user flow.



Home > Azure AD B2C - User flows (policies) > Create a user flow > Create

Create
Sign up and sign in

← Select a different type of user flow

Get started with your user flow with a few basic selections. Don't worry about getting everything right here, you can modify your user flow after you've created it.

1. Name *

The unique string used to identify this user flow in requests to Azure AD B2C. This cannot be changed after a user flow has been created.

* B2C_1_ signinsignupe2e

2. Identity providers *

Identity providers are the different types of accounts your users can use to log into your application. You need to select at least one for a valid user flow and you ca providers.

Please select at least one identity provider

☑ Email signup

3. Multifactor authentication

Enabling multifactor authentication (MFA) requires your users to verify their identity with a second factor before allowing them into your application. Learn more al

Multifactor authentication Enabled [✓ Disabled]

4. User attributes and claims

User attributes are values collected on sign up. Claims are values about the user returned to the application in the token. You can create custom attributes for use i

	Collect attribute	Return claim
Given Name	□	□
Surname	□	□
City	□	□
Country/Region	□	□
Email Address	□	□

Show more...

Create

4. User attributes and claims

User attributes are values collected on sign up. Claims are values about the user returned to the application in the token. You can create custom attributes for use in your directory.

	Collect attribute	Return claim
City	□	□
Country/Region	□	□
Display Name	□	□
Email Address	☑	□
Email Addresses	□	☑
Given Name	☑	☑
Identity Provider	□	□
Job Title	□	□
Postal Code	□	□
State/Province	□	□
Street Address	□	□
Surname	☑	☑
User is new	□	□
User's Object ID	□	☑

CREATE A PROFILE EDITING USER FLOW POLICY

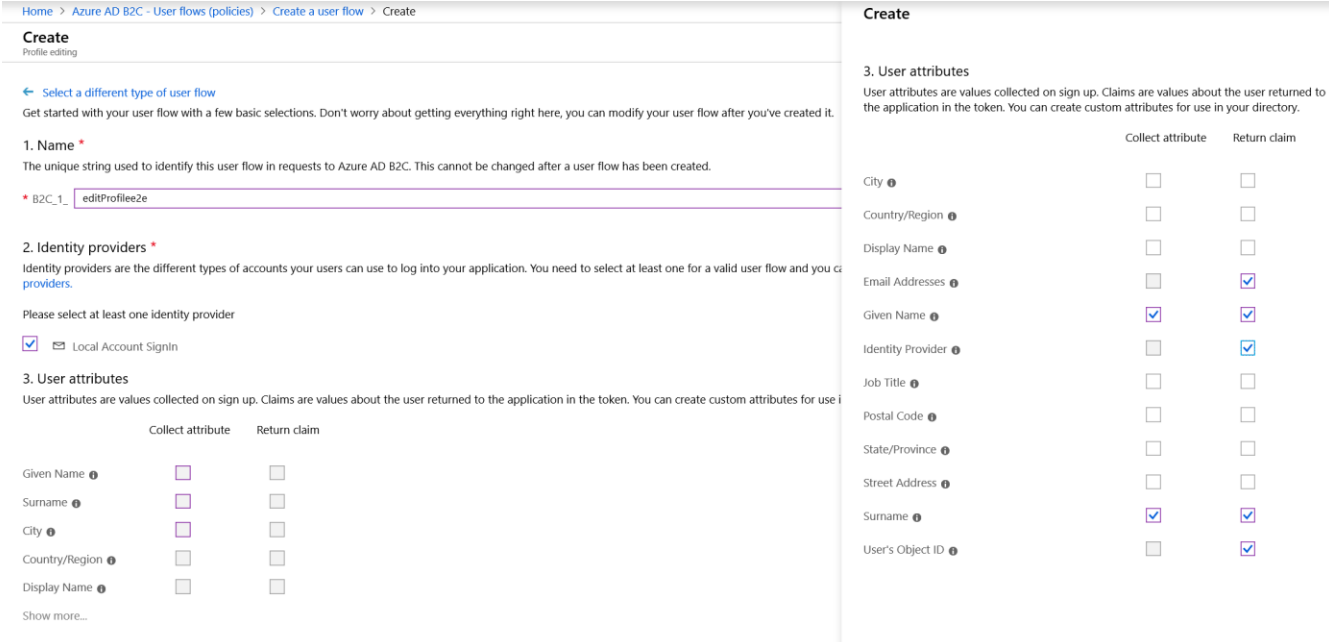
To create a profile editing user flow policy, follow these steps.

1. In the Azure portal, select **User flows (policies)** in the left navigation pane.
2. On the **Azure AD B2C – User flows (policies)** page, select **New User Flow**.
3. Select **Profile editing**, and then select the **Recommended** version.
4. Under **Name**, enter the profile editing user flow. This name will display afterwards with a prefix the portal assigns (for example, "B2C_1_").
5. Under **Identity providers**, in the **Local accounts** section, select **Email SignIn**.
6. Under **User attributes**, select the following check boxes:

Collect attribute	Return claim
	Email Addresses
Given Name	Given Name
	Identity Provider
Surname	Surname
	User's Object ID

7. Select **Create**.

The following image shows an example of the Azure AD B2C profile editing user flow.



Home > Azure AD B2C - User flows (policies) > Create a user flow > Create

Create
Profile editing

← Select a different type of user flow

Get started with your user flow with a few basic selections. Don't worry about getting everything right here, you can modify your user flow after you've created it.

1. Name *

The unique string used to identify this user flow in requests to Azure AD B2C. This cannot be changed after a user flow has been created.

* B2C_1_ editProfilee2e

2. Identity providers *

Identity providers are the different types of accounts your users can use to log into your application. You need to select at least one for a valid user flow and you ca providers.

Please select at least one identity provider

☑ Local Account SignIn

3. User attributes

User attributes are values collected on sign up. Claims are values about the user returned to the application in the token. You can create custom attributes for use i

	Collect attribute	Return claim
Given Name	□	□
Surname	□	□
City	□	□
Country/Region	□	□
Display Name	□	□

Show more...

Create

3. User attributes

User attributes are values collected on sign up. Claims are values about the user returned to the application in the token. You can create custom attributes for use in your directory.

	Collect attribute	Return claim
City	□	□
Country/Region	□	□
Display Name	□	□
Email Addresses	□	☑
Given Name	☑	☑
Identity Provider	□	☑
Job Title	□	□
Postal Code	□	□
State/Province	□	□
Street Address	□	□
Surname	☑	☑
User's Object ID	□	☑

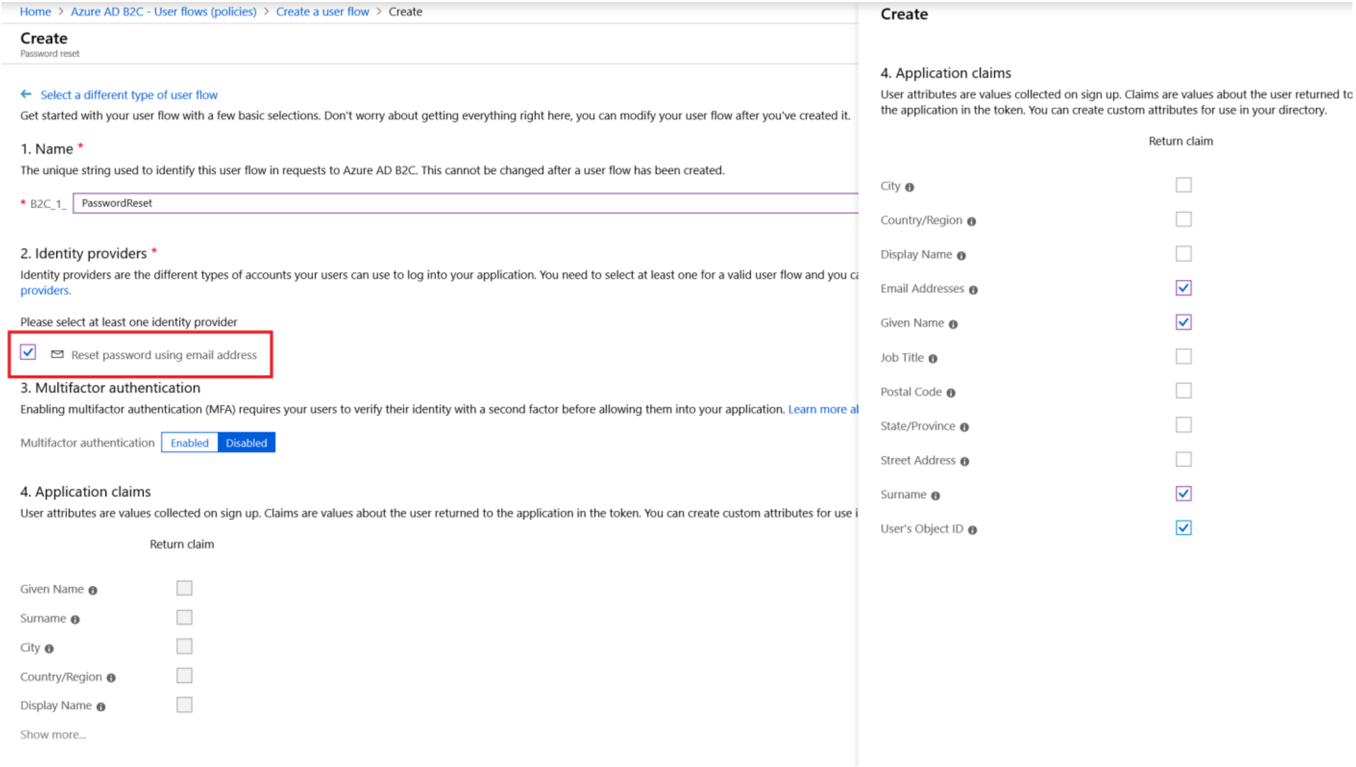
CREATE A PASSWORD RESET USER FLOW POLICY

To create a password reset user flow policy, follow these steps.

1. In the Azure portal, select **User flows (policies)** in the left navigation pane.
2. On the **Azure AD B2C – User flows (policies)** page, select **New User Flow**.
3. Select **Password Reset**, and then select the **Recommended** version.
4. Under **Name**, enter a name for the password reset user flow.
5. Under **Identity providers**, select **Reset password using email address**.
6. Select **Create**.

7. Under **Application claims**, select the following check boxes:
   - **Email addresses**
   - **Given Name**
   - **Surname**
   - **User's Object ID**
8. Select **Create**.

The following image shows where to set **Reset Password using mail address** in the Azure AD B2C password reset user flow.



Home > Azure AD B2C - User flows (policies) > Create a user flow > Create

Create
Password reset

← Select a different type of user flow

Get started with your user flow with a few basic selections. Don't worry about getting everything right here, you can modify your user flow after you've created it.

1. Name *

The unique string used to identify this user flow in requests to Azure AD B2C. This cannot be changed after a user flow has been created.

* B2C_1_ PasswordReset

2. Identity providers *

Identity providers are the different types of accounts your users can use to log into your application. You need to select at least one for a valid user flow and you ca providers.

Please select at least one identity provider

☑ ✉ Reset password using email address

3. Multifactor authentication

Enabling multifactor authentication (MFA) requires your users to verify their identity with a second factor before allowing them into your application. Learn more al

Multifactor authentication: Enabled [✓ Disabled]

4. Application claims

User attributes are values collected on sign up. Claims are values about the user returned to the application in the token. You can create custom attributes for use i

	Return claim
Given Name ❶	□
Surname ❶	□
City ❶	□
Country/Region ❶	□
Display Name ❶	□

Show more...

Create

4. Application claims

User attributes are values collected on sign up. Claims are values about the user returned to the application in the token. You can create custom attributes for use in your directory.

	Return claim
City ❶	□
Country/Region ❶	□
Display Name ❶	□
Email Addresses ❶	☑
Given Name ❶	☑
Job Title ❶	□
Postal Code ❶	□
State/Province ❶	□
Street Address ❶	□
Surname ❶	☑
User's Object ID ❶	☑

UPDATE COMMERCE HEADQUARTERS WITH THE NEW AZURE AD B2C INFORMATION

Once the Azure AD B2C provisioning steps above are completed, the Azure AD B2C application must be registered in your Dynamics 365 Commerce environment.

To update headquarters with the new Azure AD B2C information, follow these steps.

1. In Commerce, go to **Commerce Shared Parameters** and select **Identity Providers** in the left menu.
2. Under **Identity Providers**, do the following:
    1. In the **Issuer** box, enter the identity provider issuer URL. To find your issuer URL, see next topic.
    2. In the **Name** box, enter a name for your issuer record.
    3. In the **Type** box, enter **Azure AD B2C (id_token)**.
3. Under **Relying Parties**, with the above B2C identity provider item selected, do the following:
    1. In the **ClientID** box, enter your B2C application ID. You can find this in the **Application ID** box of your B2C application's properties page.
    2. In the **Type** box, enter **Public**.
    3. In the **User Type** box, enter **Customer**.
4. On the action pane, select **Save**.
5. In the Commerce search box, search for **Distribution schedule**
6. In the left navigation menu of the **Distribution schedules** page, select job **1110 Global configuration**.
7. On the action pane, select **Run Now**.

OBTAIN ISSUER URL

To obtain your identity provider issuer URL, follow these steps.

1. On the Azure AD B2C page of the Azure portal, navigate to your **Sign up and sign in** user flow.
2. Select **Page layouts** in the left navigation menu, under **Layout name** select **Unified sign up or sign in page**, and then select **Run user flow**.
3. Make sure your application is set to your intended Azure AD B2C application created above, and then select the link under the **Run user flow** header that includes .../.well-known/openid-configuration?p=<B2CSIGN-INPOLICY>.
4. A metadata page is displayed in your browser tab. Copy the identity provider issuer URL (the value for **"issuer"**).
    - Example: https://login.fabrikam.com/011115c3-0113-4f43-b5e2-df01266e24ae/v2.0/.

OR: To construct the same metadata URL manually, do the following steps.

1. Create a metadata address URL in the following format using your B2C tenant and policy: https://<B2CTENANTNAME>.b2clogin.com/<B2CTENANTNAME>.onmicrosoft.com/v2.0/.well-known/openid-configuration?p=<B2CSIGN-INPOLICY>
    - Example: https://d365plc.b2clogin.com/d365plc.onmicrosoft.com/v2.0/.well-known/openid-configuration?p=B2C_1_signinup.
2. Enter the metadata address URL into a browser address bar.
3. In the metadata, copy the identity provider issuer URL (the value for **"issuer"**).

CONFIGURE YOUR B2C TENANT IN COMMERCE SITE BUILDER

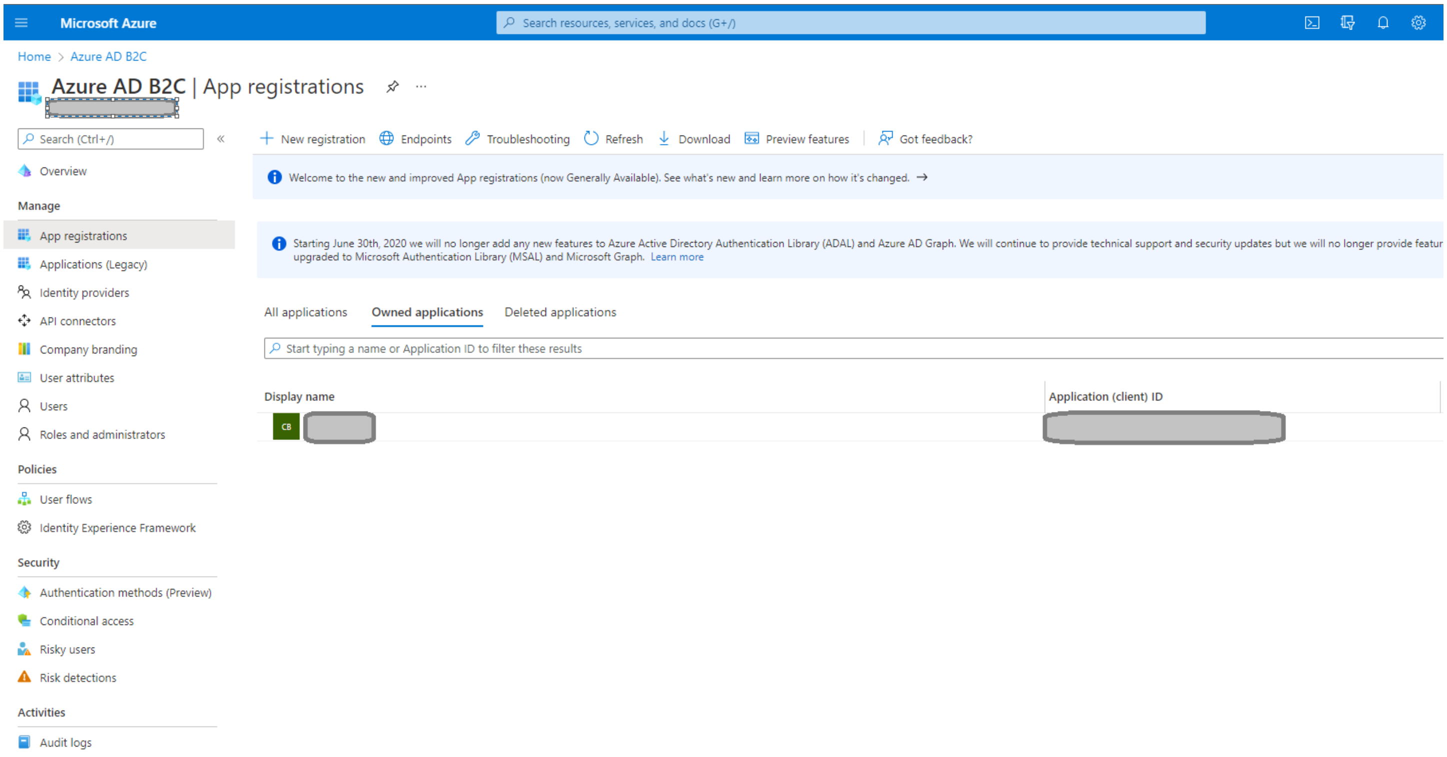
Once setup of your Azure AD B2C tenant is completed, you must configure the B2C tenant in Commerce site builder. Configuration steps include collecting B2C application information from the Azure portal, entering that B2C application information into site builder, and then associating the B2C application with your site and channel.

COLLECT THE REQUIRED APPLICATION INFORMATION

To collect the required application information, follow these steps.

1. In the Azure portal, go to **Home > Azure AD B2C – App registrations**.
2. Select your application, copy **Application (client) ID** information as this will later be entered as the **Client GUID** in site builder.
3. Under **Reply URL**, collect the reply URL.
4. Go to **Home > Azure AD B2C – User flows** and then collect the names of each user flow policy.

The following image shows an example of the **Azure AD B2C - App registration** page.



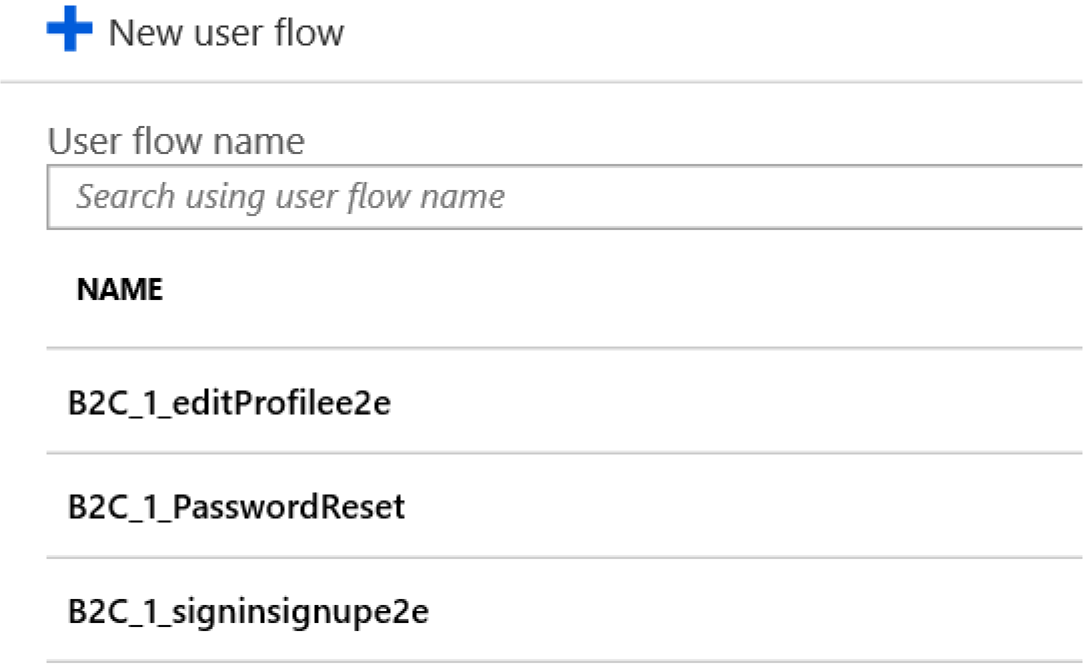
[redacted]

Welcome to the new and improved App registrations (now Generally Available). See what's new and learn more on how it's changed. →

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to provide technical support and security updates but we will no longer provide featur
upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. Learn more

Display name	Application (client) ID
CB [redacted]	[redacted]

The following image shows an example of user flow policies on the **Azure AD B2C – User flows** page.



B2C_1_editProfilee2e

B2C_1_PasswordReset

B2C_1_signinsignupe2e

ENTER YOUR AZURE AD B2C TENANT APPLICATION INFORMATION INTO COMMERCE

You must enter details of the Azure AD B2C tenant into Commerce site builder before associating the B2C tenant with your site(s).

To add your Azure AD B2C tenant application information to Commerce, follow these steps.

1. Sign in as an administrator to Commerce site builder for your environment.
2. In the left navigation pane, select **Tenant Settings** to expand it.
3. Under **Tenant Settings**, select **Site authentication setup**.
4. In the main window next **Site authentication profile**, select **Manage**. (If your tenant appears in the B2C Applications list, then it was already added by an administrator. Verify that the items in step 6 below match your B2C Application.)
5. Select **Add B2C Application**.
6. Enter the following required items in the form displayed, using values from your B2C tenant and application. Fields that are not required (those without an asterisk) may be left blank.
   - **Application Name**: The name for your B2C Application, for example "Fabrikam B2C".
   - **Tenant Name**: The name of your B2C tenant (for example, use "fabrikam" if the domain appears as "fabrikam.onmicrosoft.com" for the B2C tenant).
   - **Forget Password Policy ID**: The forget password user flow policy ID, for example "B2C_1_PasswordReset".
   - **Signup Signin Policy ID**: The sign up and sign in user flow policy ID, for example "B2C_1_signup_signin".

- **Client GUID**: The B2C application ID, for example "22290eb2-c52e-42e9-8b35-a2b0a3bcb9e6".
    - **Edit Profile Policy ID**: The profile editing user flow policy ID, for example "B2C_1A_ProfileEdit".
7. Select **OK**. You should now see the name of your B2C application appear in the list.
8. Select **Save** to save your changes.

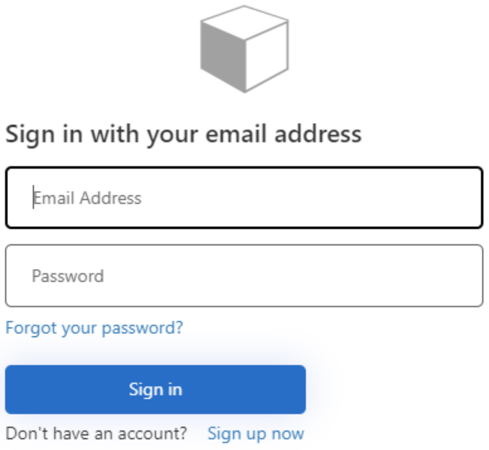
ASSOCIATE THE B2C APPLICATION TO YOUR SITE AND CHANNEL

To associate the B2C application to your site and channel, follow these steps.

1. Navigate to your site in Commerce site builder.
2. In the left navigation pane, select **Site Settings** to expand it.
3. Below **Site Settings**, select **Channels**.
4. In the main window under **Channels**, select your channel.
5. In the channel properties pane on the right, select your B2C application name from the **Select B2C Application** drop-down menu.
6. Select **Close**, and then select **Save and Publish**.

SET UP CUSTOM PAGES FOR USER SIGN-INS

You have options of using the standard out of the box Azure AD B2C functionality which will look like the picture below or you can create your own customer pages for user sign-in.


Sign in with your email address

If you would like more information on setting up custom pages for user sign-ins please see our doc site.

[Set up custom pages for user sign-ins - Commerce | Dynamics 365 | Microsoft Docs]

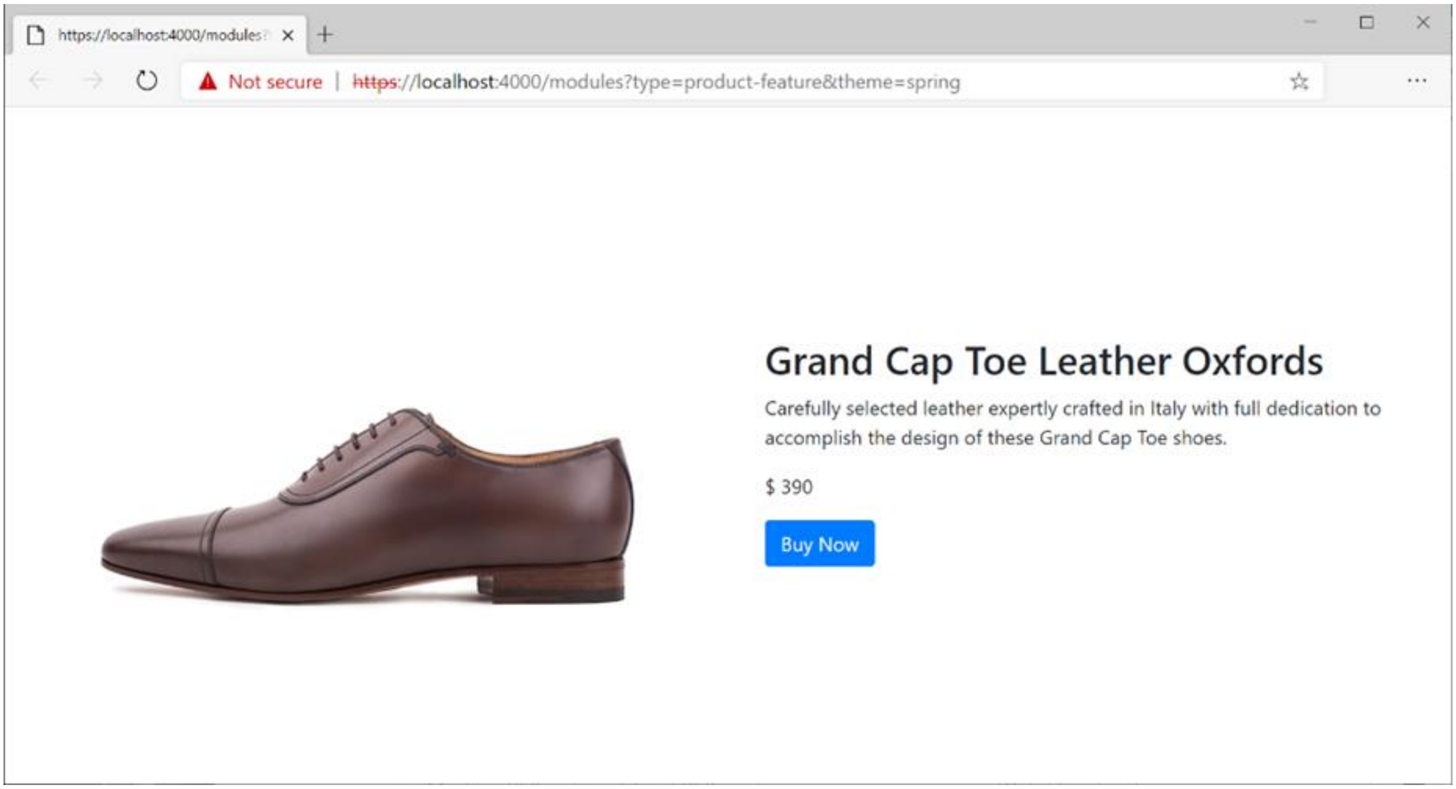
UNIT 3 EXTENSIBILITY – CUSTOM MODULES AND THEMING

This next section involves setting up a development environment and walks through how to create a custom module, leverage a data action and build a new site theme than can be packaged up and uploaded to a site.

E-Commerce pages are made up of various 'modules' that use 'data actions' to get data from the CMS system, Commerce Scale Unit and other services to render appropriate html that make up a page. The Commerce online store SDK provides support to build custom modules, data actions and themes which contain module styling (SCSS) and module view overrides.

Our end goal will be to:

1. Create a custom product "Feature" module which will be configurable with product information (title, description, price and image) coming from the Dynamics Commerce catalog looking similar to below image.


https://localhost:4000/modules?
Not secure
https://localhost:4000/modules?type=product-feature&theme=spring

Grand Cap Toe Leather Oxfords

Carefully selected leather expertly crafted in Italy with full dedication to accomplish the design of these Grand Cap Toe shoes.

$ 390

Buy Now

2. Create a new site theme.
3. Create an e-Commerce extension package that can be uploaded to LCS and deploy to an environment

Familiarity with the following technology will be helpful but you should be able to follow along without full knowledge:

- Node and package management via NPM
- React
- JavaScript / Typescript
- SCSS for theme development (or CSS)

SETTING UP A DEVELOPMENT ENVIRONMENT AND SDK

Online documentation can be found here to set up a development environment or by following the below steps.

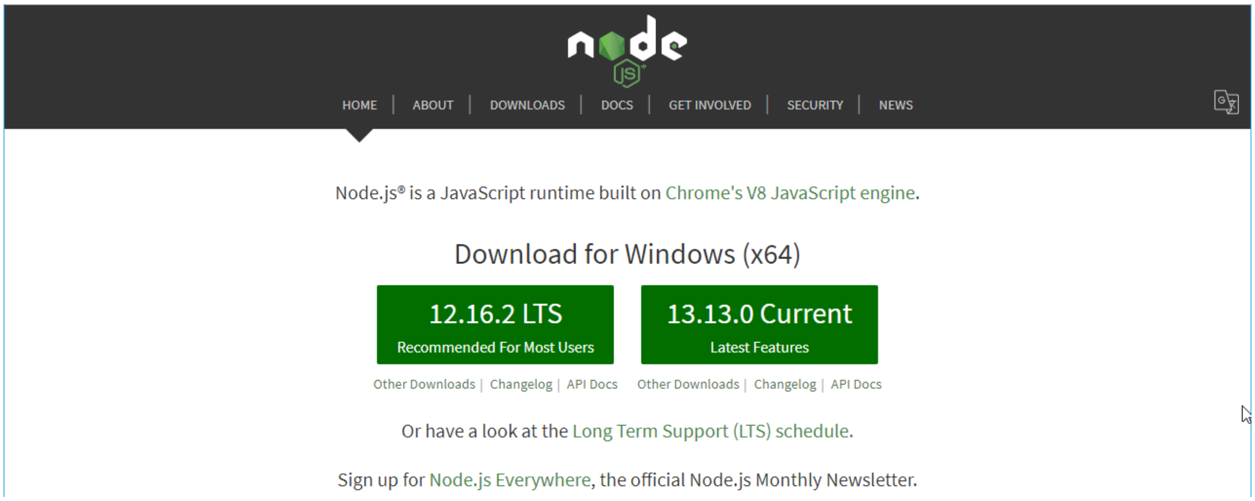
Setting up a development environment involves installing three freely available tools (Visual Studio Code, Node.js, Yarn) and installing the Dynamics Commerce online SDK (by cloning a Git repository or downloading the zip file).

The SDK includes CLI commands (commands to create modules, data actions theme, build extension package, …), APIs (globalization, telemetry, …) and the store starter kit (SSK) which is a full set of pre-built modules, data actions and themes to run an e-Commerce front end site. Note the order of installing the below tools is not important.

INSTALLING NODE.JS

Node.js is a JavaScript runtime built on Chrome's V8 JavaScript Engine.

Visit https://nodejs.org to find out more. Click on the 12.x version as shown below.



Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Download for Windows (x64)

HOME
ABOUT
DOWNLOADS
DOCS
GET INVOLVED
SECURITY
NEWS
12.16.2 LTS
Recommended For Most Users
13.13.0 Current
Latest Features
Other Downloads | Changelog | API Docs
Other Downloads | Changelog | API Docs

Or have a look at the Long Term Support (LTS) schedule.

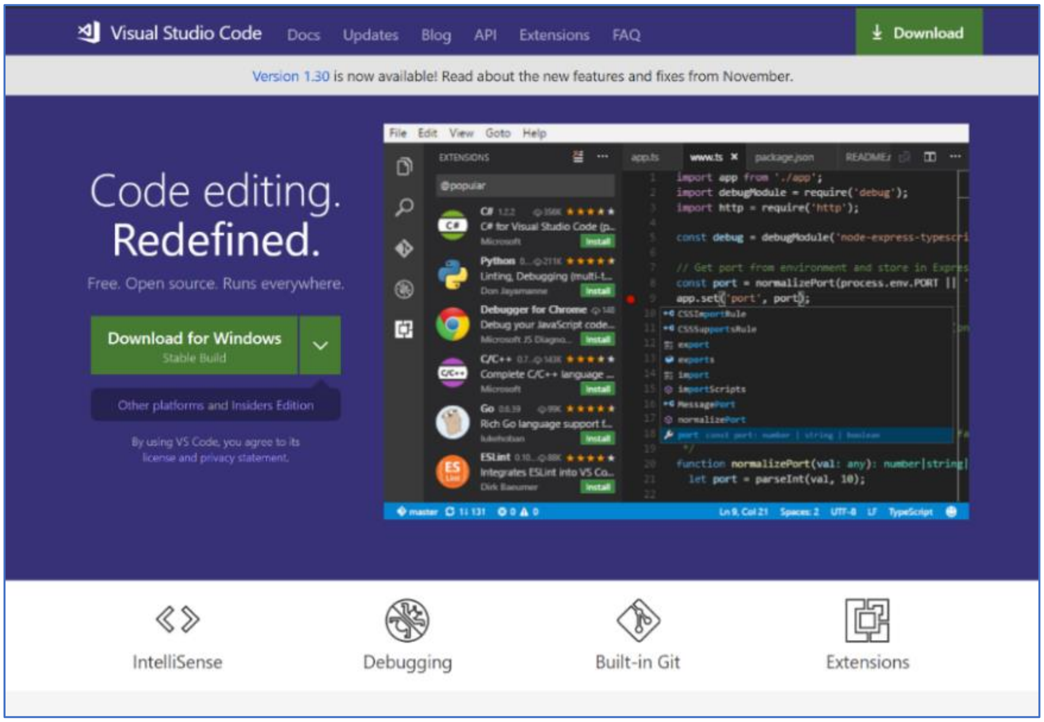
Sign up for Node.js Everywhere, the official Node.js Monthly Newsletter.

Once downloaded, launch the installer and hit the "Next" button (and accept the License Agreement) until you see the "Install" button, you should be able to leave all values at their defaults.

INSTALLING VISUAL STUDIO CODE

Visual Studio Code is a lightweight source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built in support for JavaScript, TypeScript and Node.js.

Navigate to https://code.visualstudio.com and download the latest stable build. Once downloaded launch the installer and you should be able to leave all values at their defaults during the install and accept the user license agreement.



Version 1.30 is now available! Read about the new features and fixes from November.

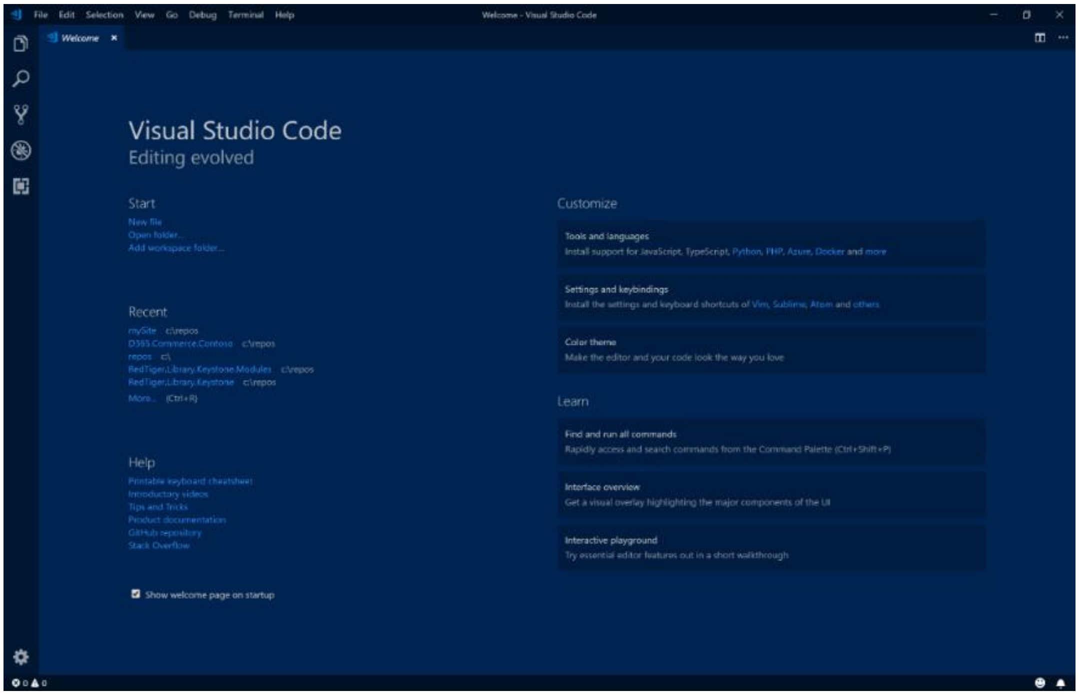
Code editing. Redefined.

Free. Open source. Runs everywhere.

By using VS Code, you agree to its license and privacy statement.

```
import app from './app';
import debugModule = require('debug');
import http = require('http');

const debug = debugModule('node-express-typescri

// Get port from environment and store in Expres
const port = normalizePort(process.env.PORT || '
app.set('port', port);

function normalizePort(val: any): number|string|
  let port = parseInt(val, 10);
```

When installation is complete it will auto launch and should look similar to the below image.



Visual Studio Code
Editing evolved

Start
New file
Open folder...
Add workspace folder...

Recent
mySite c:\repos
D365.Commerce.Contoso c:\repos
repos c:\
RedTiger.Library.Keystone.Modules c:\repos
RedTiger.Library.Keystone c:\repos
More... (Ctrl+R)

Help
Printable keyboard cheatsheet
Introductory videos
Tips and Tricks
Product documentation
GitHub repository
Stack Overflow

☑ Show welcome page on startup

Customize

Tools and languages
Install support for JavaScript, TypeScript, Python, PHP, Azure, Docker and more

Settings and keybindings
Install the settings and keyboard shortcuts of Vim, Sublime, Atom and others

Color theme
Make the editor and your code look the way you love

Learn

Find and run all commands
Rapidly access and search commands from the Command Palette (Ctrl+Shift+P)

Interface overview
Get a visual overlay highlighting the major components of the UI

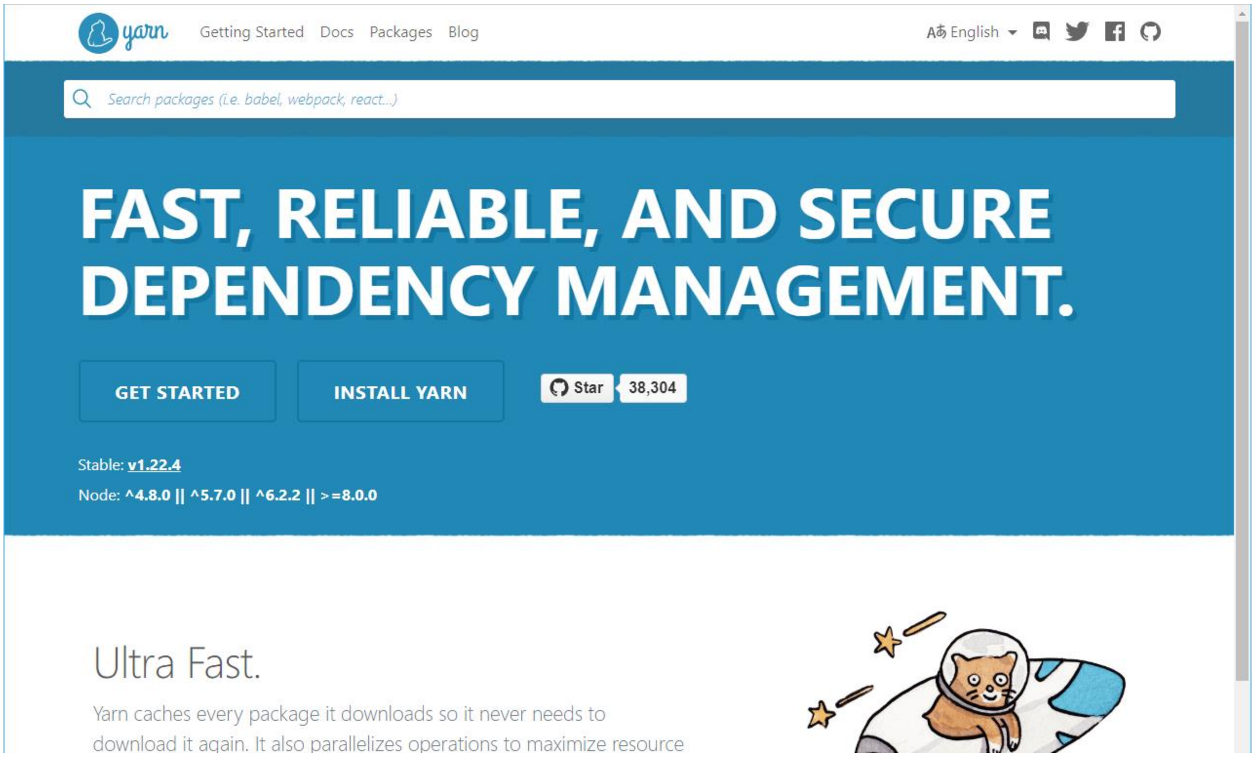
Interactive playground
Try essential editor features out in a short walkthrough

INSTALLING YARN

Yarn is a dependency management tool that will ensure you have all the latest packages needed for your Dynamics 365 Commerce Extensibility needs.

Navigate to https://classic.yarnpkg.com download the latest 1.x stable build. Once downloaded launch the installer and hit the "Next" button (and accept the License Agreement) until you see the "Install" button, you should be able to leave all values at their defaults.

The below screen shot may differ slightly from what you see depending on versions at time of installation and environment you are working with.



FAST, RELIABLE, AND SECURE DEPENDENCY MANAGEMENT.

Stable: v1.22.4

Node: ^4.8.0 || ^5.7.0 || ^6.2.2 || >=8.0.0

Ultra Fast.

Yarn caches every package it downloads so it never needs to download it again. It also parallelizes operations to maximize resource

INSTALLING GIT (OPTIONAL)

Git is a free and open source distributed version control system. We will be using git to clone our SDK/Starter Kit code but optionally you can just download the zip file for the SDK in a later step instead of using Git. Navigate to https://git-scm.com/downloads followed by downloading and installing the latest build. You should be able to accept all the default install values.

INSTALLING DYNAMICS 365 COMMERCE ONLINE SDK/STARTER KIT

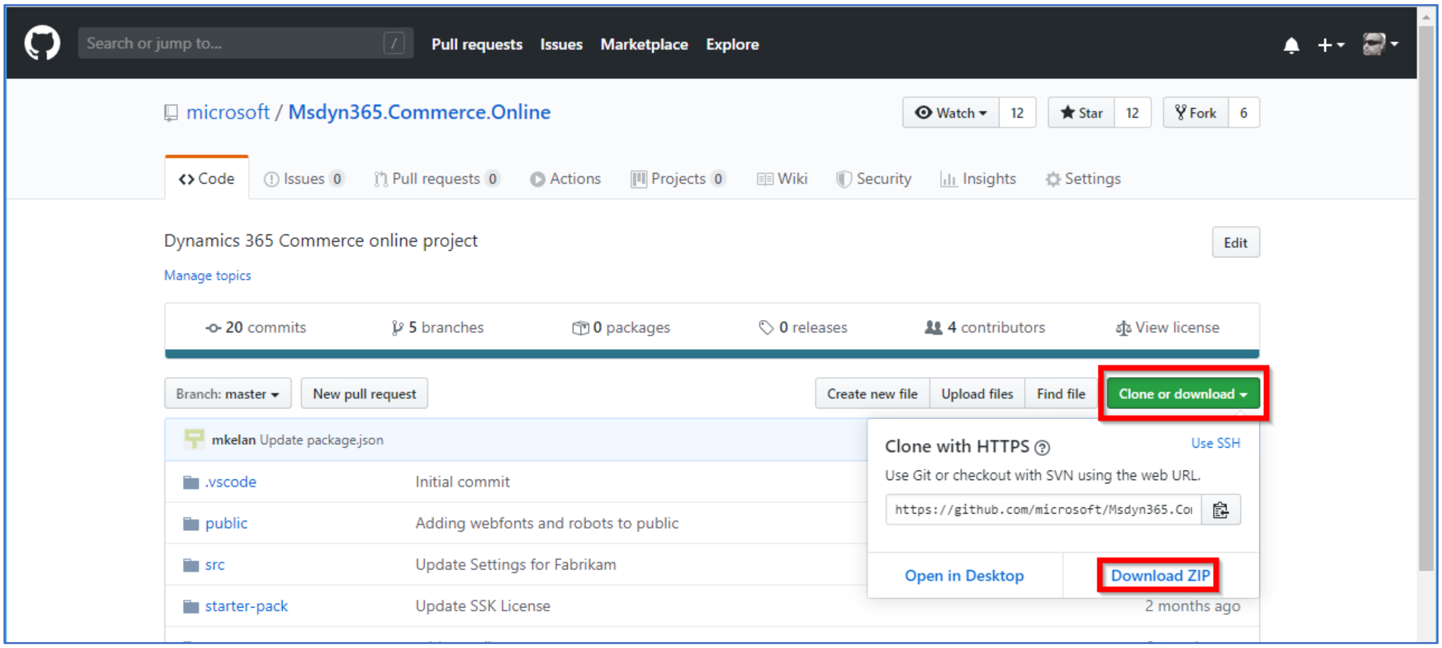
The SDK will give you everything you need to extend your e-Commerce site including the ability to create new modules, data actions and themes.

The SDK configuration package is available through the following GitHub repository: https://github.com/microsoft/Msdyn365.Commerce.Online

There are two ways to get the SDK configuration packages to your development machine, either directly download the zip file from the GitHub repo or clone the repo.

DOWNLOAD THE SDK ZIP FILE

From the above Github site, press the “Clone or download” button and select “Download ZIP”. Save the file and unzip it to a directory such as “c:\repos”.



Dynamics 365 Commerce online project

mkelan Update package.json

.vscode	Initial commit
public	Adding webfonts and robots to public
src	Update Settings for Fabrikam
starter-pack	Update SSK License

Use Git or checkout with SVN using the web URL.

https://github.com/microsoft/Msdyn365.Co

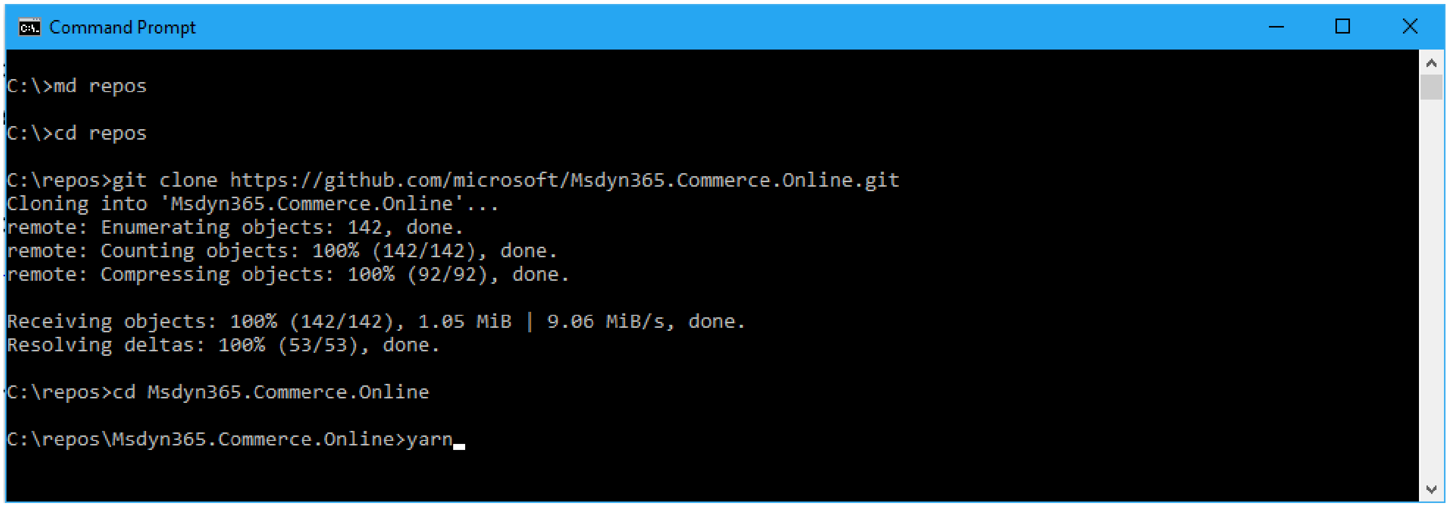
2 months ago

CLONE THE SDK

To clone, launch a Command Prompt with Administrator privileges and **create a directory** to hold your e-Commerce site code, we’ll use **“c:\repos”** below. You will need the Git tools installed for this method.

From within the new directory type in the command “**git clone SDK_GIT_REPO**”. Example:

```
C:>md repos
C:>cd repos
C:\repos>git clone https://github.com/microsoft/Msdyn365.Commerce.Online.giyarn t
C:\repos>cd Msdyn365.Commerce.Online
```



```
C:\>md repos

C:\>cd repos

C:\repos>git clone https://github.com/microsoft/Msdyn365.Commerce.Online.git
Cloning into 'Msdyn365.Commerce.Online'...
remote: Enumerating objects: 142, done.
remote: Counting objects: 100% (142/142), done.
remote: Compressing objects: 100% (92/92), done.

Receiving objects: 100% (142/142), 1.05 MiB | 9.06 MiB/s, done.
Resolving deltas: 100% (53/53), done.

C:\repos>cd Msdyn365.Commerce.Online

C:\repos\Msdyn365.Commerce.Online>yarn
```

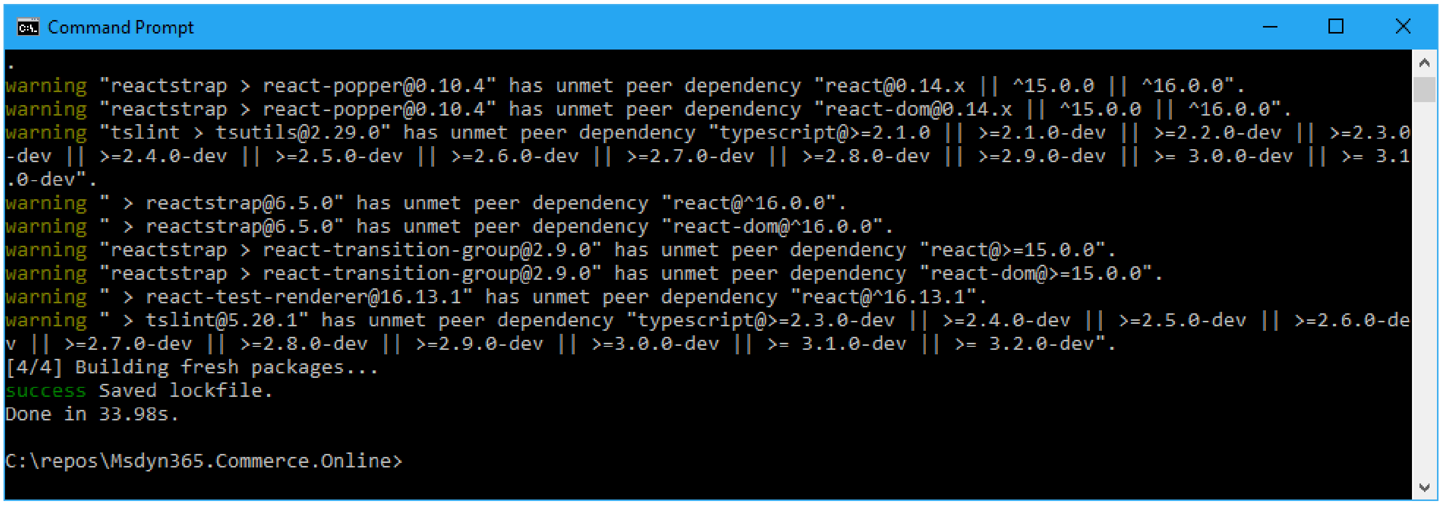
We now have the SDK ‘configuration’ files but have not fully downloaded the SDK and all the dependency packages which include the store starter kit, the Fabrikam theme, Retail proxy files, etc …

We will do this by running the “yarn” command which leverage the Node package manager (NPM).

From the new directory (example: “c:\repos\Msdyn365.Commerce.Online”) directory type “**yarn**” followed by the Enter key. This step can take a couple of minutes to complete.

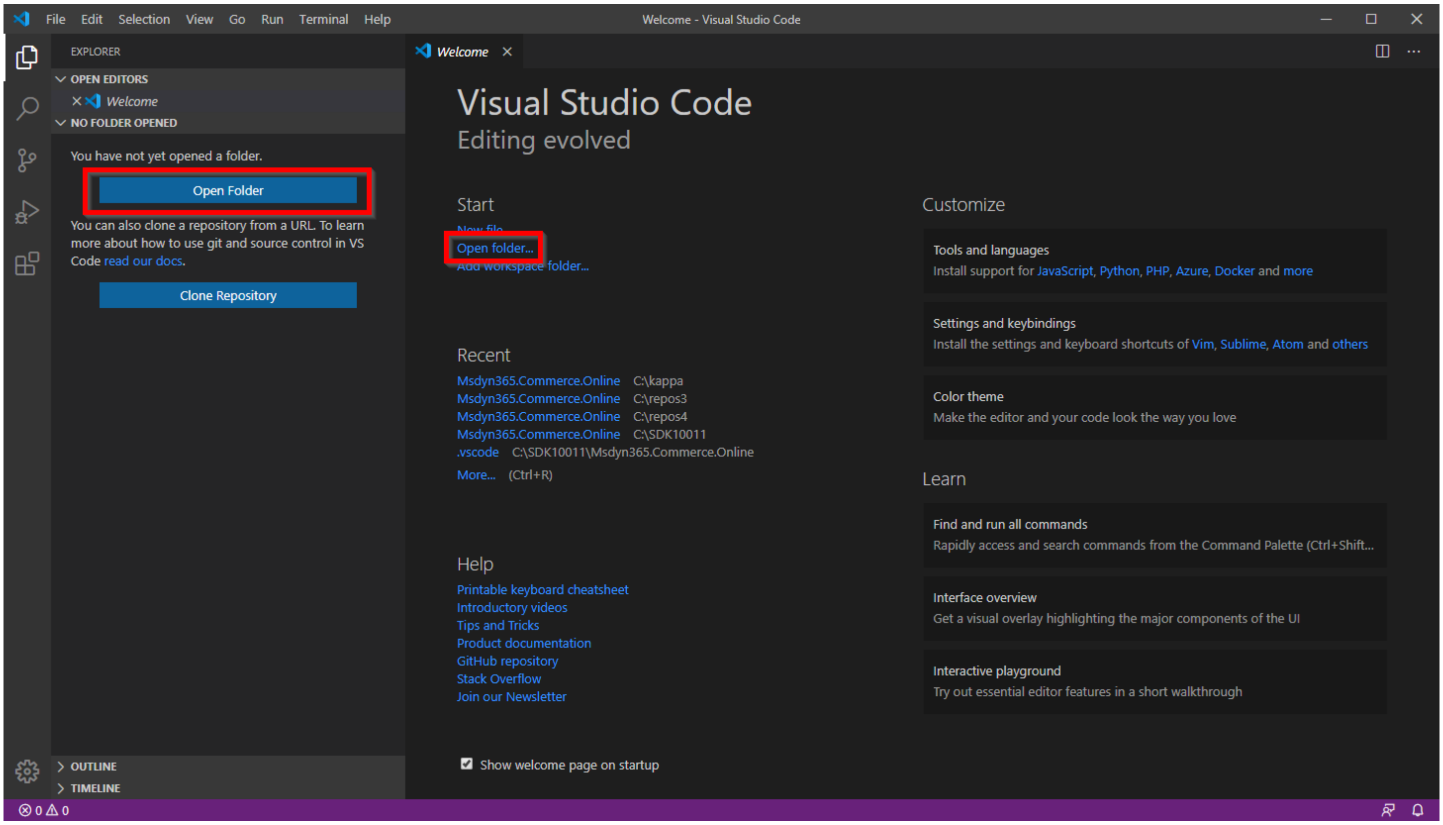
(Note: if you get “**would clobber existing tag**” warnings, run “**yarn cache clean**”)

When complete, the results should look like the below screen shot. Note: warnings in the output are ok, no need to worry about these.



```
.
warning "reactstrap > react-popper@0.10.4" has unmet peer dependency "react@0.14.x || ^15.0.0 || ^16.0.0".
warning "reactstrap > react-popper@0.10.4" has unmet peer dependency "react-dom@0.14.x || ^15.0.0 || ^16.0.0".
warning "tslint > tsutils@2.29.0" has unmet peer dependency "typescript@>=2.1.0 || >=2.1.0-dev || >=2.2.0-dev || >=2.3.0
-dev || >=2.4.0-dev || >=2.5.0-dev || >=2.6.0-dev || >=2.7.0-dev || >=2.8.0-dev || >=2.9.0-dev || >= 3.0.0-dev || >= 3.1
.0-dev".
warning " > reactstrap@6.5.0" has unmet peer dependency "react@^16.0.0".
warning " > reactstrap@6.5.0" has unmet peer dependency "react-dom@^16.0.0".
warning "reactstrap > react-transition-group@2.9.0" has unmet peer dependency "react@>=15.0.0".
warning "reactstrap > react-transition-group@2.9.0" has unmet peer dependency "react-dom@>=15.0.0".
warning " > react-test-renderer@16.13.1" has unmet peer dependency "react@^16.13.1".
warning " > tslint@5.20.1" has unmet peer dependency "typescript@>=2.3.0-dev || >=2.4.0-dev || >=2.5.0-dev || >=2.6.0-de
v || >=2.7.0-dev || >=2.8.0-dev || >=2.9.0-dev || >=3.0.0-dev || >= 3.1.0-dev || >= 3.2.0-dev".
[4/4] Building fresh packages...
success Saved lockfile.
Done in 33.98s.

C:\repos\Msdyn365.Commerce.Online>
```

Now that we have our SDK/starter kit source code, launch Visual Studio Code and open the new root repo folder (example: “c:\repos\Msdyn365.Commerce.Online”).



You have not yet opened a folder.

You can also clone a repository from a URL. To learn more about how to use git and source control in VS Code read our docs.

Install support for JavaScript, Python, PHP, Azure, Docker and more

Install the settings and keyboard shortcuts of Vim, Sublime, Atom and others

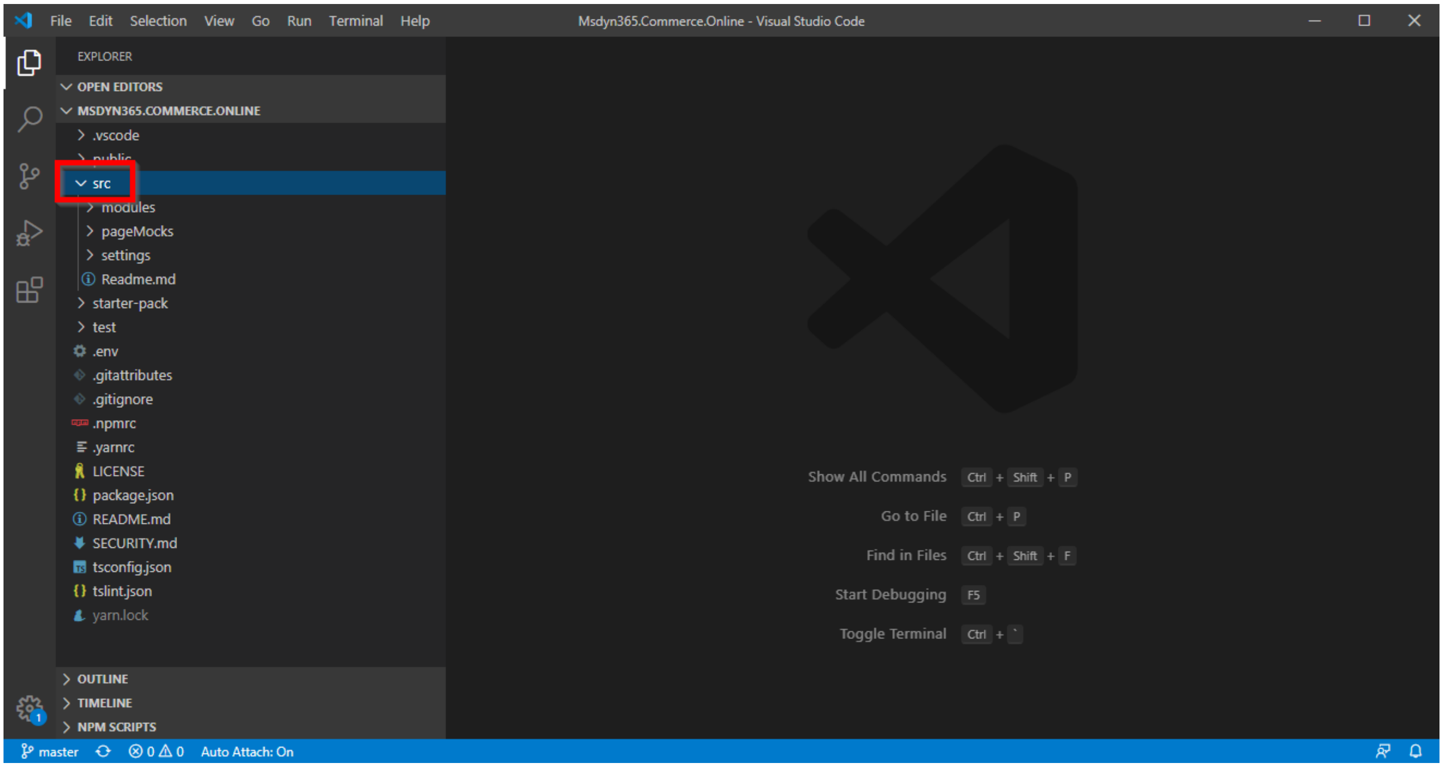
Make the editor and your code look the way you love

Rapidly access and search commands from the Command Palette (Ctrl+Shift...

Get a visual overlay highlighting the major components of the UI

Try out essential editor features in a short walkthrough

Take note of the “**/src**” directory, this is where all your customization code will live.

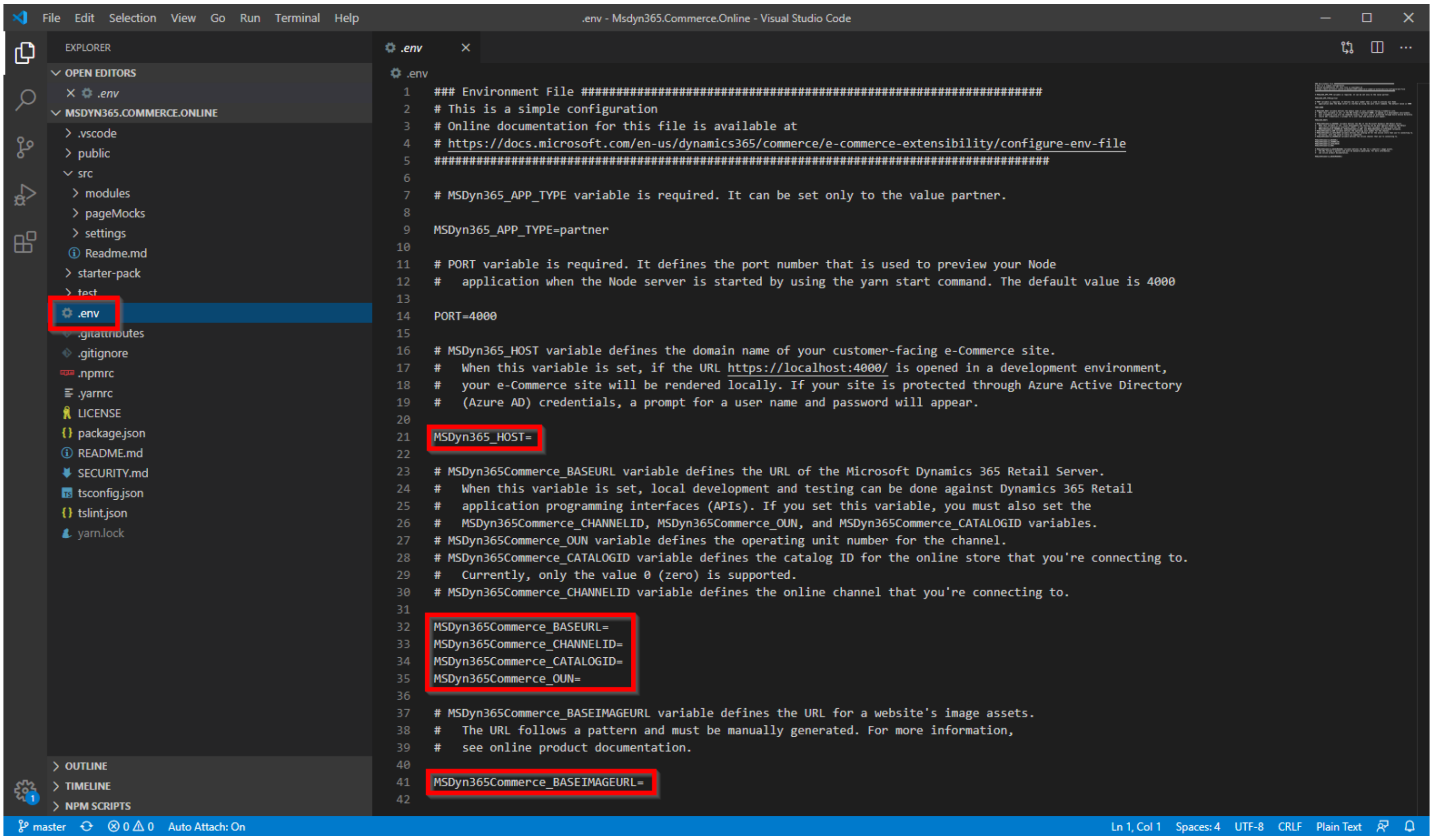

File Edit Selection View Go Run Terminal Help
Msdyn365.Commerce.Online - Visual Studio Code
EXPLORER
OPEN EDITORS
MSDYN365.COMMERCE.ONLINE
.vscode
public
src
modules
pageMocks
settings
Readme.md
starter-pack
test
.env
.gitattributes
.gitignore
.npmrc
.yarnrc
LICENSE
package.json
README.md
SECURITY.md
tsconfig.json
tslint.json
yarn.lock
OUTLINE
TIMELINE
NPM SCRIPTS
Show All Commands Ctrl + Shift + P
Go to File Ctrl + P
Find in Files Ctrl + Shift + F
Start Debugging F5
Toggle Terminal Ctrl + `
master
⊗ 0 ⚠ 0
Auto Attach: On

CONFIGURING DEVELOPMENT .ENV FILE

Follow the docs to configure the .env file to point to your environment. This step is optional for a development environment but will help us when developing themes to test against our live site as well as to get live data from the Commerce server when developing with data actions.

If you are do not currently have a Dynamics 365 Retail server and/or Online Channel, you can still do local module development.

Open the .env file in VS Code and modify the highlighted line to point to your environment:



```
### Environment File ##################################################################
# This is a simple configuration
# Online documentation for this file is available at
# https://docs.microsoft.com/en-us/dynamics365/commerce/e-commerce-extensibility/configure-env-file
########################################################################################

# MSDyn365_APP_TYPE variable is required. It can be set only to the value partner.

MSDyn365_APP_TYPE=partner

# PORT variable is required. It defines the port number that is used to preview your Node
#   application when the Node server is started by using the yarn start command. The default value is 4000

PORT=4000

# MSDyn365_HOST variable defines the domain name of your customer-facing e-Commerce site.
#   When this variable is set, if the URL https://localhost:4000/ is opened in a development environment,
#   your e-Commerce site will be rendered locally. If your site is protected through Azure Active Directory
#   (Azure AD) credentials, a prompt for a user name and password will appear.

MSDyn365_HOST=

# MSDyn365Commerce_BASEURL variable defines the URL of the Microsoft Dynamics 365 Retail Server.
#   When this variable is set, local development and testing can be done against Dynamics 365 Retail
#   application programming interfaces (APIs). If you set this variable, you must also set the
#   MSDyn365Commerce_CHANNELID, MSDyn365Commerce_OUN, and MSDyn365Commerce_CATALOGID variables.
# MSDyn365Commerce_OUN variable defines the operating unit number for the channel.
# MSDyn365Commerce_CATALOGID variable defines the catalog ID for the online store that you're connecting to.
#   Currently, only the value 0 (zero) is supported.
# MSDyn365Commerce_CHANNELID variable defines the online channel that you're connecting to.

MSDyn365Commerce_BASEURL=
MSDyn365Commerce_CHANNELID=
MSDyn365Commerce_CATALOGID=
MSDyn365Commerce_OUN=

# MSDyn365Commerce_BASEIMAGEURL variable defines the URL for a website's image assets.
#   The URL follows a pattern and must be manually generated. For more information,
#   see online product documentation.

MSDyn365Commerce_BASEIMAGEURL=

```

The below is a sample .env file (note that the comments have been removed), save when complete.

```
### Environment File ##################################################################
# This is a simple configuration
# Online documentation for this file is available at
# https://docs.microsoft.com/en-us/dynamics365/commerce/e-commerce-extensibility/configure-env-file
########################################################################################

# MSDyn365_APP_TYPE variable is required. It can be set only to the value partner.

MSDyn365_APP_TYPE=partner

# PORT variable is required. It defines the port number that is used to preview your Node
#   application when the Node server is started by using the yarn start command. The default value
is 4000

PORT=4000

# MSDyn365_HOST variable defines the domain name of your customer-facing e-Commerce site.
#   When this variable is set, if the URL https://localhost:4000/ is opened in a development
environment,
#   your e-Commerce site will be rendered locally. If your site is protected through Azure Active
Directory
#   (Azure AD) credentials, a prompt for a user name and password will appear.

MSDyn365_HOST=commerce-in-day-eval.commerce.dynamics.com

# MSDyn365Commerce_BASEURL variable defines the URL of the Microsoft Dynamics 365 Retail Server.
#   When this variable is set, local development and testing can be done against Dynamics 365
Retail
#   application programming interfaces (APIs). If you set this variable, you must also set the
#   MSDyn365Commerce_CHANNELID, MSDyn365Commerce_OUN, and MSDyn365Commerce_CATALOGID variables.
# MSDyn365Commerce_OUN variable defines the operating unit number for the channel.
# MSDyn365Commerce_CATALOGID variable defines the catalog ID for the online store that you're
connecting to.
#   Currently, only the value 0 (zero) is supported.
# MSDyn365Commerce_CHANNELID variable defines the online channel that you're connecting to.

MSDyn365Commerce_BASEURL=https://commerce-in-day11b59da3b157638bret.cloud.retail.dynamics.com/
MSDyn365Commerce_CHANNELID=68719478279
MSDyn365Commerce_CATALOGID=0
MSDyn365Commerce_OUN=128

# MSDyn365Commerce_BASEIMAGEURL variable defines the URL for a website's image assets.
#   The URL follows a pattern and must be manually generated. For more information,
#   see online product documentation.

MSDyn365Commerce_BASEIMAGEURL=https://images-us-
sb.cms.commerce.dynamics.com/cms/api/rnbrmdbgjb/imageFileData/search?fileName=/
```

RUN NODE SERVER

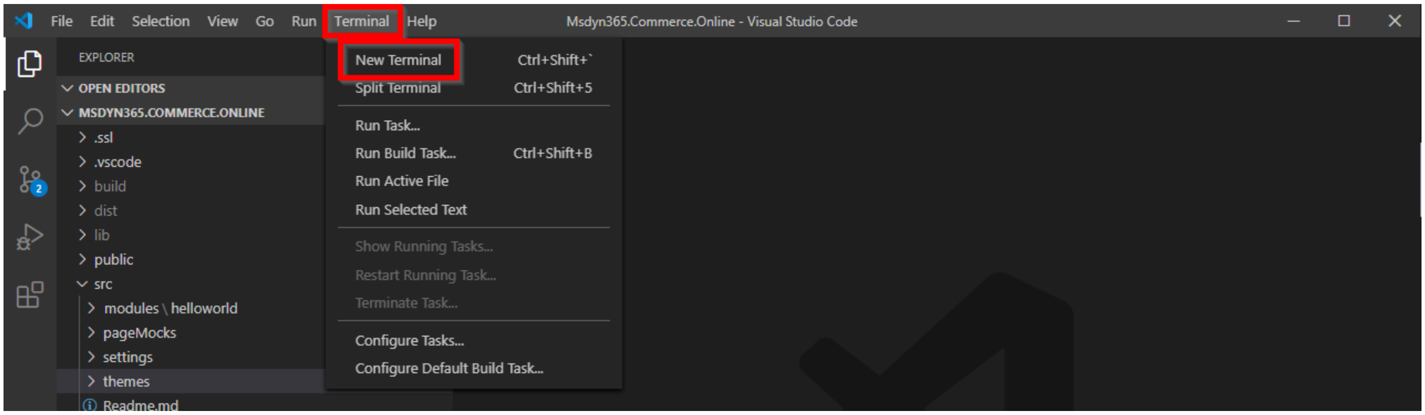
To check if everything worked, we will start the Node server and load some pages on your machine.

We use the Node.js JavaScript runtime to render modules and our e-Commerce rending engine uses it to render full pages in the production environment. To see modules in the development environment you need to run the node server with the following steps.

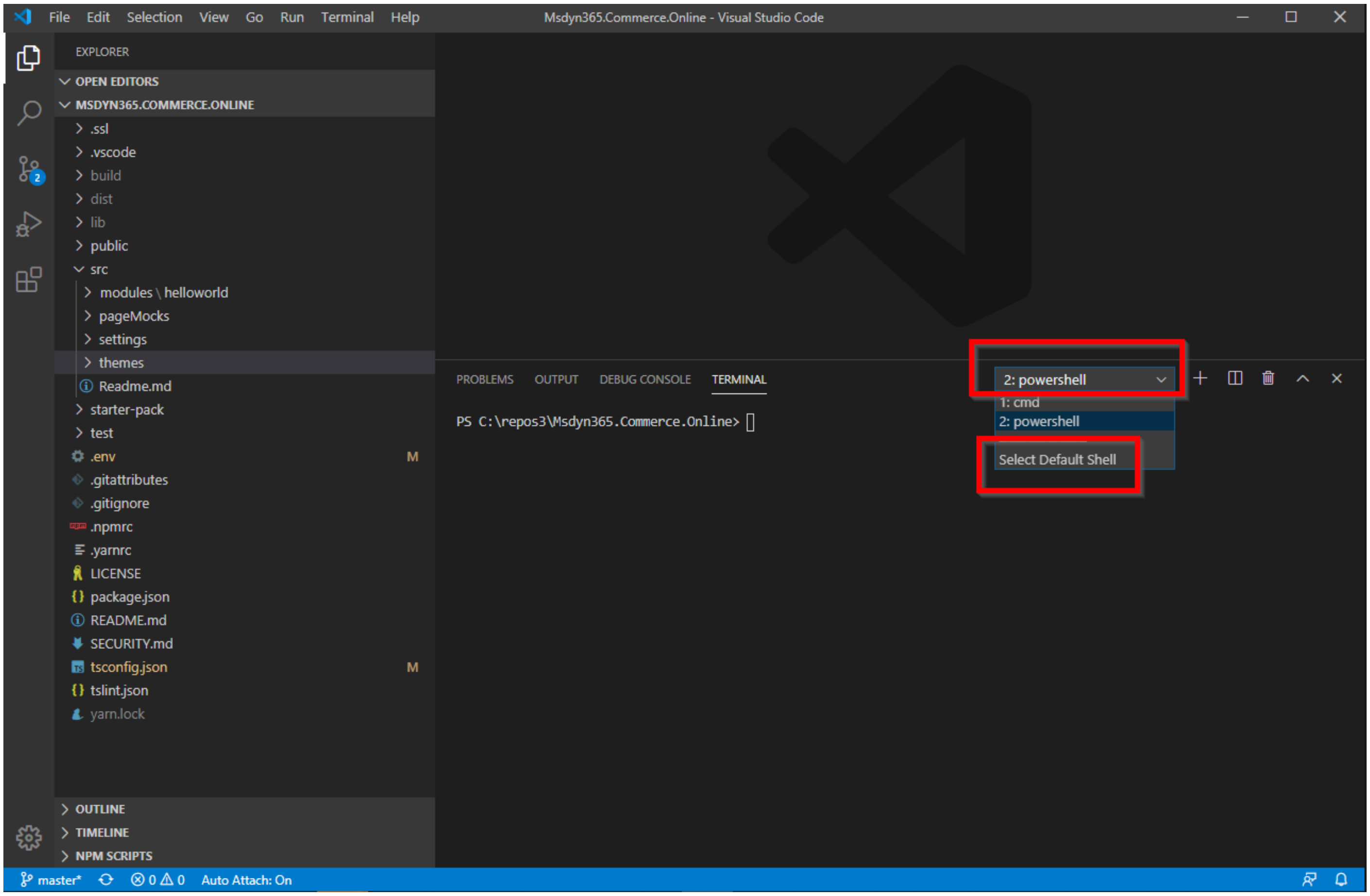
Run "**yarn start**" in the command prompt which will perform a build followed by starting up the Node server.

Below instructions is a great way to get a command prompt right inside Visual Studio Code for this.

From within Visual Studio Code, open a command prompt terminal window.

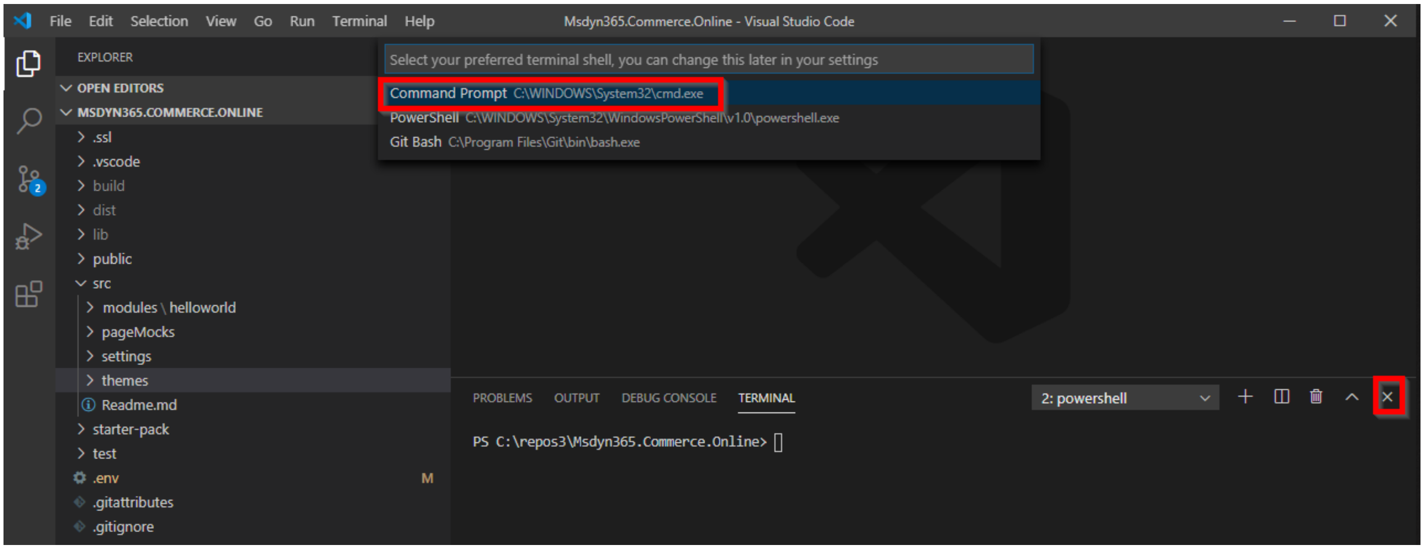


You should see a new window open up similar to below, by default it will open a powershell window, however we want a command prompt window. Select the "**Select default shell**" link as shown below.



```
PS C:\repos3\Msdyn365.Commerce.Online> 
```

Next you can select the Command Prompt as our default.

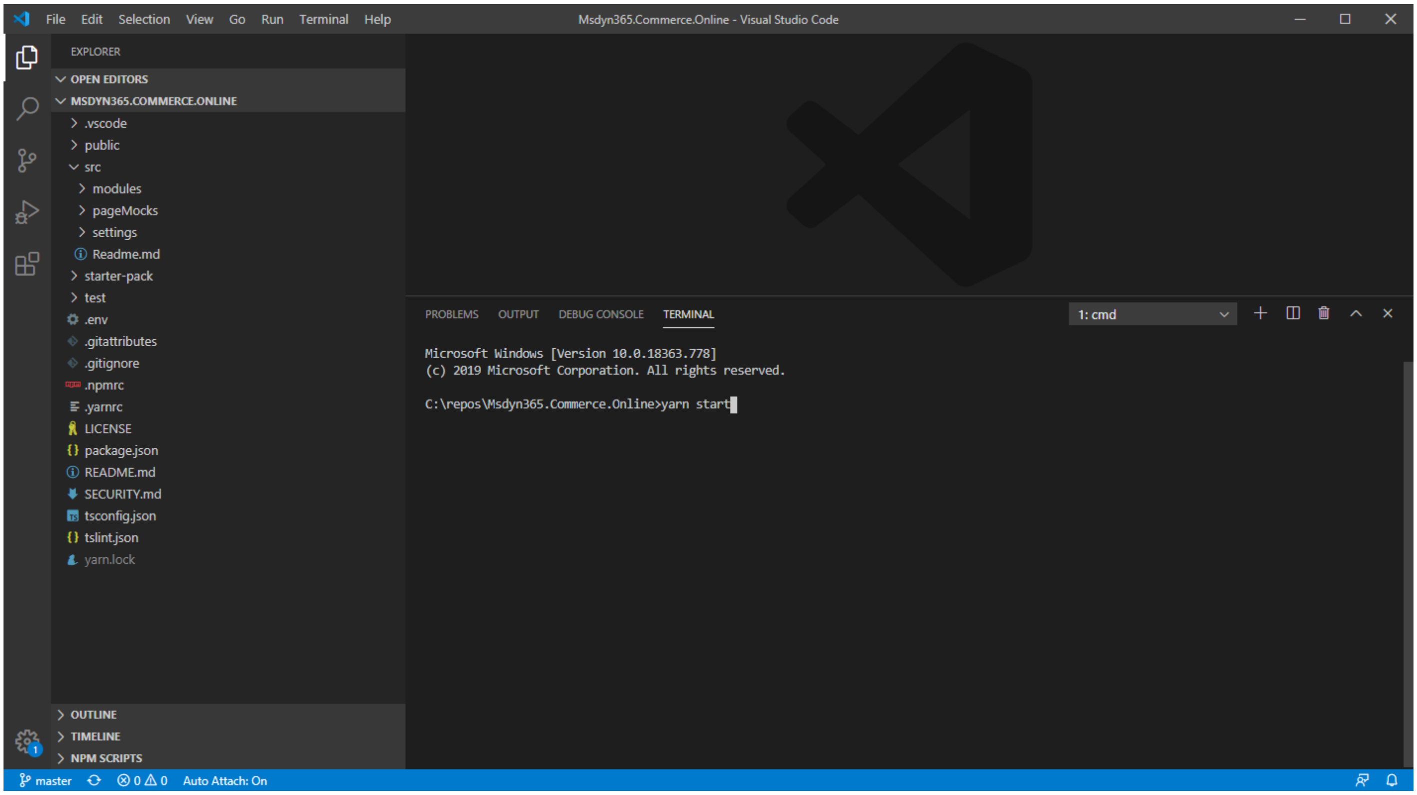


Select your preferred terminal shell, you can change this later in your settings

- Command Prompt C:\WINDOWS\System32\cmd.exe
- PowerShell C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe
- Git Bash C:\Program Files\Git\bin\bash.exe

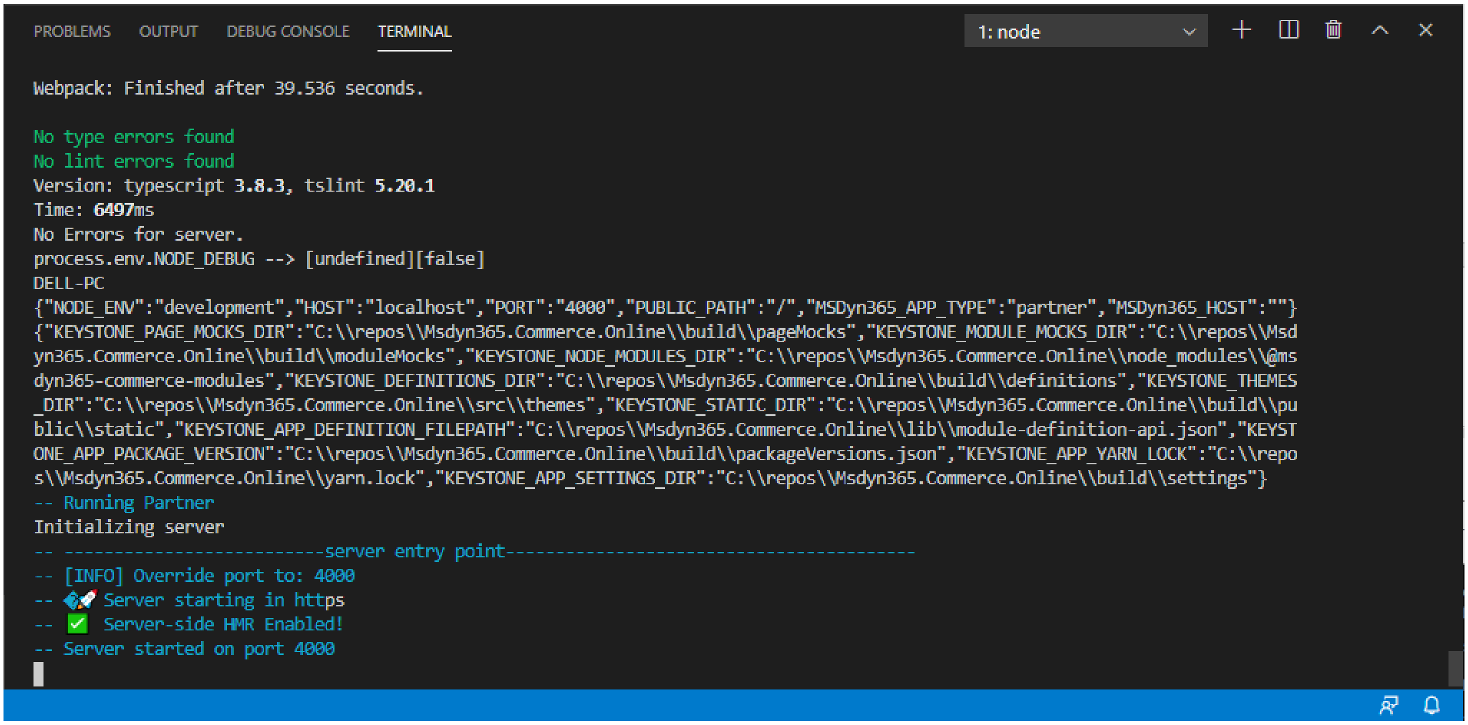
```
PS C:\repos3\Msdyn365.Commerce.Online>
```

We can now hit the “**x**” to close the powershell window and select “Terminal” -> “New terminal” again, and you should see the “cmd” listed in the terminal window as shown below. Notice how it opens right up into our SDK directory ready for our commands.



```
Microsoft Windows [Version 10.0.18363.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\repos\Msdyn365.Commerce.Online>yarn start
```

Type in “yarn start” followed by the Enter key to start the compiling and node server. This may take a couple of minutes the first time it is run as it compiles all of the code down to Javascript and css. You may also see a Windows Defender Firewall warning when Node server starts up, you should accept the port.

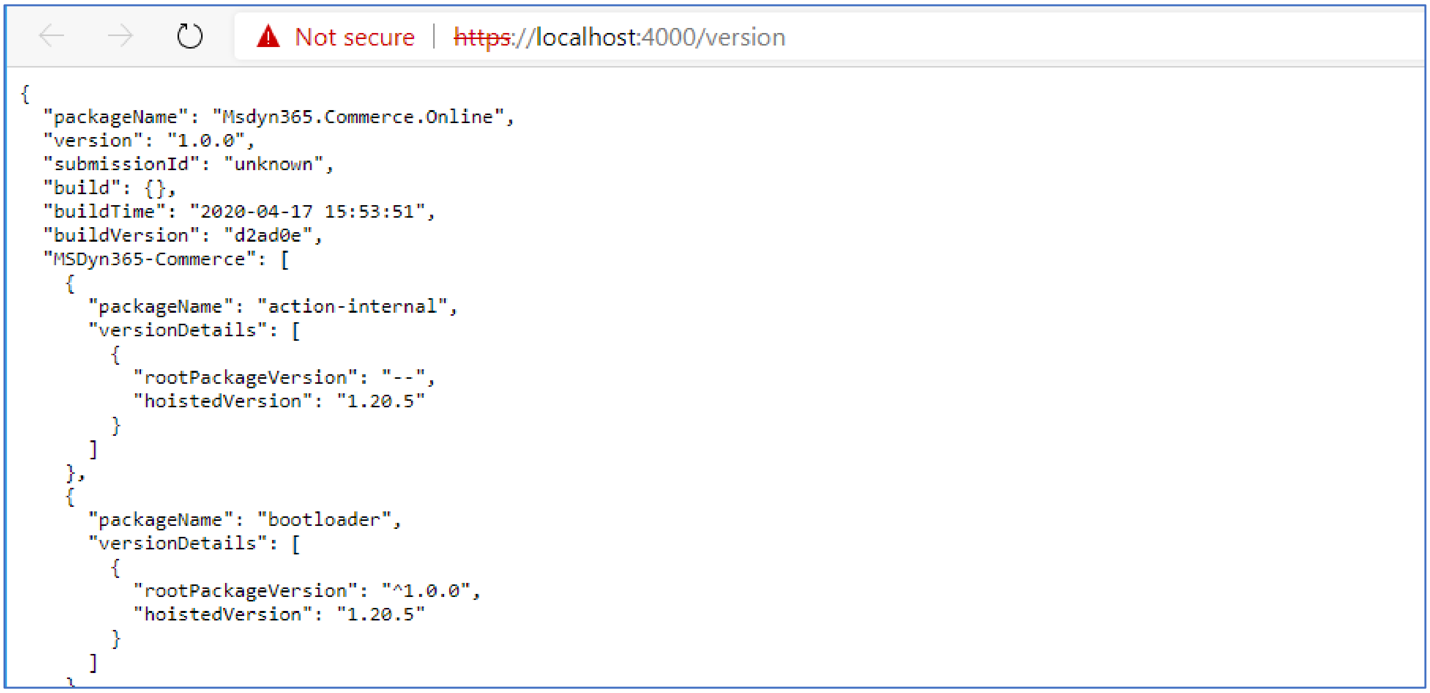


```
Webpack: Finished after 39.536 seconds.

No type errors found
No lint errors found
Version: typescript 3.8.3, tslint 5.20.1
Time: 6497ms
No Errors for server.
process.env.NODE_DEBUG --> [undefined][false]
DELL-PC
{"NODE_ENV":"development","HOST":"localhost","PORT":"4000","PUBLIC_PATH":"/","MSDyn365_APP_TYPE":"partner","MSDyn365_HOST":""}
{"KEYSTONE_PAGE_MOCKS_DIR":"C:\\repos\\Msdyn365.Commerce.Online\\build\\pageMocks","KEYSTONE_MODULE_MOCKS_DIR":"C:\\repos\\Msd
yn365.Commerce.Online\\build\\moduleMocks","KEYSTONE_NODE_MODULES_DIR":"C:\\repos\\Msdyn365.Commerce.Online\\node_modules\\@ms
dyn365-commerce-modules","KEYSTONE_DEFINITIONS_DIR":"C:\\repos\\Msdyn365.Commerce.Online\\build\\definitions","KEYSTONE_THEMES
_DIR":"C:\\repos\\Msdyn365.Commerce.Online\\src\\themes","KEYSTONE_STATIC_DIR":"C:\\repos\\Msdyn365.Commerce.Online\\build\\pu
blic\\static","KEYSTONE_APP_DEFINITION_FILEPATH":"C:\\repos\\Msdyn365.Commerce.Online\\lib\\module-definition-api.json","KEYST
ONE_APP_PACKAGE_VERSION":"C:\\repos\\Msdyn365.Commerce.Online\\build\\packageVersions.json","KEYSTONE_APP_YARN_LOCK":"C:\\repo
s\\Msdyn365.Commerce.Online\\yarn.lock","KEYSTONE_APP_SETTINGS_DIR":"C:\\repos\\Msdyn365.Commerce.Online\\build\\settings"}
-- Running Partner
Initializing server
-- --------------------------server entry point-----------------------------------------
-- [INFO] Override port to: 4000
-- 🚀 Server starting in https
-- ✅ Server-side HMR Enabled!
-- Server started on port 4000
```

The node server is configured to use port 4000 by default (see .env file we configured earlier to change this).

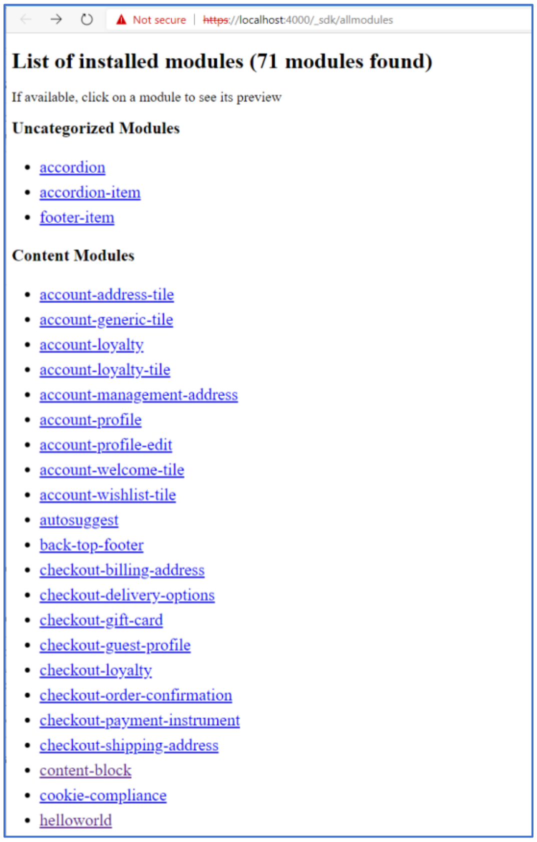
Launch https://localhost:4000/version

Note you may be prompted that the site is unsafe since we’re using https and do not have a local certificate installed, in this case click the “Advanced” button followed by the “Proceed to localhost (unsafe)” link. You should see a similar page as below which shows the current versions of the installed packages.



```
{
  "packageName": "Msdyn365.Commerce.Online",
  "version": "1.0.0",
  "submissionId": "unknown",
  "build": {},
  "buildTime": "2020-04-17 15:53:51",
  "buildVersion": "d2ad0e",
  "MSDyn365-Commerce": [
    {
      "packageName": "action-internal",
      "versionDetails": [
        {
          "rootPackageVersion": "--",
          "hoistedVersion": "1.20.5"
        }
      ]
    },
    {
      "packageName": "bootloader",
      "versionDetails": [
        {
          "rootPackageVersion": "^1.0.0",
          "hoistedVersion": "1.20.5"
        }
      ]
```

The module list page can be shown with the following link: https://localhost:4000/_sdk/allmodules. You should see a list of starter kit modules and custom modules listed as shown below.



List of installed modules (71 modules found)

If available, click on a module to see its preview

Uncategorized Modules

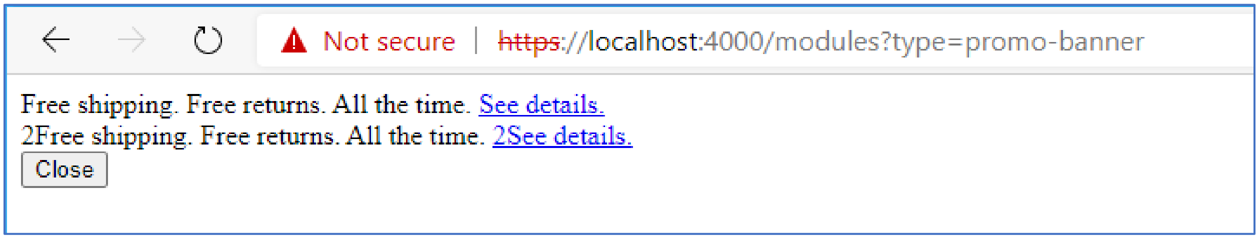
- accordion
- accordion-item
- footer-item

Content Modules

- account-address-tile
- account-generic-tile
- account-loyalty
- account-loyalty-tile
- account-management-address
- account-profile
- account-profile-edit
- account-welcome-tile
- account-wishlist-tile
- autosuggest
- back-top-footer
- checkout-billing-address
- checkout-delivery-options
- checkout-gift-card
- checkout-guest-profile
- checkout-loyalty
- checkout-order-confirmation
- checkout-payment-instrument
- checkout-shipping-address
- content-block
- cookie-compliance
- helloworld

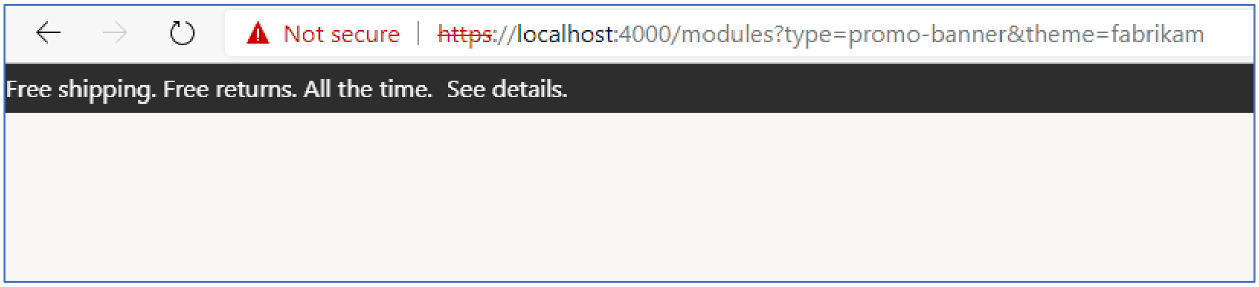
You can then select a starter kit module such as the "**promo-banner**" module and notice the URL has the module name in it, which can be changed to show any module: https://localhost:4000/modules?type=promo-banner

Notice how the module mock data is not rendering very well (note some modules may show up as blank because mock data is not setup for them):



Free shipping. Free returns. All the time. See details.
2Free shipping. Free returns. All the time. 2See details.

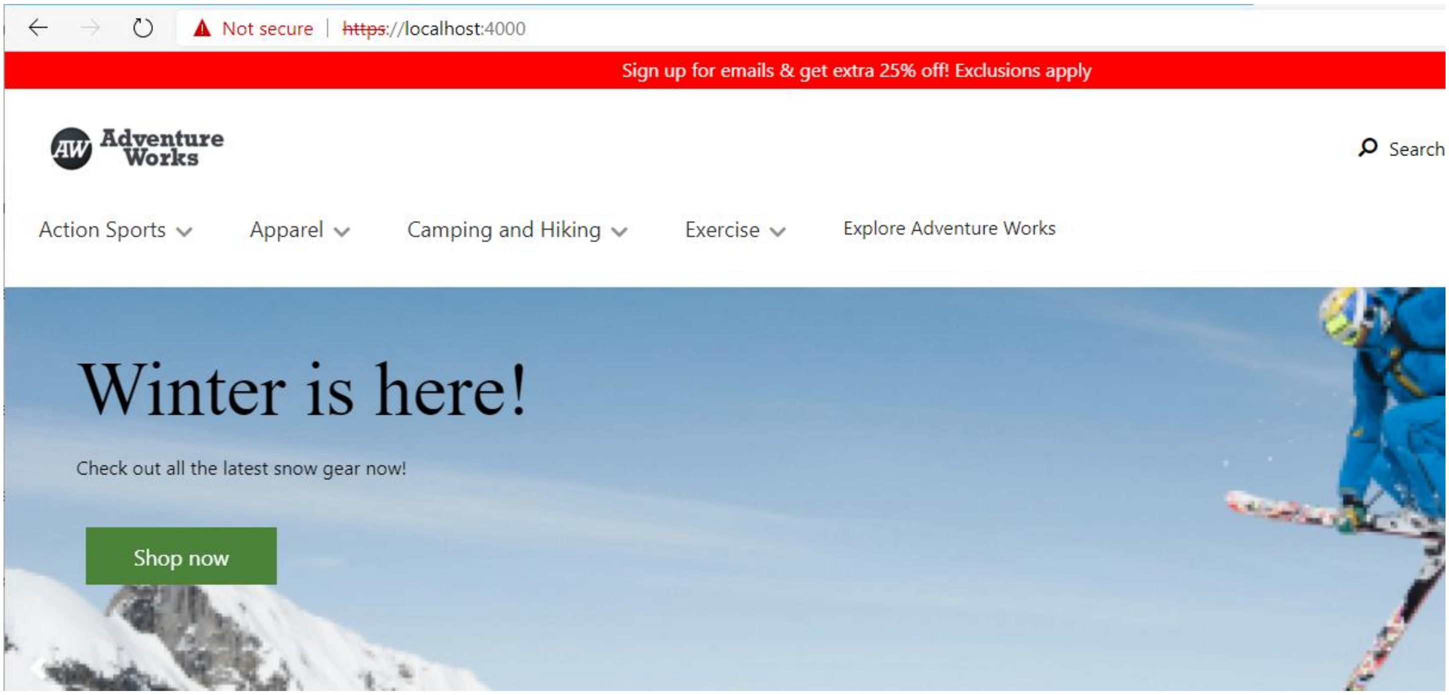
This is because no CSS has been applied to the module. This is easy to fix, we can apply the "Fabrikam" starter kit theme by changing the URL to https://localhost:4000/modules?type=promo-banner&theme=fabrikam. Later we can do the same with any theme we create.


Not secure
~~https~~://localhost:4000/modules?type=promo-banner&theme=fabrikam

Free shipping. Free returns. All the time. See details.

One last test is to change the URL to https://www.localhost.com:4000/

With this URL, if you configured the "MSDyn365_HOST" in the .env file to point to your online web site, you will see your web site, but take note that the rendering of the page is done on the local Node development environment, this is handle when making local changes as see how they would look on your live site.



Now click CTRL-C in the Command Window to stop the Node server. You can keep this window open we will be running the Node server again soon.

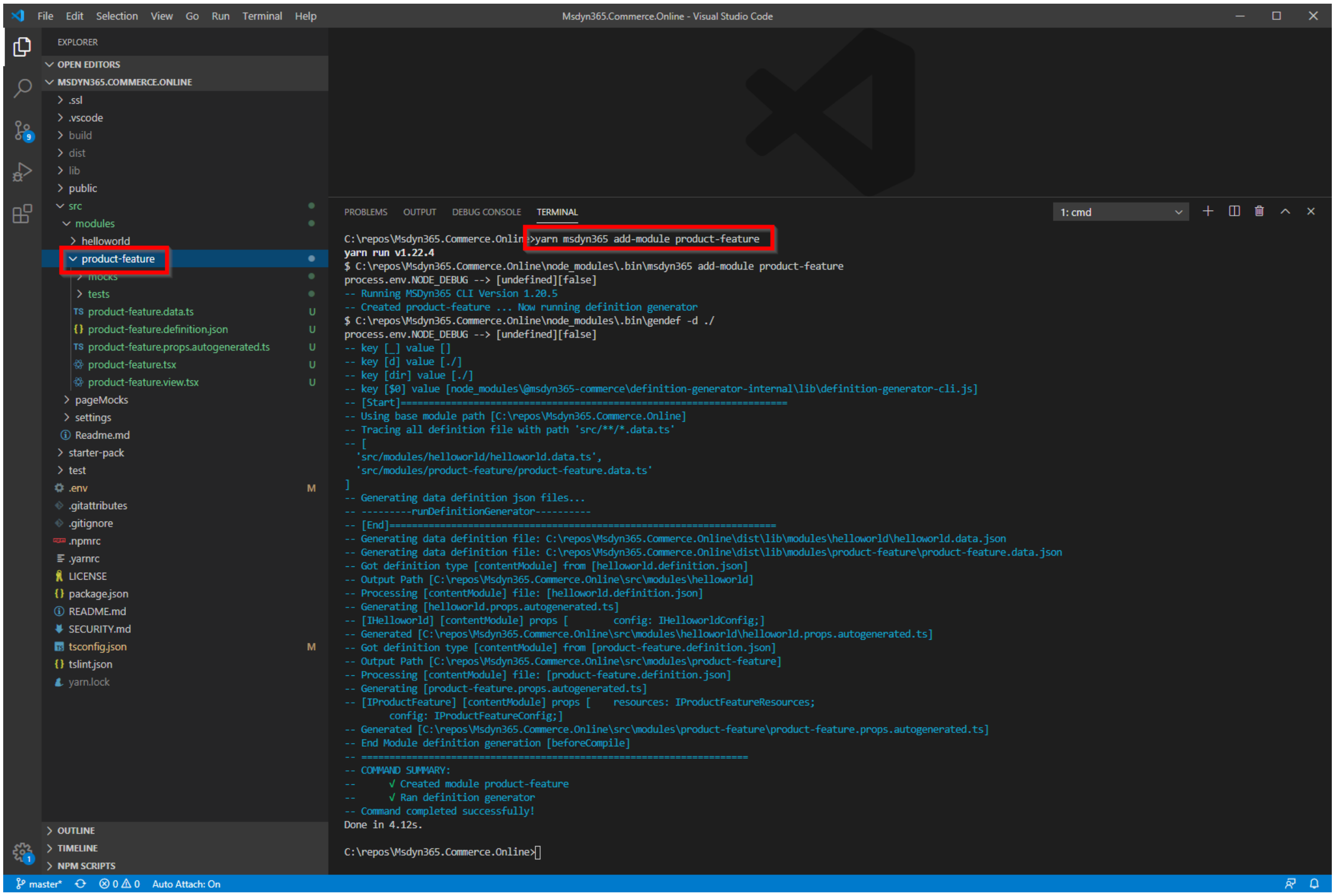
CREATING A NEW MODULE

We are now ready to create our first module.

The online SDK includes several Command Line Interface (CLI) commands to complete specific tasks. All CLI commands are pre-pended with "**yarn msdyn365**" followed by the command name and any arguments.

From the Command Prompt we were previously using in Visual Studio Code type in the below command to create a module titled "product-feature" (feel free to change this name).

```
yarn msdyn365 add-module product-feature
```



```
C:\repos\Msdyn365.Commerce.Online>yarn msdyn365 add-module product-feature
yarn run v1.22.4
$ C:\repos\Msdyn365.Commerce.Online\node_modules\.bin\msdyn365 add-module product-feature
process.env.NODE_DEBUG --> [undefined][false]
-- Running MSDyn365 CLI Version 1.20.5
-- Created product-feature ... Now running definition generator
$ C:\repos\Msdyn365.Commerce.Online\node_modules\.bin\gendef -d ./
process.env.NODE_DEBUG --> [undefined][false]
-- key [_] value []
-- key [d] value [./]
-- key [dir] value [./]
-- key [$0] value [node_modules\@msdyn365-commerce\definition-generator-internal\lib\definition-generator-cli.js]
-- [Start]==================================================================
-- Using base module path [C:\repos\Msdyn365.Commerce.Online]
-- Tracing all definition file with path 'src/**/*.data.ts'
-- [
  'src/modules/helloworld/helloworld.data.ts',
  'src/modules/product-feature/product-feature.data.ts'
]
-- Generating data definition json files...
-- ---------runDefinitionGenerator----------
-- [End]====================================================================
-- Generating data definition file: C:\repos\Msdyn365.Commerce.Online\dist\lib\modules\helloworld\helloworld.data.json
-- Generating data definition file: C:\repos\Msdyn365.Commerce.Online\dist\lib\modules\product-feature\product-feature.data.json
-- Got definition type [contentModule] from [helloworld.definition.json]
-- Output Path [C:\repos\Msdyn365.Commerce.Online\src\modules\helloworld]
-- Processing [contentModule] file: [helloworld.definition.json]
-- Generating [helloworld.props.autogenerated.ts]
-- [IHelloworld] [contentModule] props [        config: IHelloworldConfig;]
-- Generated [C:\repos\Msdyn365.Commerce.Online\src\modules\helloworld\helloworld.props.autogenerated.ts]
-- Got definition type [contentModule] from [product-feature.definition.json]
-- Output Path [C:\repos\Msdyn365.Commerce.Online\src\modules\product-feature]
-- Processing [contentModule] file: [product-feature.definition.json]
-- Generating [product-feature.props.autogenerated.ts]
-- [IProductFeature] [contentModule] props [    resources: IProductFeatureResources;
        config: IProductFeatureConfig;]
-- Generated [C:\repos\Msdyn365.Commerce.Online\src\modules\product-feature\product-feature.props.autogenerated.ts]
-- End Module definition generation [beforeCompile]
-- ==================================================================
-- COMMAND SUMMARY:
--       √ Created module product-feature
--       √ Ran definition generator
-- Command completed successfully!
Done in 4.12s.

C:\repos\Msdyn365.Commerce.Online>
```

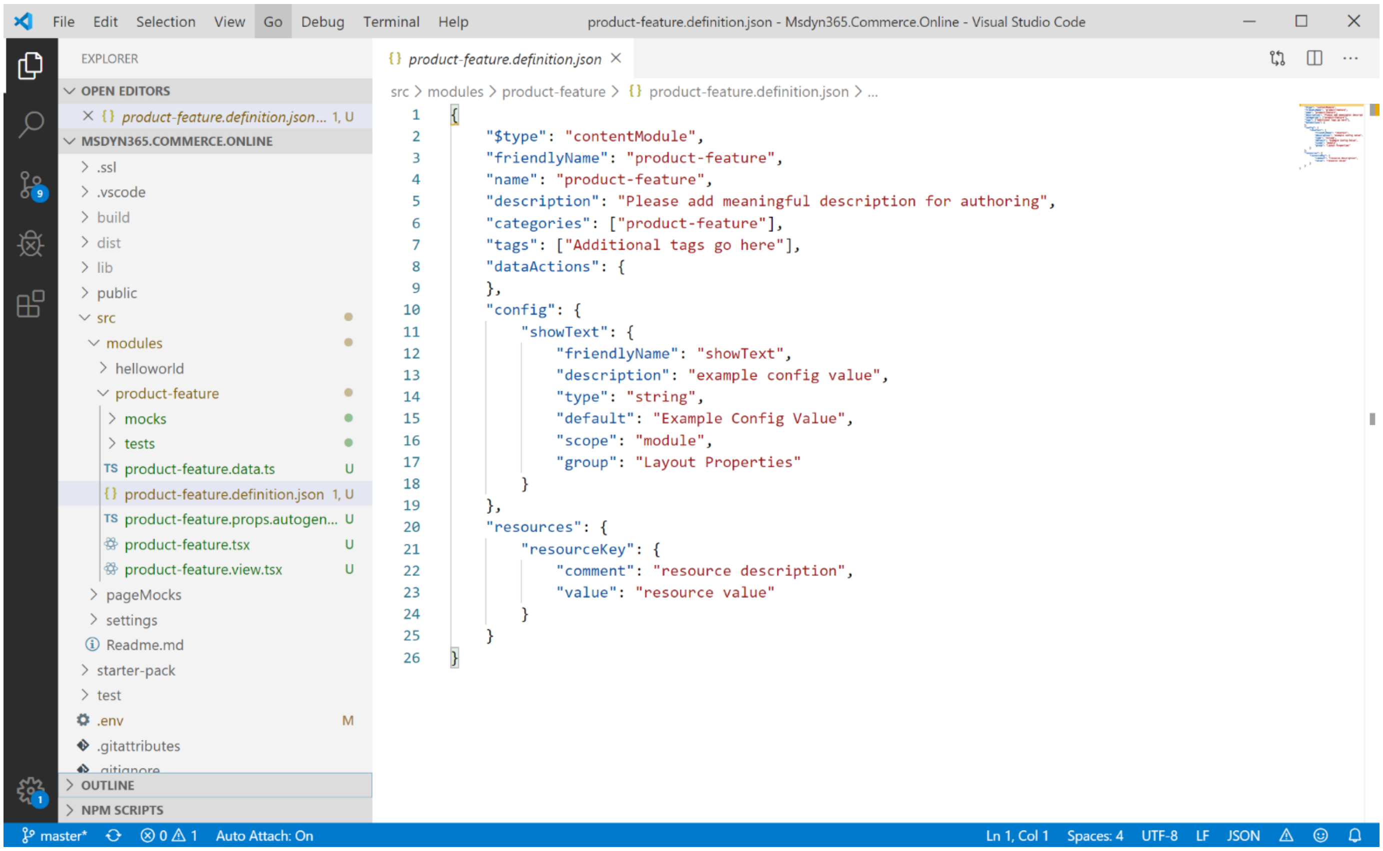
You have now successfully created a new module! Notice how it showed up under the "/src/modules/" directory. We'll next look at the template files that were created.

ANATOMY OF A MODULE

From within Visual Studio Code let us briefly examine the new module directory that was created “/src/modules/product-feature” to understand the anatomy of a module.

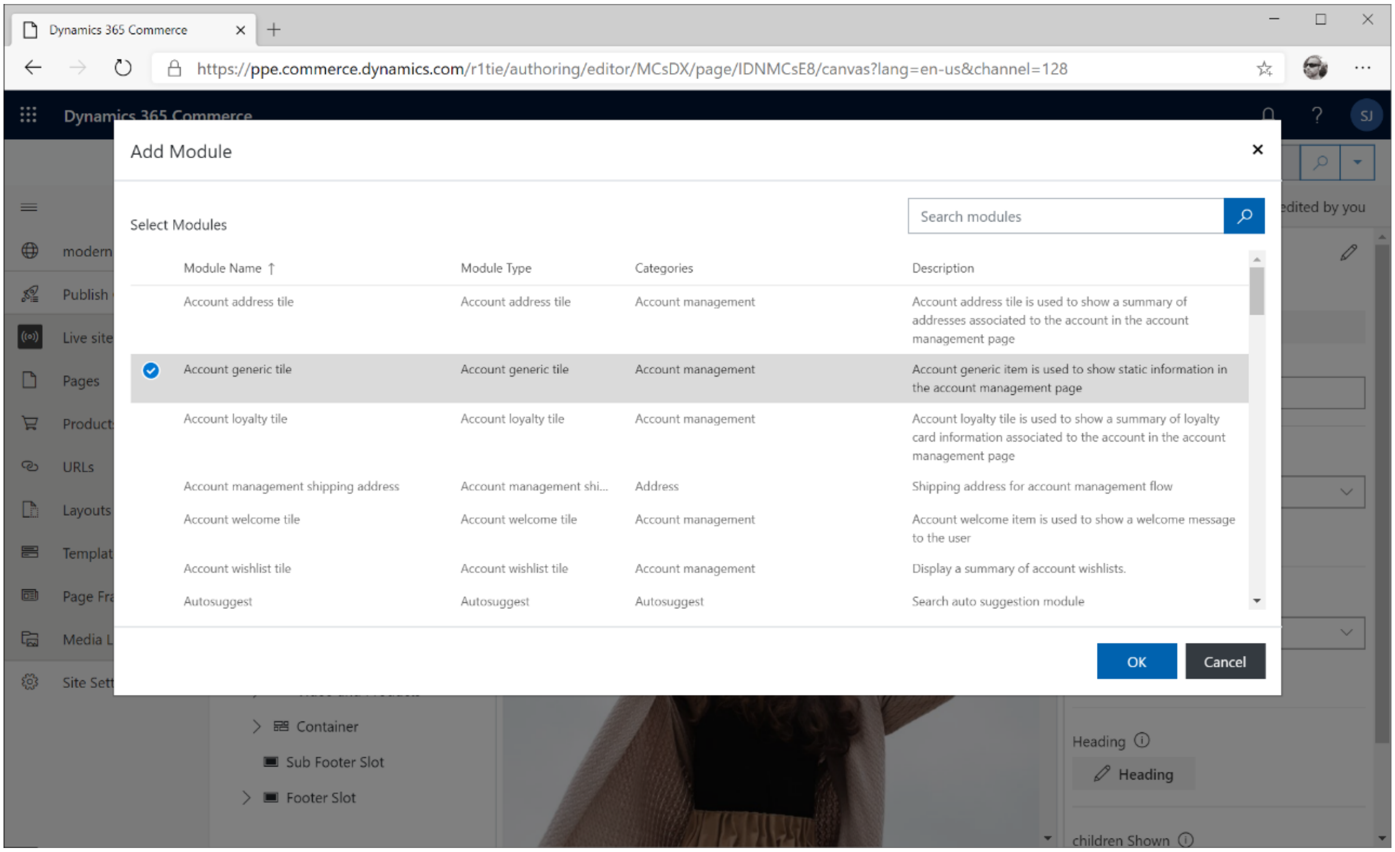
MODULE_NAME.DEFINITION.JSON

This file is referred to as the module definition file which is used to register the module and provides metadata to the site builder tool including the module name, description and category.


File Edit Selection View Go Debug Terminal Help
product-feature.definition.json - Msdyn365.Commerce.Online - Visual Studio Code
EXPLORER
OPEN EDITORS
product-feature.definition.json... 1, U
MSDYN365.COMMERCE.ONLINE
.ssl
.vscode
build
dist
lib
public
src
modules
helloworld
product-feature
mocks
tests
product-feature.data.ts U
product-feature.definition.json 1, U
product-feature.props.autogen... U
product-feature.tsx U
product-feature.view.tsx U
pageMocks
settings
Readme.md
starter-pack
test
.env M
.gitattributes
.gitignore
OUTLINE
NPM SCRIPTS
src > modules > product-feature > product-feature.definition.json > ...
master* ⊗ 0 ⚠ 1 Auto Attach: On
Ln 1, Col 1 Spaces: 4 UTF-8 LF JSON

```json
{
    "$type": "contentModule",
    "friendlyName": "product-feature",
    "name": "product-feature",
    "description": "Please add meaningful description for authoring",
    "categories": ["product-feature"],
    "tags": ["Additional tags go here"],
    "dataActions": {
    },
    "config": {
        "showText": {
            "friendlyName": "showText",
            "description": "example config value",
            "type": "string",
            "default": "Example Config Value",
            "scope": "module",
            "group": "Layout Properties"
        }
    },
    "resources": {
        "resourceKey": {
            "comment": "resource description",
            "value": "resource value"
        }
    }
}
```

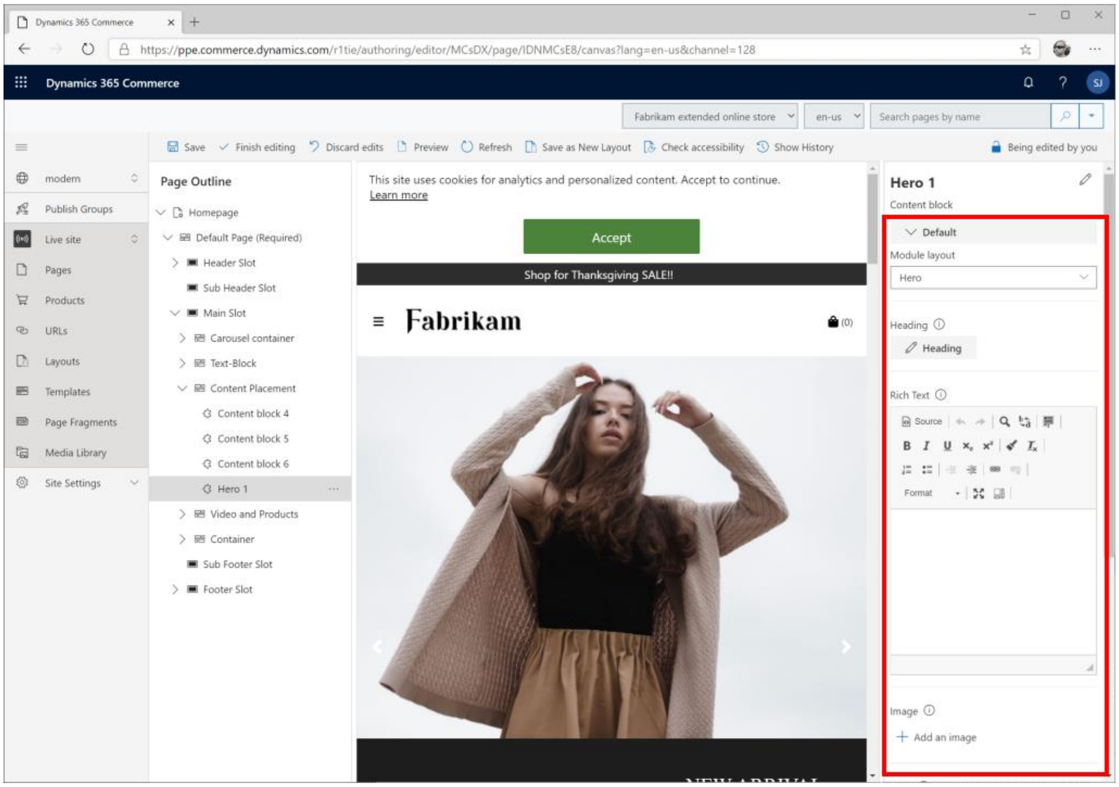
Below screen shot shows where the module friendly name, description and category show up.



Module Name ↑	Module Type	Categories	Description
Account address tile	Account address tile	Account management	Account address tile is used to show a summary of addresses associated to the account in the account management page
☑ Account generic tile	Account generic tile	Account management	Account generic item is used to show static information in the account management page
Account loyalty tile	Account loyalty tile	Account management	Account loyalty tile is used to show a summary of loyalty card information associated to the account in the account management page
Account management shipping address	Account management shi...	Address	Shipping address for account management flow
Account welcome tile	Account welcome tile	Account management	Account welcome item is used to show a welcome message to the user
Account wishlist tile	Account wishlist tile	Account management	Display a summary of account wishlists.
Autosuggest	Autosuggest	Autosuggest	Search auto suggestion module

The definition file is also used to expose “config” fields, which allow a page author to set module settings. An example may be a layout alignment setting (with left, right and center values to choose from), a module title or heading, rich text description, call to action link, image URLs or Dynamics Retail product data.

The page author can then choose the specific configurations for the module on a specific page without affecting the setting of the module on other pages. Below is an example of what a page author would see, note the right-side panel has various config settings for the selected module.


Dynamics 365 Commerce
https://ppe.commerce.dynamics.com/r1tie/authoring/editor/MCsDX/page/IDNMCsE8/canvas?lang=en-us&channel=128
Dynamics 365 Commerce
Fabrikam extended online store
en-us
Search pages by name
Save
Finish editing
Discard edits
Preview
Refresh
Save as New Layout
Check accessibility
Show History
Being edited by you
modern
Publish Groups
Live site
Pages
Products
URLs
Layouts
Templates
Page Fragments
Media Library
Site Settings
Page Outline
Homepage
Default Page (Required)
Header Slot
Sub Header Slot
Main Slot
Carousel container
Text-Block
Content Placement
Content block 4
Content block 5
Content block 6
Hero 1
Video and Products
Container
Sub Footer Slot
Footer Slot
This site uses cookies for analytics and personalized content. Accept to continue.
Learn more
Accept
Shop for Thanksgiving SALE!!
Fabrikam
(0)
Hero 1
Content block
Default
Module layout
Hero
Heading
Heading
Rich Text
Source
Format
Image
Add an image

DEFINITION SCHEMA

$type: The type of the module. A module can be either a content module (**contentModule**), a container module (**containerModule**), a page module (**pageModule**), a script injector module (**scriptModule**) or a theme module (**themeModule**). Container and page modules also optionally define “slots” which are used for layout regions. Script injector modules also define an "attributes" section that is used for specifying where script can be injected.

friendlyName: The friendly name of the module that gets displayed to page authors. Minimum length is three characters.

name: The name of the module. This name must be unique across the application. It is used as the ID of the module and is referenced by the site builder tool. It should not be changed.

description: The description of the module. The description provides a friendly string that is shown in the authoring tools when modules are added to pages.

categories: The categories that the module can subscribe to. Container modules use the values that are specified here to allow or disallow some modules in specific slots.

tags: The tags that are used to search for the module. All the categories are automatically added as tags.

dataActions: The data actions node is used to register the data actions that should be run for the module.

slots: Slots are defined only on “containerModules” and exposed in the authoring tool. Slots can define allow and deny lists to allow or disallow specific modules from being accepted in the slot.

attributes: Attributes are used to control script injectors properties.

MODULE CONFIG SCHEMA

Configuration fields can be defined that are surfaced in the authoring tool for page authors to provide data.

Config name: The local name that is used to access the configuration values from your React source code. Note this name is not case sensitive thus if you define multiple configs with the same name but only case differences they will be treated as the same and cause an error.

friendlyName: The friendly name that is shown as the configuration name in the authoring tools.

description: The description that is shown as the configuration description in the authoring tools.

type: The type of the configuration. Possible values are "**string**", "**bool**", "**number**", "**integer**", "**richText**", "**image**", "**imageSettings**", “**css**”, "**video**", and "**array**".

NOTE ON CSS CONFIG TYPE:

A CSS type must specify a set of string enums of possible classes to apply to a module and only one of the enum options can be selected for a given module config property. When an enum option is selected the selected class will be appended to the list of classes to pass down to the module in the format propertyName__propertyValue via this.props.config.className. Note that config types with type css will not be directly accessible from this.props.config as they are merged into this.props.config.className.

enum: For an enumerator type, the value must be set to "**string**".

default: The default value that is set if no value is set in the authoring tools.

scope: This field is used to scope the configuration to either a specific module instance or all modules on the site. Possible values are "**module**" and "**site**". If the value is set to "**site**", the module configuration does not appear on a page and can't be configured there. It appears and can be configured only at the site level. In this way, the value can be set one time for the whole site.

group: Groups are used to organize the configurations into organized groups in the authoring tools.

required: A flag that specifies whether a property must be set on the module. When "required" is set to true the authoring tools will show an error if the required property is not set, and an error will be displayed when the module is rendered.

MODULE RESOURCE SCHEMA

resources: Used for localizing resources. When resources strings are defined, the localized strings are pulled from corresponding json files. These files are stored under the /src/resources/modules/ directory and include a global.json for default locale values and any needed localized json files, example fr-fr.json.

resourcekey: The name of the resource. Resource keys can then be accessed in code via the this.props.resources.resourceKey property.

comment: Comment is a string used to identify the purpose of the string to aid in localization.

value: Resource string data that will be used in the module.

MODULE DEFINITIONS SCHEMA

definitions: This field can contain complex config type definitions, which can be referenced in the config sections as extended types.

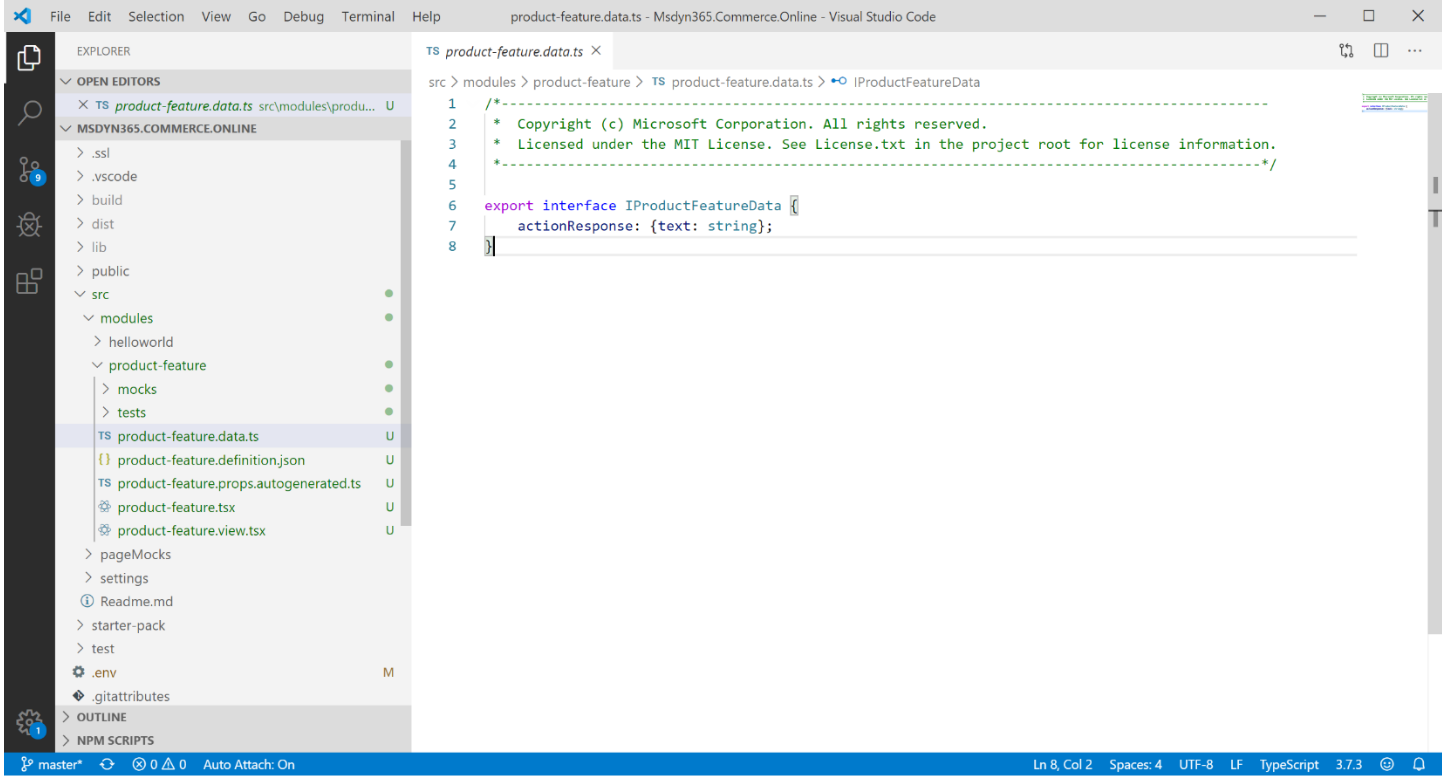
Below is a sample that shows the usage of various supported data types:

```
{
  "$type": "contentModule",
  "friendlyName": "Sample Config",
  "name": "sample-config",
  "description": "Sample Config",
  "categories": ["sample-config"],
  "tags": ["samples"],
  "dataActions": {},
  "config": {
      "title": {
          "type": "string",
          "friendlyName": "Title",
          "description": "example config value",
          "default": "Example Config Value",
          "scope": "module",
          "group": "Layout Properties"
      },
      "subTitle": {
        "type": "richText",
        "friendlyName": "SubTitle",
        "description": "Sub title rich text field"
      },
      "bgImage": {
          "type": "image",
          "friendlyName": "Background image",
          "description": "Background image"
      },
      "images": {
          "type": "array",
          "friendlyName": "Images",
          "description": "Image Array",
          "items": {
              "type": "image"
          }
      },
      "backgroundImageSettings": {
          "type": "imageSettings"
          "friendlyName": "Background Image Settings",
          "description": "Image settings for background iamge settings",
      },
      "ambientVideo": {
          "type": "video"
          "friendlyName": "Ambient Video",
          "description": "Ambient Video",
      },
      "headingArray":{
          "type": "array",
          "friendlyName": "Heading Array",
          "description": "Heading Array",
          "items": {
              "$ref": "#/definitions/heading"
          }
      },
      "heading":{
          "$ref": "#/definitions/heading"
      },
      "heading2":{
          "type": "object",
          "friendlyName": "Heading2",
          "description": "Heading2 property with its own enum",
          "properties": {
              "style": {
                  "type": "string",
                  "enum": {
                      "bold": "Bold",
                      "underline": "Underline",
                      "italics": "Italics",
                      "strong": "Strong",
                      "emphasized": "Emphasized",
                      "none": "None"
                  },
                  "friendlyName": "Style",
                  "description": "Heading style"
```

```
                }
            }
        },
        "textPlacement":{
            "friendlyName": "Text placement",
            "description": "Placement of the text",
            "type": "object",
            "enum":{
                "left": "Left",
                "right": "Right",
                "center": "Center"
            },
            "default": "left"
        }
    },
    "definitions": {
        "heading": {
            "type": "object",
            "friendlyName": "Heading",
            "description": "Heading property",
            "properties": {
                "text": {
                    "type": "string",
                    "friendlyName": "Text",
                    "description": "Heading Text"
                },
                "style": {
                    "type": "string",
                    "enum": {
                        "bold": "Bold",
                        "underline": "Underline",
                        "none": "None"
                    },
                    "friendlyName": "Style",
                    "description": "Heading style"
                },
                "showImage":{
                    "type":"boolean",
                    "friendlyName": "Show Image?",
                    "description": "Should Show Image"
                },
                "bgImage": {
                    "type": "image",
                    "friendlyName": "Background image",
                    "description": "Background image"
                },
                "imageArray":{
                    "type": "array",
                    "friendlyName": "Images",
                    "description": "Image Array",
                    "items": {
                        "type": "image"
                    }
                }
            }
        }
    }
}
```

MODULE_NAME.DATA.TS

This *.data.ts file contains the typings for data actions that are used by the module to fetch data. Once a data action gets its data it is stored in the properties in this file.

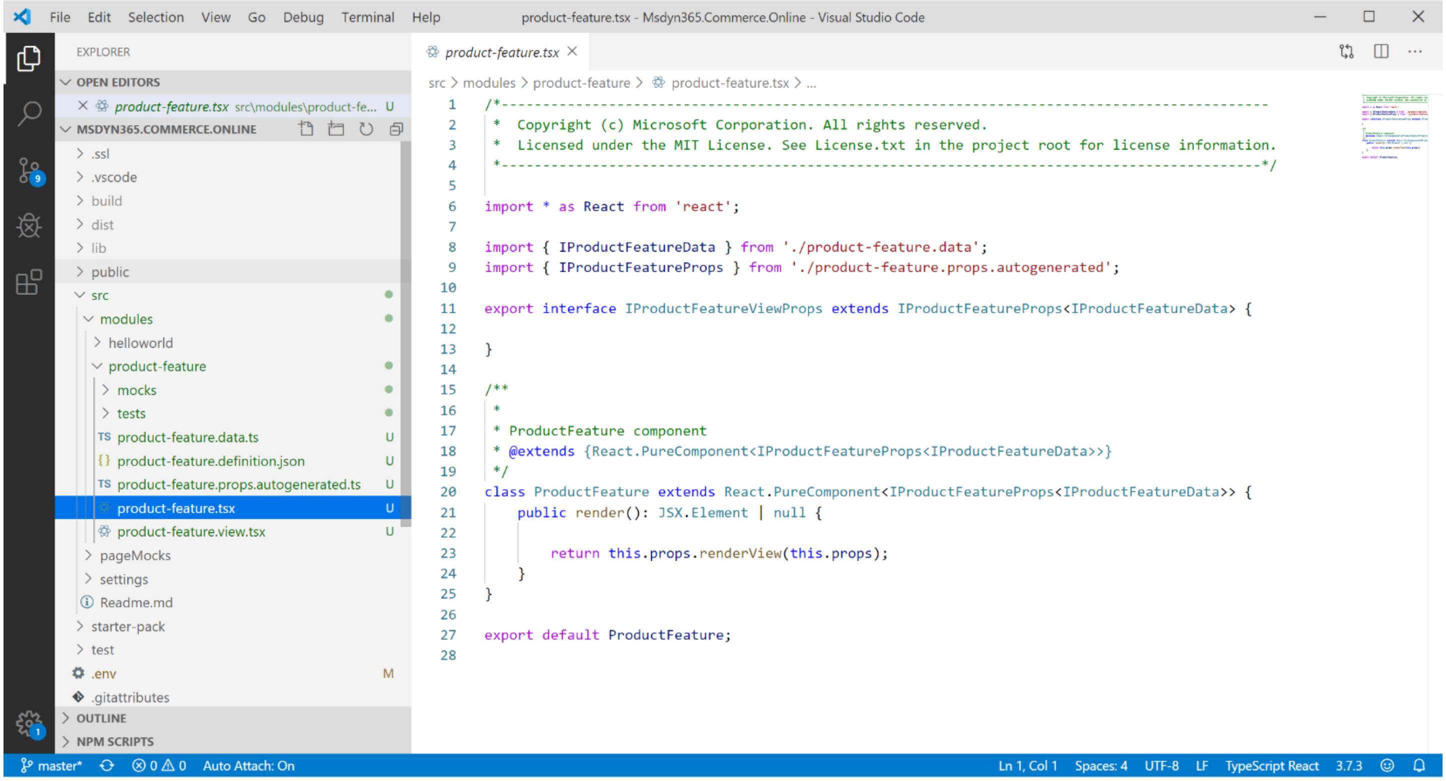


File Edit Selection View Go Debug Terminal Help
product-feature.data.ts - Msdyn365.Commerce.Online - Visual Studio Code
EXPLORER
OPEN EDITORS
TS product-feature.data.ts src\modules\produ... U
MSDYN365.COMMERCE.ONLINE
.ssl
.vscode
build
dist
lib
public
src
modules
helloworld
product-feature
mocks
tests
product-feature.data.ts U
product-feature.definition.json U
product-feature.props.autogenerated.ts U
product-feature.tsx U
product-feature.view.tsx U
pageMocks
settings
Readme.md
starter-pack
test
.env M
.gitattributes
OUTLINE
NPM SCRIPTS
product-feature.data.ts
src > modules > product-feature > TS product-feature.data.ts > IProductFeatureData
master* ⊗ 0 ⚠ 0 Auto Attach: On
Ln 8, Col 2 Spaces: 4 UTF-8 LF TypeScript 3.7.3

```
/*---------------------------------------------------------------------------------------------
 *  Copyright (c) Microsoft Corporation. All rights reserved.
 *  Licensed under the MIT License. See License.txt in the project root for license information.
 *--------------------------------------------------------------------------------------------*/

export interface IProductFeatureData {
    actionResponse: {text: string};
}
```

MODULE_NAME.TSX

This file is a React component that controls the module's view model and calls the module view. You'll find a React "render()" method with some default embedded HTML.

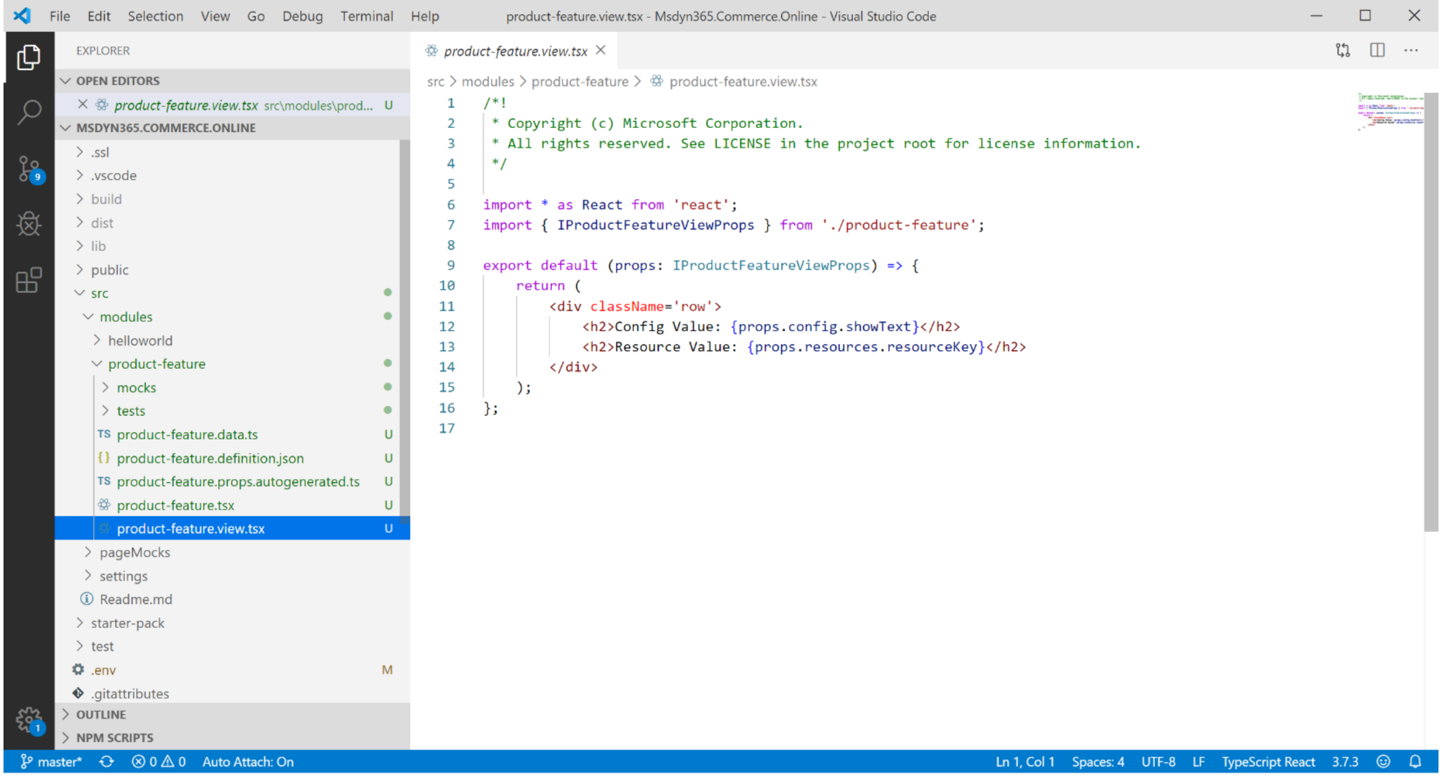

File Edit Selection View Go Debug Terminal Help
product-feature.tsx - Msdyn365.Commerce.Online - Visual Studio Code
EXPLORER
OPEN EDITORS
× product-feature.tsx src\modules\product-fe... U
MSDYN365.COMMERCE.ONLINE
.ssl
.vscode
build
dist
lib
public
src
modules
helloworld
product-feature
mocks
tests
product-feature.data.ts U
product-feature.definition.json U
product-feature.props.autogenerated.ts U
product-feature.tsx U
product-feature.view.tsx U
pageMocks
settings
Readme.md
starter-pack
test
.env M
.gitattributes
OUTLINE
NPM SCRIPTS
product-feature.tsx ×
src > modules > product-feature > product-feature.tsx > ...
master* ⊗ 0 ⚠ 0 Auto Attach: On
Ln 1, Col 1 Spaces: 4 UTF-8 LF TypeScript React 3.7.3

```tsx
/*---------------------------------------------------------------------------------------------
 *  Copyright (c) Microsoft Corporation. All rights reserved.
 *  Licensed under the MIT License. See License.txt in the project root for license information.
 *--------------------------------------------------------------------------------------------*/

import * as React from 'react';

import { IProductFeatureData } from './product-feature.data';
import { IProductFeatureProps } from './product-feature.props.autogenerated';

export interface IProductFeatureViewProps extends IProductFeatureProps<IProductFeatureData> {

}

/**
 *
 * ProductFeature component
 * @extends {React.PureComponent<IProductFeatureProps<IProductFeatureData>>}
 */
class ProductFeature extends React.PureComponent<IProductFeatureProps<IProductFeatureData>> {
    public render(): JSX.Element | null {

        return this.props.renderView(this.props);
    }
}

export default ProductFeature;
```

Configuration properties that you defined (in the module definition file) can be accessed within this file using the "this.props.config.*" API and data properties (as defined in the MODULE_NAME.data.ts file) can be accessed within the file using the this.props.data.* property.

MODULE_NAME.VIEW.TSX

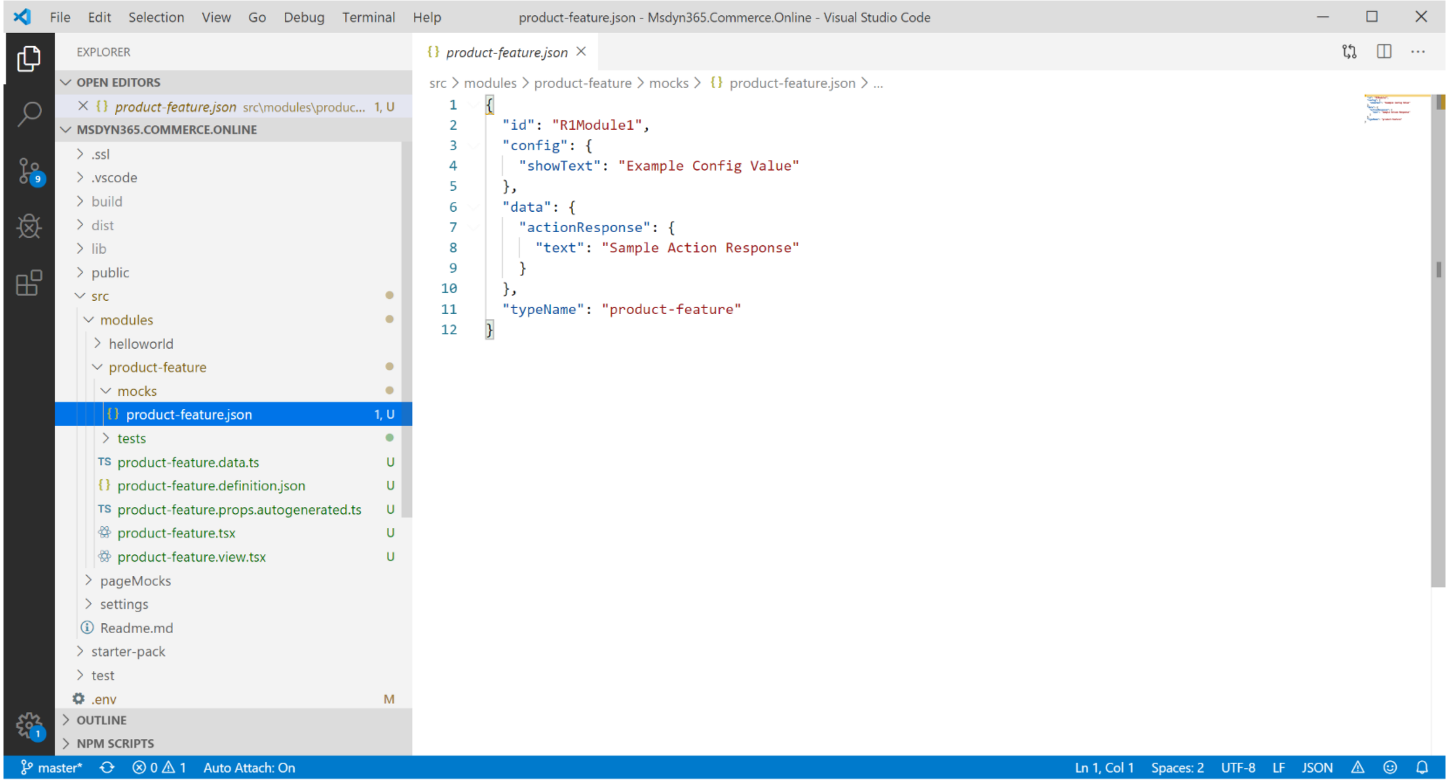
This file is the component that controls the module's view. Additional module views can be provided in various themes to render a module differently depending on the requirements of the theme.



```
/*!
 * Copyright (c) Microsoft Corporation.
 * All rights reserved. See LICENSE in the project root for license information.
 */

import * as React from 'react';
import { IProductFeatureViewProps } from './product-feature';

export default (props: IProductFeatureViewProps) => {
    return (
        <div className='row'>
            <h2>Config Value: {props.config.showText}</h2>
            <h2>Resource Value: {props.resources.resourceKey}</h2>
        </div>
    );
};
```

MOCKS/MODULE_NAME.JSON

This json file under the “mocks” directory is used to configure mock data for your module to be tested locally. As you can see in this file there are two data properties “myTitle” and “myText” which have a default “text” string field and the “showText” config property also has some default text. When modules are run on a production server this data comes from the Dynamics CMS database as configured by the page authors instead of the mock file.



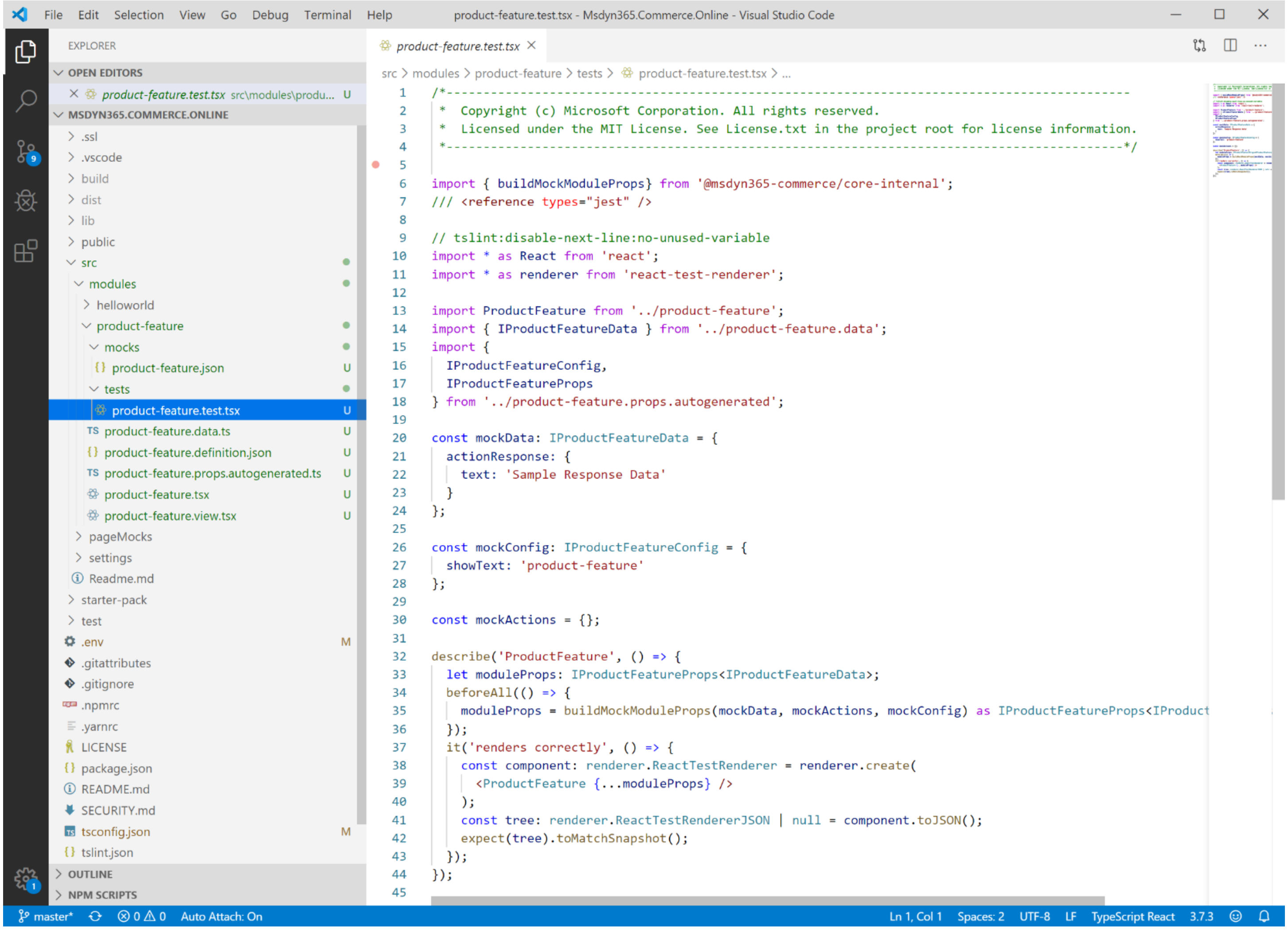
File Edit Selection View Go Debug Terminal Help
product-feature.json - Msdyn365.Commerce.Online - Visual Studio Code
EXPLORER
OPEN EDITORS
product-feature.json src\modules\produc... 1, U
MSDYN365.COMMERCE.ONLINE
.ssl
.vscode
build
dist
lib
public
src
modules
helloworld
product-feature
mocks
product-feature.json 1, U
tests
product-feature.data.ts U
product-feature.definition.json U
product-feature.props.autogenerated.ts U
product-feature.tsx U
product-feature.view.tsx U
pageMocks
settings
Readme.md
starter-pack
test
.env M
OUTLINE
NPM SCRIPTS
product-feature.json
src > modules > product-feature > mocks > product-feature.json > ...
master* ⊗ 0 ⚠ 1 Auto Attach: On
Ln 1, Col 1 Spaces: 2 UTF-8 LF JSON

```
{
  "id": "R1Module1",
  "config": {
    "showText": "Example Config Value"
  },
  "data": {
    "actionResponse": {
      "text": "Sample Action Response"
    }
  },
  "typeName": "product-feature"
}
```

Note: later in this guide you’ll learn about creating a “page mock” to test a page of modules together, where the above mock is only leveraged for testing a single module.

TESTS/MODULE_NAME.TEST.TSX

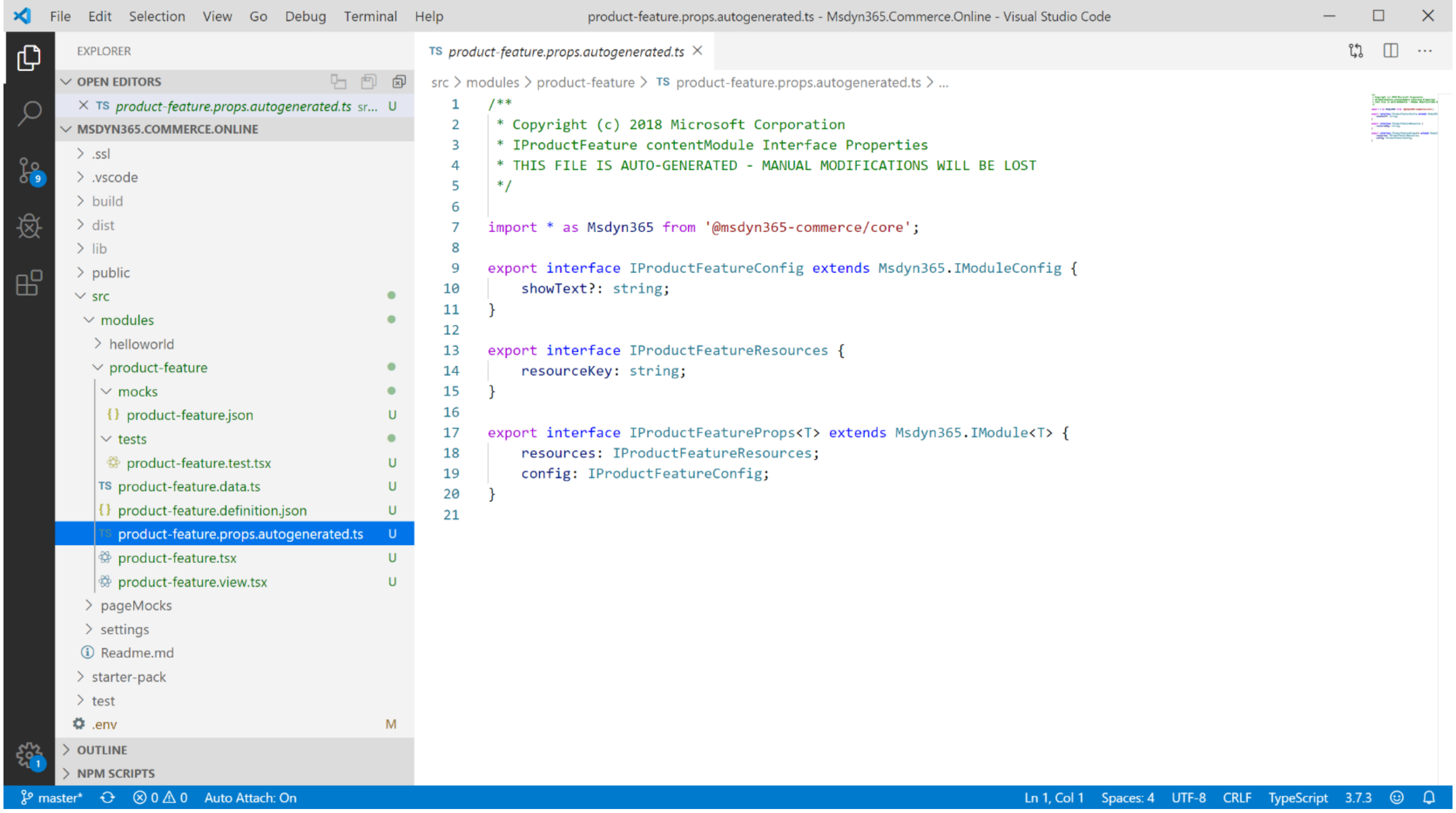
This file is used for local unit test and contains the mock data needed for the tests to run.


File Edit Selection View Go Debug Terminal Help
product-feature.test.tsx - Msdyn365.Commerce.Online - Visual Studio Code
EXPLORER
OPEN EDITORS
product-feature.test.tsx src\modules\produ... U
MSDYN365.COMMERCE.ONLINE
.ssl
.vscode
build
dist
lib
public
src
modules
helloworld
product-feature
mocks
product-feature.json U
tests
product-feature.test.tsx U
product-feature.data.ts U
product-feature.definition.json U
product-feature.props.autogenerated.ts U
product-feature.tsx U
product-feature.view.tsx U
pageMocks
settings
Readme.md
starter-pack
test
.env M
.gitattributes
.gitignore
.npmrc
.yarnrc
LICENSE
package.json
README.md
SECURITY.md
tsconfig.json M
tslint.json
OUTLINE
NPM SCRIPTS
product-feature.test.tsx
src > modules > product-feature > tests > product-feature.test.tsx > ...
master* ⊗ 0 ⚠ 0 Auto Attach: On
Ln 1, Col 1 Spaces: 2 UTF-8 LF TypeScript React 3.7.3

```
/*---------------------------------------------------------------------------------------------
 *  Copyright (c) Microsoft Corporation. All rights reserved.
 *  Licensed under the MIT License. See License.txt in the project root for license information.
 *--------------------------------------------------------------------------------------------*/

import { buildMockModuleProps} from '@msdyn365-commerce/core-internal';
/// <reference types="jest" />

// tslint:disable-next-line:no-unused-variable
import * as React from 'react';
import * as renderer from 'react-test-renderer';

import ProductFeature from '../product-feature';
import { IProductFeatureData } from '../product-feature.data';
import {
  IProductFeatureConfig,
  IProductFeatureProps
} from '../product-feature.props.autogenerated';

const mockData: IProductFeatureData = {
  actionResponse: {
    text: 'Sample Response Data'
  }
};

const mockConfig: IProductFeatureConfig = {
  showText: 'product-feature'
};

const mockActions = {};

describe('ProductFeature', () => {
  let moduleProps: IProductFeatureProps<IProductFeatureData>;
  beforeAll(() => {
    moduleProps = buildMockModuleProps(mockData, mockActions, mockConfig) as IProductFeatureProps<IProduct
  });
  it('renders correctly', () => {
    const component: renderer.ReactTestRenderer = renderer.create(
      <ProductFeature {...moduleProps} />
    );
    const tree: renderer.ReactTestRendererJSON | null = component.toJSON();
    expect(tree).toMatchSnapshot();
  });
});
```

MODULE_NAME.PROPS.AUTOGENERATED.TS

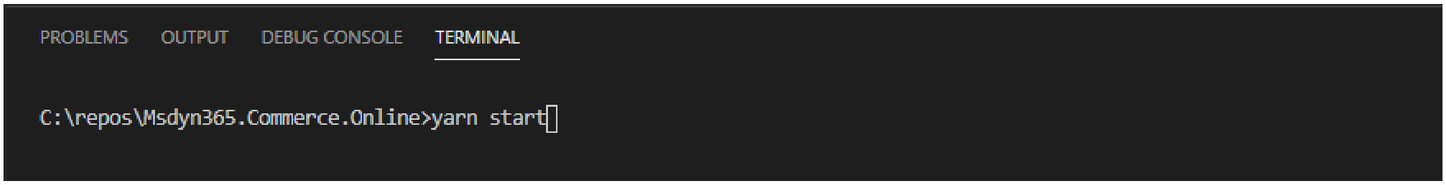
This file is an auto generated typescript file that should not be manually touched. Adding new configuration values to your module definition file and saving the file will automatically rebuild this file with appropriate interfaces. The interfaces in here can be accessed via the module view and react files as needed.


File Edit Selection View Go Debug Terminal Help
product-feature.props.autogenerated.ts - Msdyn365.Commerce.Online - Visual Studio Code
EXPLORER
OPEN EDITORS
TS product-feature.props.autogenerated.ts sr... U
MSDYN365.COMMERCE.ONLINE
.ssl
.vscode
build
dist
lib
public
src
modules
helloworld
product-feature
mocks
product-feature.json U
tests
product-feature.test.tsx U
product-feature.data.ts U
product-feature.definition.json U
product-feature.props.autogenerated.ts U
product-feature.tsx U
product-feature.view.tsx U
pageMocks
settings
Readme.md
starter-pack
test
.env M
OUTLINE
NPM SCRIPTS
TS product-feature.props.autogenerated.ts
src > modules > product-feature > TS product-feature.props.autogenerated.ts > ...
master* 0 0 Auto Attach: On
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF TypeScript 3.7.3

```
/**
 * Copyright (c) 2018 Microsoft Corporation
 * IProductFeature contentModule Interface Properties
 * THIS FILE IS AUTO-GENERATED - MANUAL MODIFICATIONS WILL BE LOST
 */

import * as Msdyn365 from '@msdyn365-commerce/core';

export interface IProductFeatureConfig extends Msdyn365.IModuleConfig {
    showText?: string;
}

export interface IProductFeatureResources {
    resourceKey: string;
}

export interface IProductFeatureProps<T> extends Msdyn365.IModule<T> {
    resources: IProductFeatureResources;
    config: IProductFeatureConfig;
}
```

We are now ready to test out our new module.

In the Command Prompt type in “**yarn start**”, this will build your solution as we did in the previous section.

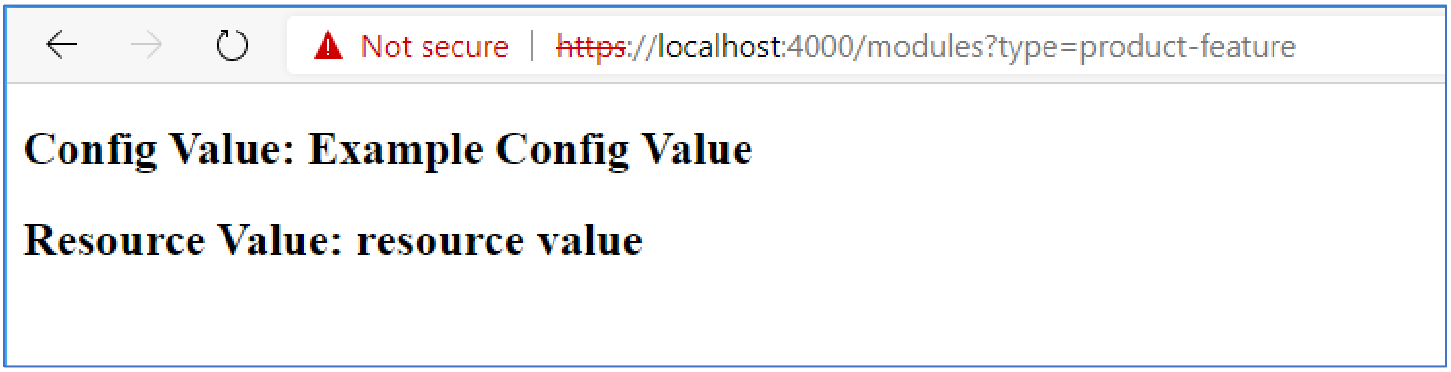


```
C:\repos\Msdyn365.Commerce.Online>yarn start
```

When the build is complete, we can load the module for testing locally with the below command (you may need to replace the module name product-feature’ with the name of your module).

https://localhost:4000/modules?type=product-feature

If everything works you should see your module render the html from the React component with the module template default as shown below:



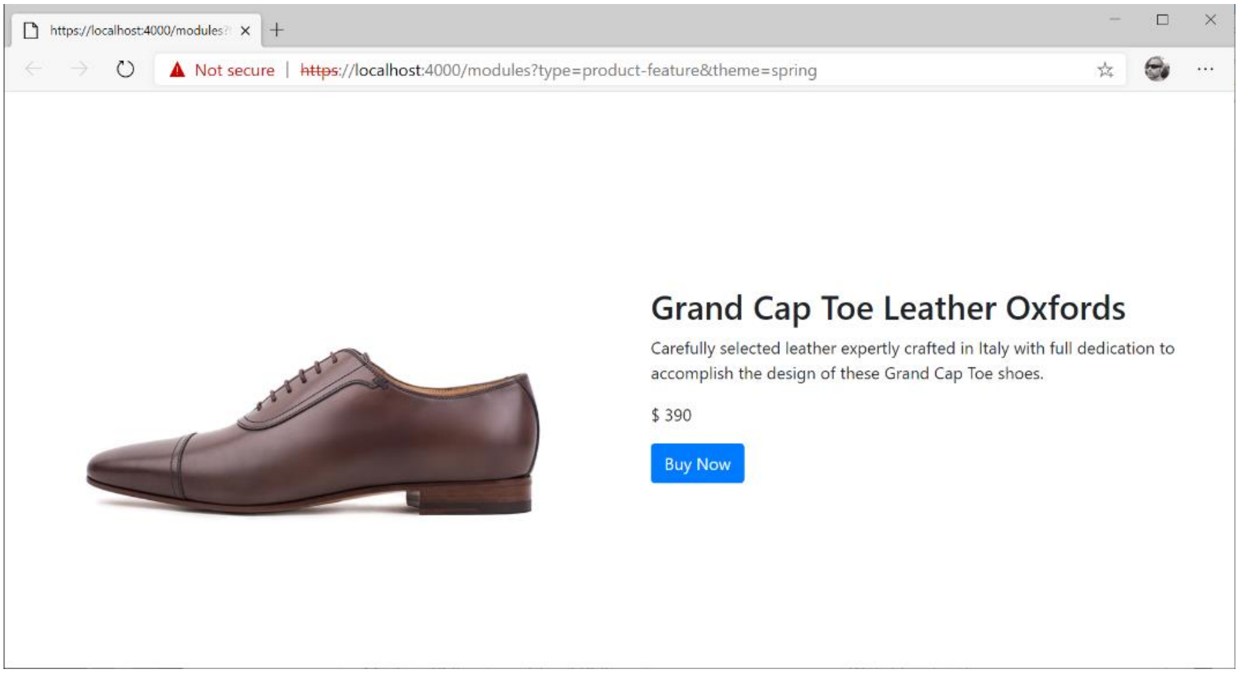
Config Value: Example Config Value

Resource Value: resource value

DESIGNING OUR MODULE

There are various ways you can start designing your module. The approach we will be taking is to build the HTML structure using a static image and text to achieve the desired mockup. In further sections we will convert the static HTML to dynamic HTML leveraging config and data properties and finally getting product data from the Dynamics 365 Retail catalog.

We want our design to look like the below screen shot, which has an image on the left, a product title and description on the right as well as a call to action button.

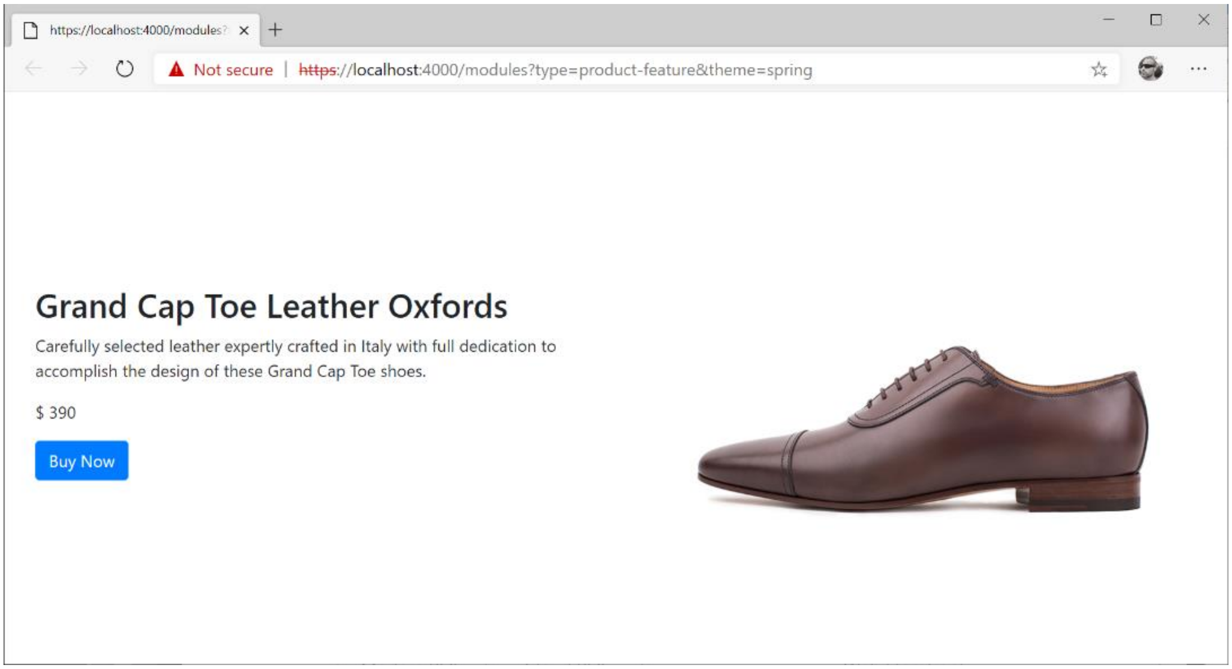

https://localhost:4000/modules?
Not secure
https://localhost:4000/modules?type=product-feature&theme=spring
Buy Now

Grand Cap Toe Leather Oxfords

Carefully selected leather expertly crafted in Italy with full dedication to accomplish the design of these Grand Cap Toe shoes.

$ 390

We'll also be adding a layout configuration property to swap the image alignment from the left to the right side:


https://localhost:4000/modules?
Not secure
https://localhost:4000/modules?type=product-feature&theme=spring
Buy Now

Grand Cap Toe Leather Oxfords

Carefully selected leather expertly crafted in Italy with full dedication to accomplish the design of these Grand Cap Toe shoes.

$ 390

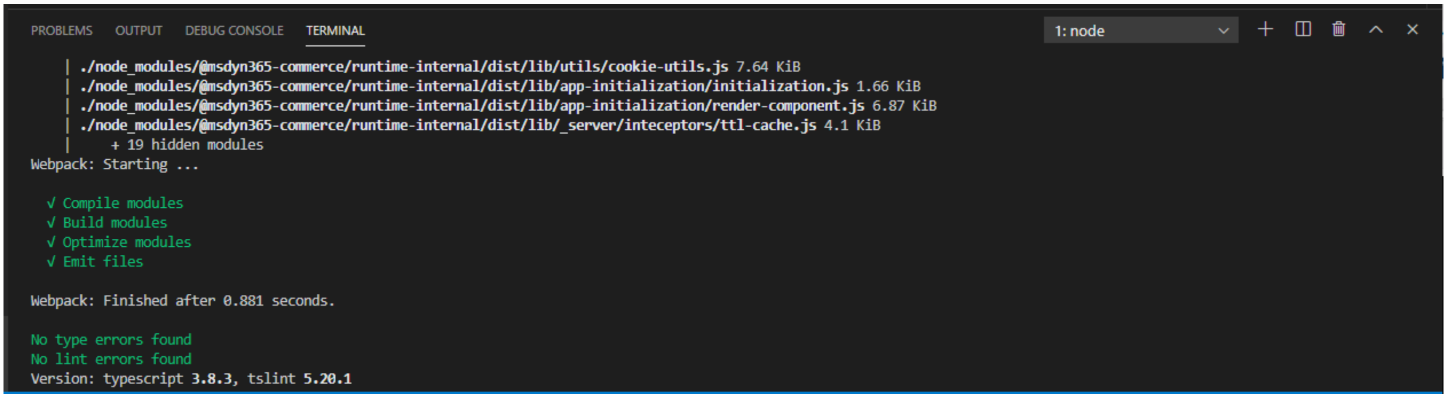
Start by adding static HTML to render the above mockup into our module React file "**product-feature.tsx**", add the below highlighted code (note you will be overriding the templated html code).

```
/*---------------------------------------------------------------------------------------------
 *  Copyright (c) Microsoft Corporation. All rights reserved.
 *  Licensed under the MIT License. See License.txt in the project root for license information.
 *--------------------------------------------------------------------------------------------*/

import * as React from 'react';

import { IProductFeatureData } from './product-feature.data';
import { IProductFeatureProps } from './product-feature.props.autogenerated';

export interface IProductFeatureViewProps extends IProductFeatureProps<IProductFeatureData> {
    productName: string;
    productInfo: string;
    productImageUrl: string;
    productPrice: string;
    buttonInfo: string;
}

/**
 *
 * ProductFeature component
 * @extends {React.PureComponent<IProductFeatureProps<IProductFeatureData>>}
 */
class ProductFeature extends React.PureComponent<IProductFeatureProps<IProductFeatureData>> {
    public render(): JSX.Element | null {
        // set default product info values
        const ProductName = 'Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses';
        const ProductInfo = 'High-quality and pioneered with the perfect blend of timeless classic and modern technology with hint of old school glamor.';
        const ProductImageUrl = 'https://bit.ly/33cMGxr';
        const ButtonInfo = 'Buy Now';
        const ProductPrice = '129';

        const ProductFeatureViewProps = {
            ...this.props,
            productName: ProductName,
            productInfo: ProductInfo,
            productImageUrl: ProductImageUrl,
            productPrice: ProductPrice,
            buttonInfo: ButtonInfo
        };
        return this.props.renderView(ProductFeatureViewProps);
    }
}

export default ProductFeature;
```

We'll also need to change our view file to render the new fields we're passing in, modify the product-feature.view.tsx file to match the below sample.

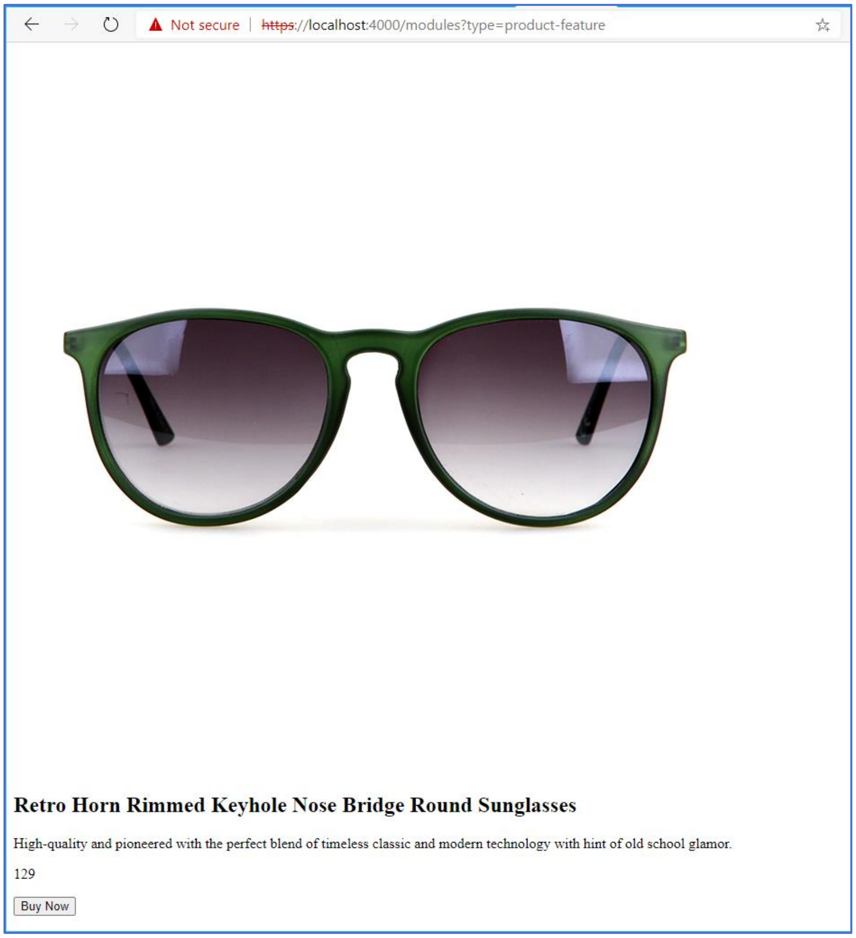
```
/*!
 * Copyright (c) Microsoft Corporation.
 * All rights reserved. See LICENSE in the project root for license information.
 */

import * as React from 'react';
import { IProductFeatureViewProps } from './product-feature';

export default (props: IProductFeatureViewProps) => {
    const { productName, productInfo, productImageUrl, productPrice, buttonInfo } = props;

    return (
        <div className='row align-items-center'>
            <div className='col-sm-6'>
              <img src={productImageUrl} alt={productName} className='img-fluid p-3' />
            </div>
            <div className='col-sm-6'>
              <h2>{productName}</h2>
              <p>{productInfo}</p>
              <p>{productPrice}</p>
              <button type='button' className='btn btn-primary'>{buttonInfo}</button>
            </div>
        </div>
    );
};
```

Once you save the above file, you should notice a recompile was triggered in the Command Prompt (that you previously ran "yarn start"), when it's complete the command prompt should look similar to below indicating no errors.



```
      | ./node_modules/@msdyn365-commerce/runtime-internal/dist/lib/utils/cookie-utils.js 7.64 KiB
      | ./node_modules/@msdyn365-commerce/runtime-internal/dist/lib/app-initialization/initialization.js 1.66 KiB
      | ./node_modules/@msdyn365-commerce/runtime-internal/dist/lib/app-initialization/render-component.js 6.87 KiB
      | ./node_modules/@msdyn365-commerce/runtime-internal/dist/lib/_server/inteceptors/ttl-cache.js 4.1 KiB
      |     + 19 hidden modules
Webpack: Starting ...

  √ Compile modules
  √ Build modules
  √ Optimize modules
  √ Emit files

Webpack: Finished after 0.881 seconds.

No type errors found
No lint errors found
Version: typescript 3.8.3, tslint 5.20.1
```

If you open https://localhost:4000/modules?type=product-feature you should see the module rendering with the above changes.



Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses

High-quality and pioneered with the perfect blend of timeless classic and modern technology with hint of old school glamor.

129

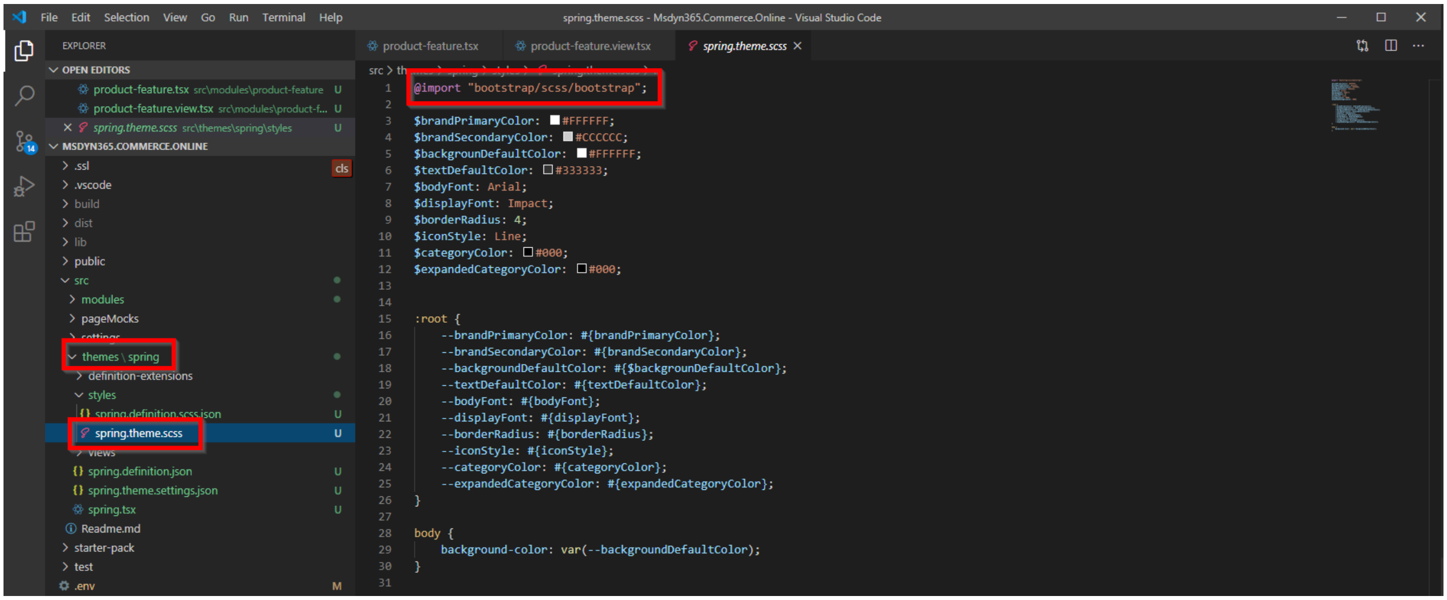
Notice the columns are not rendering as expected, this is because we are using Bootstrap SCSS classes in our html code but have not told the rendering to use Bootstap yet. For example, notice we have 2 div's with the class set to "col-sm-6", this should split each div into taking up half of the screen width for wider screens since Bootstrap defaults to a 12 column grid system.

To fix this we need to create a theme that pulls in the Bootstrap references. To create a new theme, we will use the CLI command "add-theme" and call our theme "spring". First stop "yarn start" if it is currently running and type the following command.

Note: you will need to hit ctrl-c in the command prompt to exit the Node server first.

```
yarn msdyn365 add-theme spring
```

This will create a new theme named “**spring**” under the “**themes**” directory. SCSS files for the theme are stored in the theme styles directory (example: “\src\themes\spring\styles”). Open the **\src\themes\spring \styles\spring.theme.scss** file and notice the reference (import) to Bootstrap is already included.

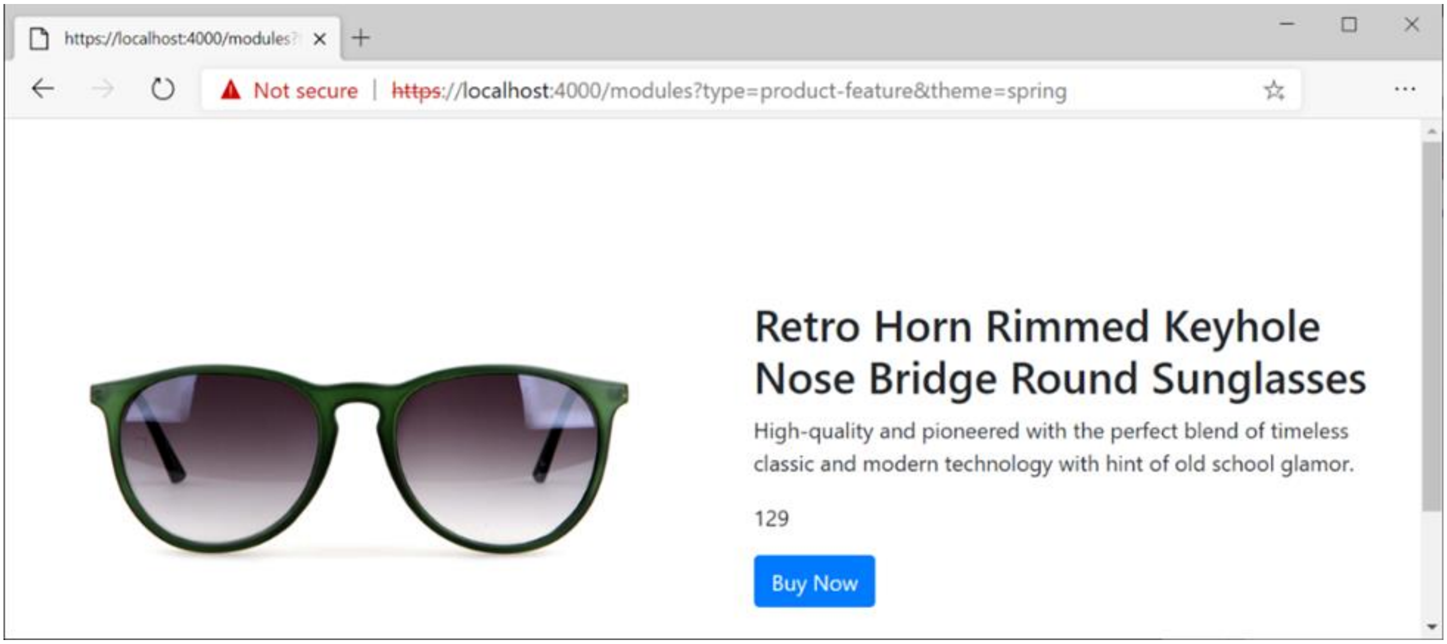

File Edit Selection View Go Run Terminal Help
spring.theme.scss - Msdyn365.Commerce.Online - Visual Studio Code
EXPLORER
OPEN EDITORS
product-feature.tsx src\modules\product-feature U
product-feature.view.tsx src\modules\product-f... U
spring.theme.scss src\themes\spring\styles U
MSDYN365.COMMERCE.ONLINE
.ssl
.vscode
build
dist
lib
public
src
modules
pageMocks
settings
themes \ spring
definition-extensions
styles
spring.definition.scss.json U
spring.theme.scss U
views
spring.definition.json U
spring.theme.settings.json U
spring.tsx U
Readme.md
starter-pack
test
.env M
product-feature.tsx
product-feature.view.tsx
spring.theme.scss

```scss
@import "bootstrap/scss/bootstrap";

$brandPrimaryColor: #FFFFFF;
$brandSecondaryColor: #CCCCCC;
$backgrounDefaultColor: #FFFFFF;
$textDefaultColor: #333333;
$bodyFont: Arial;
$displayFont: Impact;
$borderRadius: 4;
$iconStyle: Line;
$categoryColor: #000;
$expandedCategoryColor: #000;


:root {
    --brandPrimaryColor: #{brandPrimaryColor};
    --brandSecondaryColor: #{brandSecondaryColor};
    --backgroundDefaultColor: #{$backgrounDefaultColor};
    --textDefaultColor: #{textDefaultColor};
    --bodyFont: #{bodyFont};
    --displayFont: #{displayFont};
    --borderRadius: #{borderRadius};
    --iconStyle: #{iconStyle};
    --categoryColor: #{categoryColor};
    --expandedCategoryColor: #{expandedCategoryColor};
}

body {
    background-color: var(--backgroundDefaultColor);
}
```

A theme is a special type of a module, notice the theme also has a definition file “\src\themes\spring\spring.definition.json”. (The theme also has a basic tsx react file, which you will not need to modify).

```json
{
    "$type": "themeModule",
    "description": "This is SDK template theme module",
    "friendlyName": "spring",
    "name": "spring"
}
```

Now run **yarn start** again and add the theme query string parameter with the theme name to the URL : https://localhost:4000/modules?type=product-feature&theme=spring you should see the module rendering as expected.

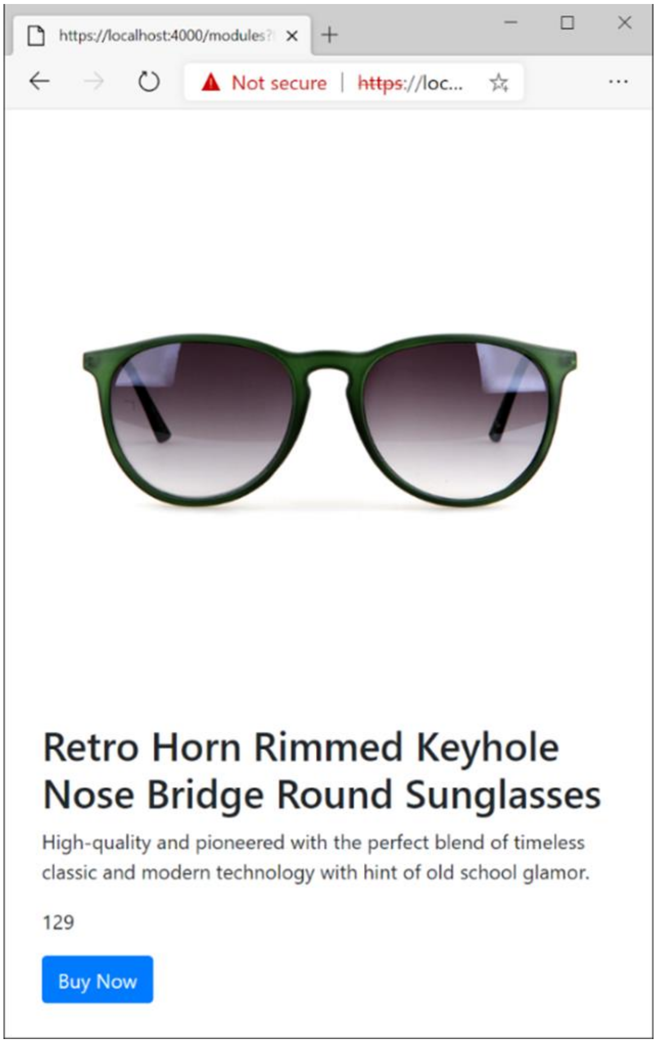

https://localhost:4000/modules?
Not secure
https://localhost:4000/modules?type=product-feature&theme=spring

Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses

High-quality and pioneered with the perfect blend of timeless classic and modern technology with hint of old school glamor.

129

It is also worth noting that if you resize the browser, you'll see the Bootstrap responsive design in action. Below screen shot shows the windows shrunk down to about a mobile size.



Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses

High-quality and pioneered with the perfect blend of timeless classic and modern technology with hint of old school glamor.

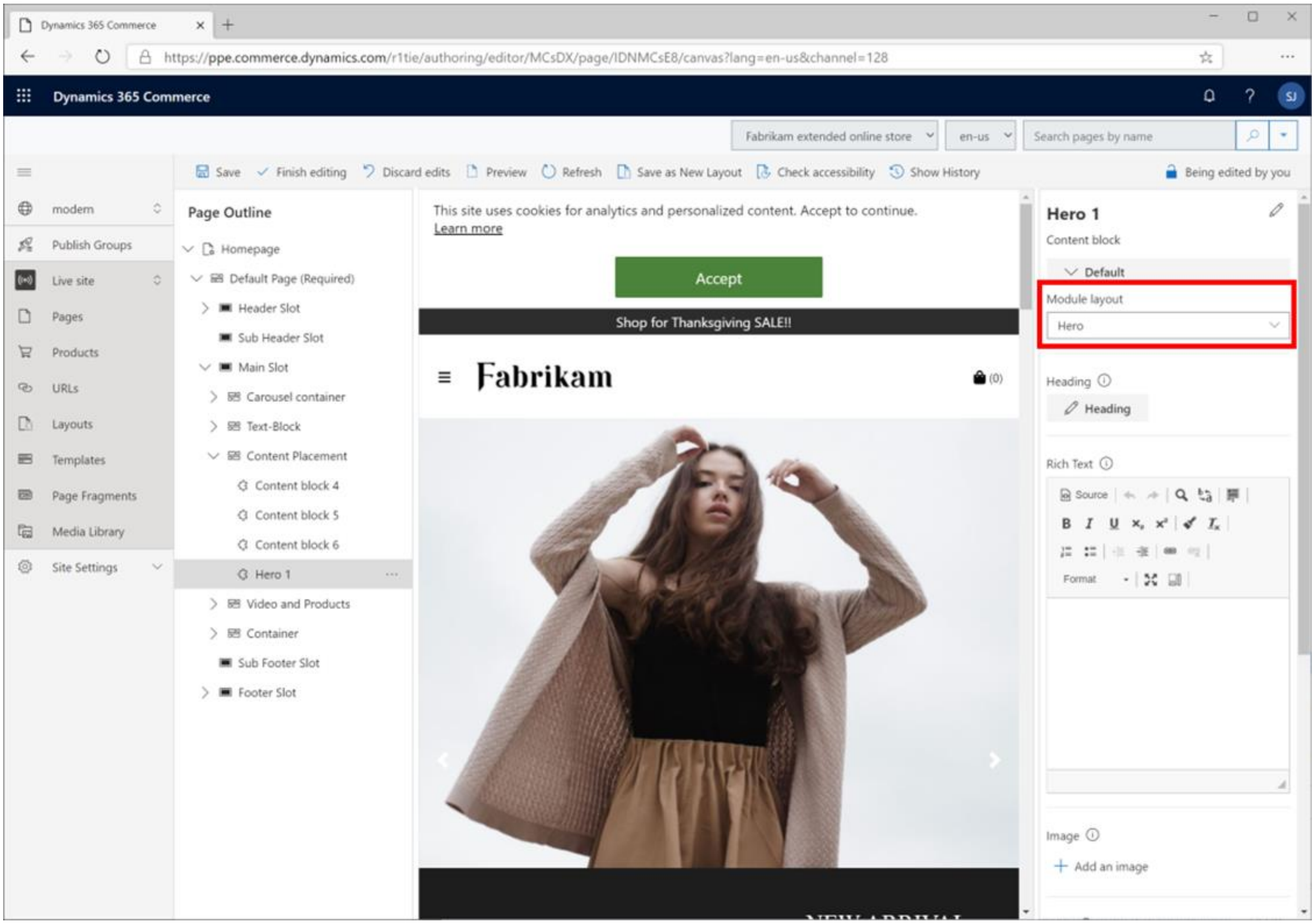
129

ADDING MODULE LAYOUT CONFIGURATION FIELDS

Configurations fields (configs) can be exposed on a module that will be presented to page authors, giving them control of some of the module features. These features can be layout properties such as different views, alignment properties or even boolean switches to turn features on or off or could be heading strings, rich text marketing details, links or images.

This data is stored in the Dynamics CMS data store and can range from simple to complex data structures.

Below is a screen shot of how these fields show up in the site builder tool. Notice with the “Content block” module selected in the tree, within the right panel you’ll see available configuration fields including one for “Module layout”.



https://ppe.commerce.dynamics.com/r1tie/authoring/editor/MCsDX/page/IDNMCsE8/canvas?lang=en-us&channel=128

This site uses cookies for analytics and personalized content. Accept to continue. Learn more

Hero 1
Content block

Module layout: Hero

For our module we are going to add a configuration field to allow configuration of the image alignment. We will have two enum options: “Left” which will be the default and “Right”.

Open up the **product-feature.definition.json** file, you should still see a default “showText” config setting in there, we’ll replace it with the below “imageAlignment” config. Note, we can also add a better friendly name, description and category for our module, see section 2 under definition file details to see where these will show up in the authoring tools.

```
{
    "$type": "contentModule",
    "friendlyName": "Product Feature",
    "name": "product-feature",
    "description": "Feature module used to highlight a product.",
    "categories": [
        "storytelling"
    ],
    "tags": [
        ""
    ],
    "dataActions": {},
    "config": {
        "imageAlignment": {
            "friendlyName": "Image Alignment",
            "description": "Sets the desired alignment of the image, either left or right on the text.",
            "type": "string",
            "enum": {
                "left": "Left",
                "right": "Right"
            },
            "default": "left",
            "scope": "module",
            "group": "Layout Properties"
        }
    },
    "resources": {
        "resourceKey": {
            "comment": "resource description",
            "value": "resource value"
        }
    }
}
```

Take note of the “**scope**” property above. Module configs can be scoped to the “**module**” or “**site**” level if you’d like the module config value to be set at the site level only. If it is set at the site level it will become the default and can’t be changed at the page level.

We will now set a mock value for the new config field we added above in the “mocks/product-feature.json” file. This will help us when rendering the module in our local dev environment.

```json
{
  "id": "R1Module1",
  "config": {
    "imageAlignment": "left"
  },
  "data": {
    "actionResponse": {
      "text": "Sample Action Response"
    }
  },
  "typeName": "product-feature"
}
```

***Note**: you may notice yarn build errors reported on the ‘tests/product-feature.test.tsx’ file as you add/remove config types and save the file. We’ll leave it to you as an exercise to fix these, however you should be able to ignore the errors or delete the test file altogether if desired.*

The final change we need is to hook up the new config field in code to render the image on the left or the right of the text area.

The below code change in product-feature.tsx will pass on the new config property in the props to the module view.

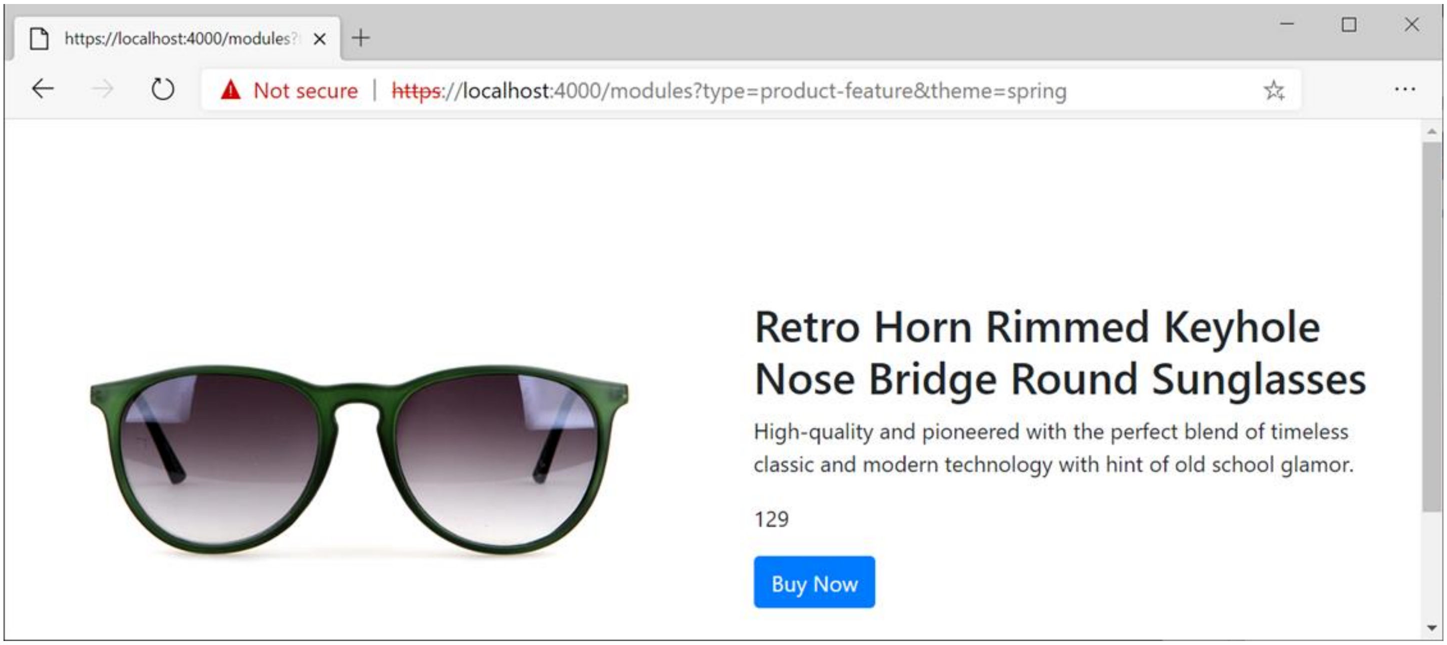
```tsx
/*---------------------------------------------------------------------------------------------
 *  Copyright (c) Microsoft Corporation. All rights reserved.
 *  Licensed under the MIT License. See License.txt in the project root for license information.
 *--------------------------------------------------------------------------------------------*/

import * as React from 'react';

import { IProductFeatureData } from './product-feature.data';
import { imageAlignment, IProductFeatureProps } from './product-feature.props.autogenerated';

export interface IProductFeatureViewProps extends IProductFeatureProps<IProductFeatureData> {
    productName: string;
    productInfo: string;
    productImageUrl: string;
    productPrice: string;
    buttonInfo: string;
    alignment: imageAlignment;
}

/**
 *
 * ProductFeature component
 * @extends {React.PureComponent<IProductFeatureProps<IProductFeatureData>>}
 */
class ProductFeature extends React.PureComponent<IProductFeatureProps<IProductFeatureData>> {
    public render(): JSX.Element | null {
        const { config } = this.props;

        // set default product info values
        const ProductName = 'Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses';
        const ProductInfo = 'High-quality and pioneered with the perfect blend of timeless classic and modern technology with hint of old school glamor.';
        const ProductImageUrl = 'https://bit.ly/33cMGxr';
        const ButtonInfo = 'Buy Now';
        const ProductPrice = '129';

        const ProductFeatureViewProps = {
            ...this.props,
            productName: ProductName,
            productInfo: ProductInfo,
            productImageUrl: ProductImageUrl,
            productPrice: ProductPrice,
            buttonInfo: ButtonInfo,
            alignment: config.imageAlignment
        };

        return this.props.renderView(ProductFeatureViewProps);
    }
}

export default ProductFeature;
```

Finally, the below code change in product-feature.view.tsx uses an “if” statement to check the new imageAlignment config field and renders the appropriate HTML. When finished save the files and it will auto compile (check your command prompt window where you ran “yarn start”).

```
/*!
 * Copyright (c) Microsoft Corporation.
 * All rights reserved. See LICENSE in the project root for license information.
 */

import * as React from 'react';
import { IProductFeatureViewProps } from './product-feature';
import { imageAlignment } from './product-feature.props.autogenerated';

const _renderImage=(productImageUrl: string, productName: string): JSX.Element => {
    return <img src={productImageUrl} alt={productName} className='img-fluid p-3' />;
};
```

```
const _renderInfo=(productName: string, productInfo: string, productPrice: string, buttonInfo:
string): JSX.Element => {
    return (
        <div className='container'>
            <h2>{productName}</h2>
            <p>{productInfo}</p>
            <p>{productPrice}</p>
            <button type='button' className='btn btn-primary'>{buttonInfo}</button>
        </div>
    );
};

export default (props: IProductFeatureViewProps) => {
    const { productName, productInfo, productImageUrl, productPrice, buttonInfo, alignment } =
props;

    let left;
    let right;

    if (alignment === imageAlignment.left) {
        left = _renderImage(productImageUrl, productName);
        right = _renderInfo(productName, productInfo, productPrice, buttonInfo);
    } else {
        right = _renderImage(productImageUrl, productName);
        left = _renderInfo(productName, productInfo, productPrice, buttonInfo);
    }

    return (
        <div className='row align-items-center'>
            <div className='col-sm-6'>
                {left}
            </div>
            <div className='col-sm-6'>
                {right}
            </div>
        </div>
    );
};
```

Save all the above files you edited and refresh the module in your browser (https://localhost:4000/modules?type=product-feature&theme=spring). It should render with the image on either the left or right depending upon the mock value you set on the imageAlignment config field.

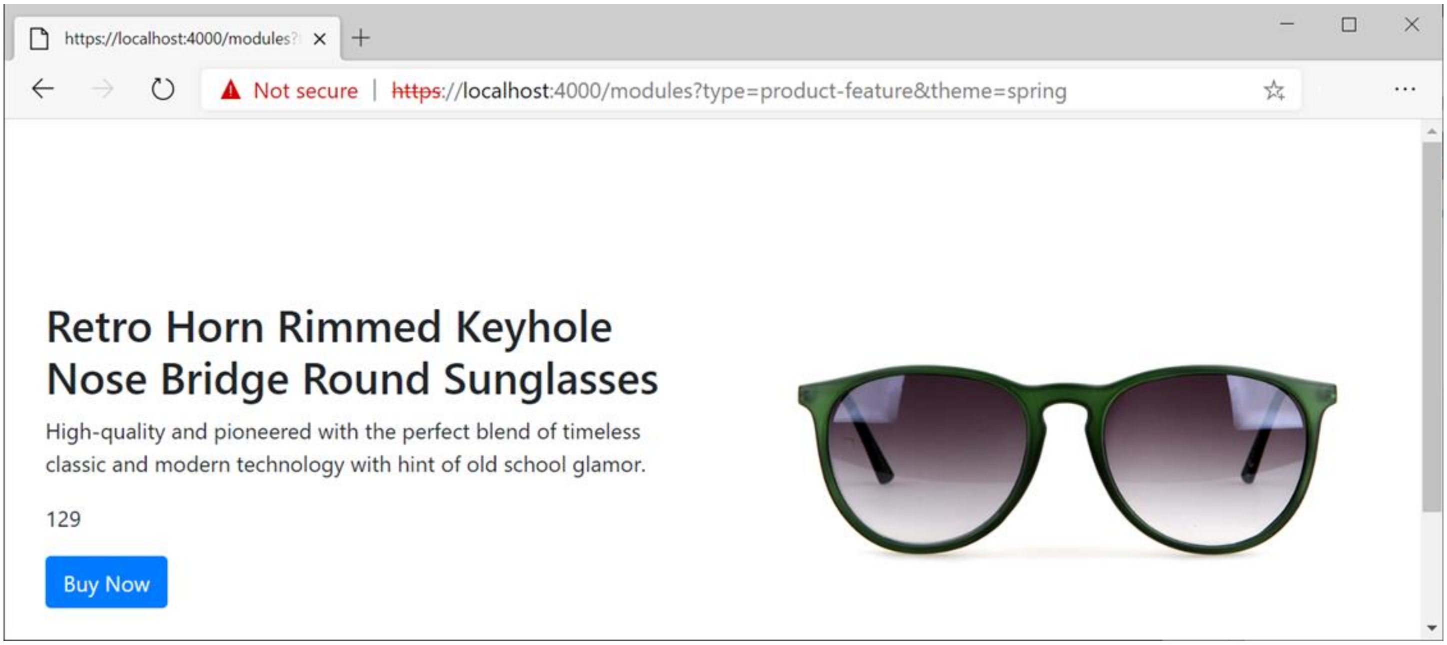

https://localhost:4000/modules?
Not secure
https://localhost:4000/modules?type=product-feature&theme=spring

Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses

High-quality and pioneered with the perfect blend of timeless classic and modern technology with hint of old school glamor.

129

Try changing the mock value, saving it then refreshing the browser to see the change.


https://localhost:4000/modules?
Not secure
https://localhost:4000/modules?type=product-feature&theme=spring

Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses

High-quality and pioneered with the perfect blend of timeless classic and modern technology with hint of old school glamor.

129

ADDING ADDITIONAL CONFIGURATION FIELDS TO A MODULE

In this section we will be replacing all the static content we added to our module with CMS Data configuration fields so page authors can provide this data from within the site builder tool.

For our product placement module, we will want fields for:

- Product title heading
- Product details paragraph
- Product image href and alt text
- Button label

Make the below changes to product-feature.definition.json and save the file.

```
{
    "$type": "contentModule",
    "friendlyName": "Product Feature",
    "name": "product-feature",
    "description": "Feature module used to highlight a product.",
    "categories": [
        "storytelling"
    ],
    "tags": [
        ""
    ],
    "dataActions": {},
    "config": {
        "imageAlignment": {
            "friendlyName": "Image Alignment",
            "description": "Sets the desired alignment of the image, either left or right on the t
ext.",
            "type": "string",
            "enum": {
                "left": "Left",
                "right": "Right"
            },
            "default": "left",
            "scope": "module",
            "group": "Layout Properties"
        },
        "productTitle": {
            "type": "string",
            "friendlyName": "Product Title",
            "description": "Product placement title"
        },
        "productDetails": {
            "type": "richText",
            "friendlyName": "Product Details",
            "description": "Rich text representing the featured product details"
        },
        "productImage": {
            "type": "image",
            "friendlyName": "Product Image",
            "description": "Image representing the featured product"
        },
        "buttonText": {
            "type": "string",
            "friendlyName": "Button Text",
            "description": "Text to show on the call to action button"
        }
    },
    "resources": {
        "resourceKey": {
            "comment": "resource description",
            "value": "resource value"
        }
    }
}
```

Next, we need to add mock data so we can test the fields out when we run it locally since we are not connected to the Dynamics Commerce CMS.

Open “/mocks/product-feature.json” and make the below changes to the “data” section:

```
{
  "id": "R1Module1",
  "config": {
    "imageAlignment": "left",
    "productTitle": "Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses",
    "productDetails": "High-
quality and pioneered with the perfect blend of timeless classic and modern technology with hint o
f old school glamor.",
    "productImage": {
      "src": "https://bit.ly/33cMGxr",
      "altText": "Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses"
    },
    "buttonText": "Buy Now"
  },
  "data": {
    "actionResponse": {
      "text": "Sample Action Response"
    }
  },
  "typeName": "product-feature"
}
```

The last thing we need to do is replace the static text in our React component (product-feature.tsx) with our new data fields. We’ll also do a safety check to make sure data was provided, otherwise we’ll add default text.

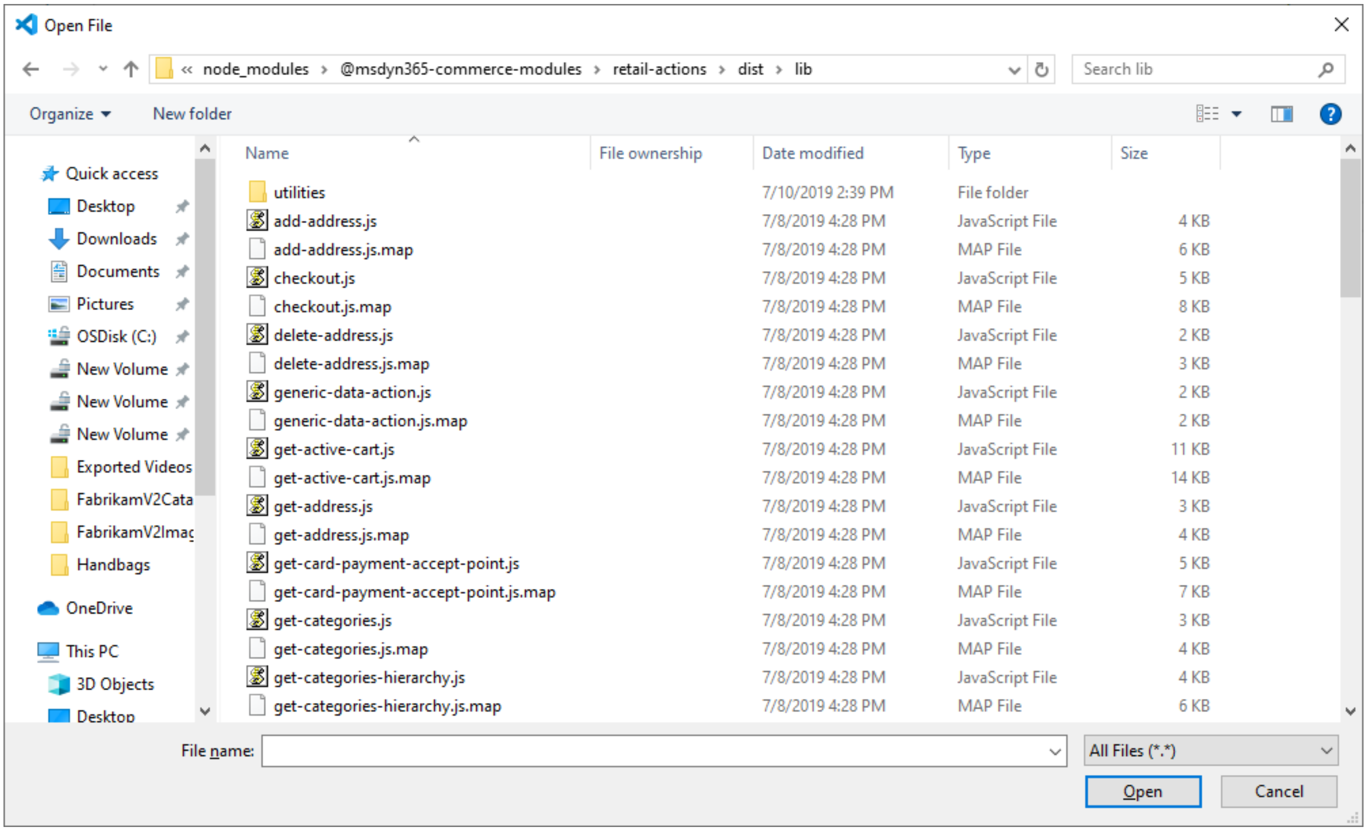
```
/*---------------------------------------------------------------------------------------------
 *  Copyright (c) Microsoft Corporation. All rights reserved.
 *  Licensed under the MIT License. See License.txt in the project root for license information.
 *--------------------------------------------------------------------------------------------*/

import * as React from 'react';

import { IProductFeatureData } from './product-feature.data';
import { imageAlignment, IProductFeatureProps } from './product-feature.props.autogenerated';

export interface IProductFeatureViewProps extends IProductFeatureProps<IProductFeatureData> {
    productName: string;
    productInfo: string;
    productImageUrl: string;
    productPrice: string;
    buttonInfo: string;
    alignment: imageAlignment;
}

/**
 *
 * ProductFeature component
 * @extends {React.PureComponent<IProductFeatureProps<IProductFeatureData>>}
 */
class ProductFeature extends React.PureComponent<IProductFeatureProps<IProductFeatureData>> {
    public render(): JSX.Element | null {
        const { config } = this.props;

        // set default product info values
        const ProductName = config.productTitle ? config.productTitle : 'No product name defined';
        const ProductInfo = config.productDetails ? config.productDetails.toString() : 'No product details defined';
        const ProductImageUrl = config.productImage ? config.productImage.src : '';
        const ButtonInfo = config.buttonText ? config.buttonText : 'No button text defined';
        const ProductPrice = '129';

        const ProductFeatureViewProps = {
            ...this.props,
            productName: ProductName,
            productInfo: ProductInfo,
            productImageUrl: ProductImageUrl,
            productPrice: ProductPrice,
            buttonInfo: ButtonInfo,
            alignment: config.imageAlignment
        };

        return this.props.renderView(ProductFeatureViewProps);
    }
}

export default ProductFeature;
```

Save the file and refresh your browser (https://localhost:4000/modules?type=product-feature&theme=spring) to see the new module now rendering content from CMS data mocks. You can try changing the data from the mocks/product-feature.json, saving it and a page refresh should pick up your change.

CALLING A PLATFORM CORE DATA ACTION

In this section we will call a data action to get product details into our module. This change will allow the page author to provide a product Id and we will pull the product title, details and price from the Dynamics Retail server. The online SDK provides several pre-build data actions for common functions like getting Retail categories or product information. A later section will cover creating your own custom data action.

To see a list of data actions that are available open (CTRL-O in VS Code) the “\node_modules\@msdyn365-commerce-modules\retail-actions\dist\lib” directory and you’ll see a similar list to below image.



Name	File ownership	Date modified	Type	Size
utilities		7/10/2019 2:39 PM	File folder	
add-address.js		7/8/2019 4:28 PM	JavaScript File	4 KB
add-address.js.map		7/8/2019 4:28 PM	MAP File	6 KB
checkout.js		7/8/2019 4:28 PM	JavaScript File	5 KB
checkout.js.map		7/8/2019 4:28 PM	MAP File	8 KB
delete-address.js		7/8/2019 4:28 PM	JavaScript File	2 KB
delete-address.js.map		7/8/2019 4:28 PM	MAP File	3 KB
generic-data-action.js		7/8/2019 4:28 PM	JavaScript File	2 KB
generic-data-action.js.map		7/8/2019 4:28 PM	MAP File	2 KB
get-active-cart.js		7/8/2019 4:28 PM	JavaScript File	11 KB
get-active-cart.js.map		7/8/2019 4:28 PM	MAP File	14 KB
get-address.js		7/8/2019 4:28 PM	JavaScript File	3 KB
get-address.js.map		7/8/2019 4:28 PM	MAP File	4 KB
get-card-payment-accept-point.js		7/8/2019 4:28 PM	JavaScript File	5 KB
get-card-payment-accept-point.js.map		7/8/2019 4:28 PM	MAP File	7 KB
get-categories.js		7/8/2019 4:28 PM	JavaScript File	3 KB
get-categories.js.map		7/8/2019 4:28 PM	MAP File	4 KB
get-categories-hierarchy.js		7/8/2019 4:28 PM	JavaScript File	4 KB
get-categories-hierarchy.js.map		7/8/2019 4:28 PM	MAP File	6 KB

We will be using the “**get-simple-products**” data action to get product details, which returns an array of products. We will start with declaring the data action we’re using in our module definition file:

```
{
    "$type": "contentModule",
    "friendlyName": "Product Feature",
    "name": "product-feature",
    "description": "Feature module used to highlight a product.",
    "categories": [
        "storytelling"
    ],
    "tags": [
        ""
    ],
    "dataActions": {
        "products": {
            "path": "@msdyn365-commerce-modules/retail-actions/dist/lib/get-simple-products",
            "runOn": "server"
        }
    },
    "config": {
        "imageAlignment": {
            "friendlyName": "Image Alignment",
            "description": "Sets the desired alignment of the image, either left or right on
the text.",
            "type": "string",
            "enum": {
                "left": "Left",
                "right": "Right"
            },
            "default": "left",
            "scope": "module",
            "group": "Layout Properties"
        },
        "productTitle": {
            "type": "string",
            "friendlyName": "Product Title",
            "description": "Product placement title"
        },
        "productDetails": {
            "type": "richText",
            "friendlyName": "Product Details",
            "description": "Rich text representing the featured product details"
        },
        "productImage": {
            "type": "image",
            "friendlyName": "Product Image",
            "description": "Image representing the featured product"
        },
        "buttonText": {
            "type": "string",
            "friendlyName": "Button Text",
            "description": "Text to show on the call to action button"
        },
        "productIds": {
            "friendlyName": "Product ID",
            "description": "Provide a Product Id that the module will display",
            "type": "string",
            "scope": "module",
            "group": "Content Properties"
        }
    },
    "resources": {
        "resourceKey": {
            "comment": "resource description",
            "value": "resource value"
        }
    }
}
```

Notice the “runOn” property is set to “server” which could also be set to “client”. You’ll generally always want data actions to run on the server but you may want to set to client while debugging.

Also, take note that a config property for the page author to provide a product ID in the product-feature.definition.json file was also added called “productIDs”. This config will also become the input for the core data action we are using. You can see this if you examine the Javascript source code for the data action (get-simple-products) not shown here.

We also need to define the return type from the data action in the product-feature.data.ts file along with the return type as seen in below highlighted section. Note how the name property “products” is defined in both the definition file above and file below.

```
/*---------------------------------------------------------------------------------------------
 *  Copyright (c) Microsoft Corporation. All rights reserved.
 *  Licensed under the MIT License. See License.txt in the project root for license information.
 *--------------------------------------------------------------------------------------------*/
import { AsyncResult, SimpleProduct } from '@msdyn365-commerce/retail-proxy';

export interface IProductFeatureData {
    products: AsyncResult<SimpleProduct>[];
}
```

Next, we’ll need specify a Product ID in our Mock file for testing, we’ll be using a known product ID from our sample data (which may not exist on your Retail server). Inside of the mocks\product-feature.json add the below highlighted code:

```

{
  "id": "R1Module1",
  "config": {
    "imageAlignment": "left",
    "productTitle": "Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses",
    "productDetails": "High-
quality and pioneered with the perfect blend of timeless classic and modern technology with hint o
f old school glamor.",
    "productImage": {
      "src": "https://bit.ly/33cMGxr",
      "altText": "Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses"
    },
    "buttonText": "Buy Now",
    "productIds": "68719498121"
  },
  "data": {
    "actionResponse": {
      "text": "Sample Action Response"
    }
  },
  "typeName": "product-feature"
}
```

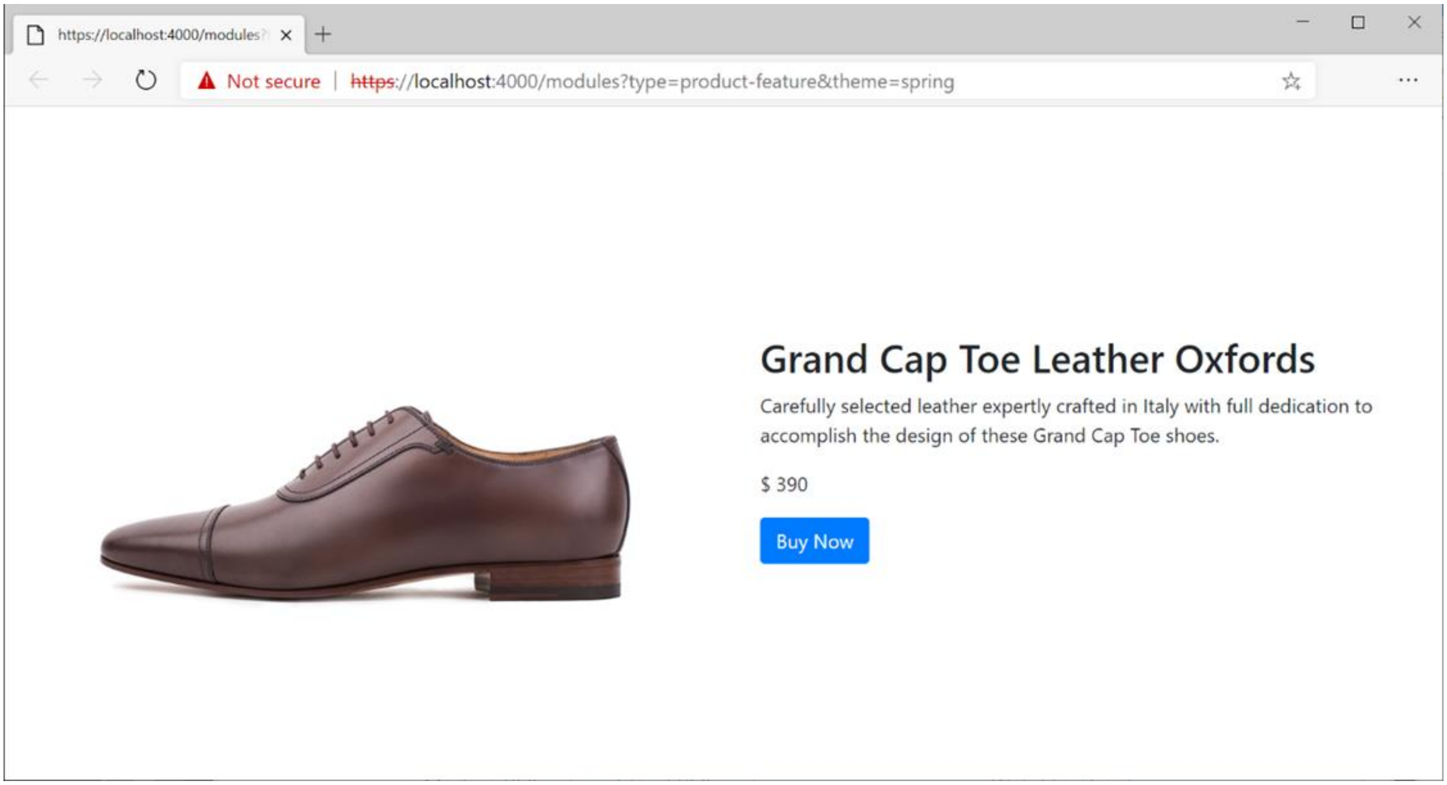
The last thing we need is to wire up the returned data in our React component. In product-feature.tsx make the below changes. We are also adding the product price.

```
/*---------------------------------------------------------------------------------------------
 *  Copyright (c) Microsoft Corporation. All rights reserved.
 *  Licensed under the MIT License. See License.txt in the project root for license information.
 *--------------------------------------------------------------------------------------------*/

import * as React from 'react';

import { IProductFeatureData } from './product-feature.data';
import { imageAlignment, IProductFeatureProps } from './product-feature.props.autogenerated';

export interface IProductFeatureViewProps extends IProductFeatureProps<IProductFeatureData> {
    productName: string;
    productInfo: string;
    productImageUrl: string;
    productPrice: string;
    buttonInfo: string;
    alignment: imageAlignment;
}

/**
 *
 * ProductFeature component
 * @extends {React.PureComponent<IProductFeatureProps<IProductFeatureData>>}
 */
class ProductFeature extends React.PureComponent<IProductFeatureProps<IProductFeatureData>> {
    public render(): JSX.Element | null {
        const {
            config,
            data: { products }
        } = this.props;

        // set default product info values
        let ProductName = config.productTitle ? config.productTitle : 'No product name defined';
        let ProductInfo = config.productDetails ? config.productDetails.toString() : 'No product
details defined';
        let ProductImageUrl = config.productImage ? config.productImage.src : '';
        const ButtonInfo = config.buttonText ? config.buttonText : 'No button text defined';
        let ProductPrice = '129';

        // if product details were returned from Retail server, then we'll use that info instead
of config values
        if (products && products.length) {
            ProductName = products[0].result && products[0].result.Name ? products[0].result.Name
: 'Invalid product data';
```

```
            ProductInfo = products[0].result && products[0].result.Description ? products[0].result.Description : 'Invalid product data';
            ProductImageUrl = products[0].result && products[0].result.PrimaryImageUrl ? products[0].result.PrimaryImageUrl : '';
            ProductPrice = products[0].result && products[0].result.Price ? `$${products[0].result.Price}` : '';
        }

        const ProductFeatureViewProps = {
            ...this.props,
            productName: ProductName,
            productInfo: ProductInfo,
            productImageUrl: ProductImageUrl,
            productPrice: ProductPrice,
            buttonInfo: ButtonInfo,
            alignment: config.imageAlignment
        };

        return this.props.renderView(ProductFeatureViewProps);
    }
}

export default ProductFeature;
```

Adding a new Data Action to the module definition file requires restarting the server, so in your opened command prompt break out (CTRL-C) if it is still running and restart with "**yarn start**".

Refreshing the module in the browser (https://localhost:4000/modules?type=product-feature&theme=spring) should now look similar to below screen shot. We can see the title, details, price and image are now leveraging data directly from the Commerce 365 Retail server. **Note**: this will only render data if the .env file is configured with a valid retail server. See next section to use MOCK data if you do not have an environment available.



https://localhost:4000/modules?
Not secure
https://localhost:4000/modules?type=product-feature&theme=spring

Grand Cap Toe Leather Oxfords

Carefully selected leather expertly crafted in Italy with full dedication to accomplish the design of these Grand Cap Toe shoes.

$ 390

MOCKING CORE DATA ACTIONS

It may be useful to test your module without invoking the actual action using an action mock. You'll need to do this if you haven't configured your Retail server in the ".env " file as we did in section 1.

To create a data action mock, create a new file under "src/module/MODULE_NAME/mocks/" directory for your module, called **myModuleMock.actionmock.json** (note: this file name must have the extension ".actionmock.json" but can start with any name).

Ex:

```
/src/modules/product-feature/mocks/mymodulemock.actionmock.json
```

The data action we are using in this section returns a SimpleProduct object, to mock this you will need to simulate the same object structure that the data action returns. In Visual Studio Code if you open the productFeature.data.ts file we edited above, you can right click on the SimpleProduct type and select "Go to definition" to see the interface, which looks like:

```
/**
 * SimpleProduct entity interface.
 */
export interface SimpleProduct {
    RecordId: number;
    ItemId?: string;
    Name?: string;
    Description?: string;
    ProductTypeValue: number;
    DefaultUnitOfMeasure?: string;
    BasePrice: number;
    Price: number;
    AdjustedPrice: number;
    MasterProductId?: number;
    Components?: ProductComponent[];
    Dimensions?: ProductDimension[];
    Behavior?: ProductBehavior;
    LinkedProducts?: SimpleLinkedProduct[];
    PrimaryImageUrl?: string;
    ItemTypeValue?: number;
    ExtensionProperties?: CommerceProperty[];
}
```

Our module does not use all of the available product fields, so we'll only add the required ones we need to the mocks. Add the following mock data to the new JSON file and save the file.

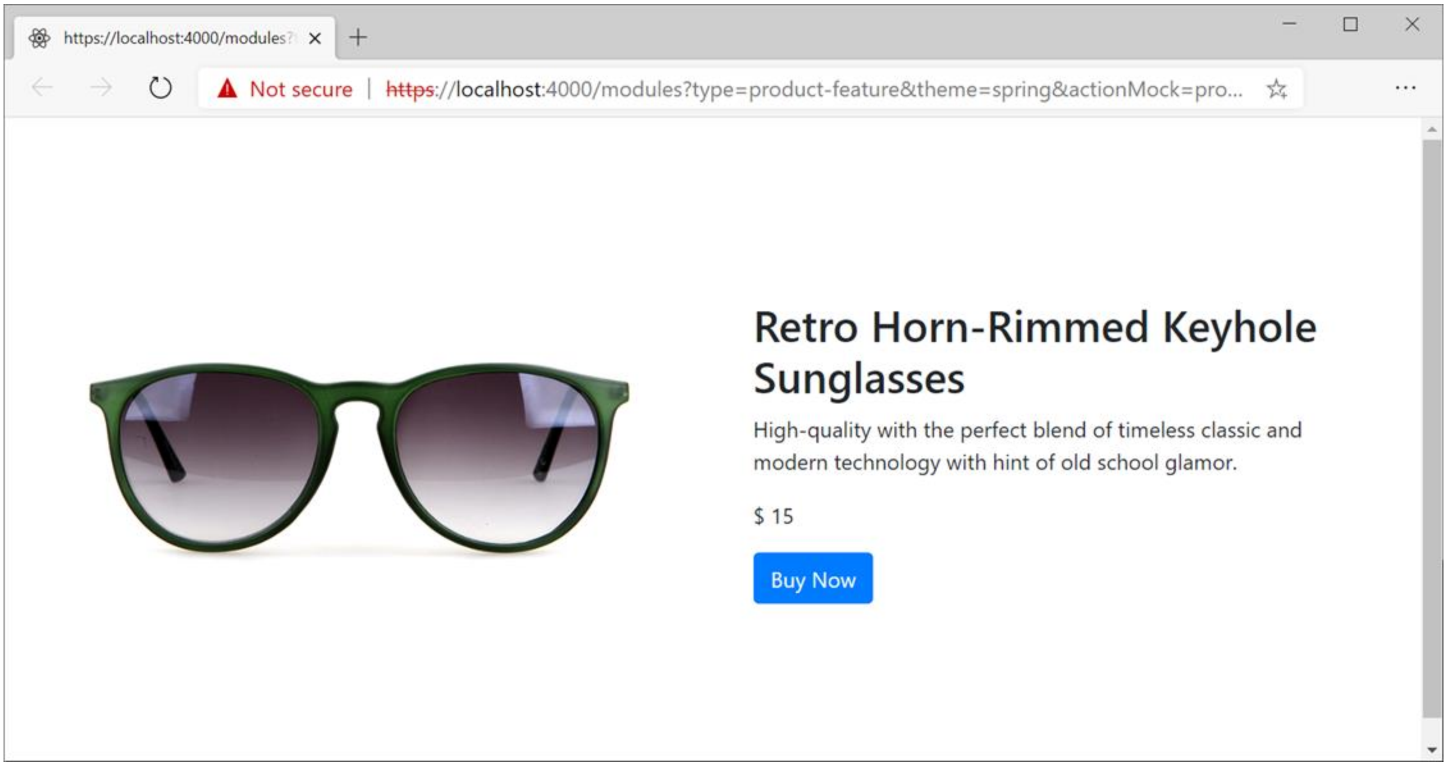
```
[
    {
        "CacheObjectType": "SimpleProduct",
        "CacheKey": "*",
        "Mock": {
            "RecordId": 22565423455,
            "ItemId": "2101",
            "Name": "Retro Horn-Rimmed Keyhole Sunglasses",
            "Description": "High-quality with the perfect blend of timeless classic and
modern technology with hint of old school glamor.",
            "ProductTypeValue": 3,
            "DefaultUnitOfMeasure": "Ea",
            "BasePrice": 15,
            "Price": 15,
            "AdjustedPrice": 14,
            "MasterProductId": null,
            "Components": null,
            "Dimensions": null,
            "Behavior": null,
            "LinkedProducts": null,
            "PrimaryImageUrl": "https://bit.ly/33cMGxr",
            "ExtensionProperties": null
        }
    }
]
```

To use the action mock locally, you just need to include the 'actionMock' query string parameter using the name of the module along with the file name for the mock file created above “MODULE_NAME:MOCK_FILE_NAME” (note you should not add the mock extension .actionmock.json):

Example:

https://localhost:4000/modules?type=product-feature&theme=spring&actionMock=product-feature:mymodulemock

If you refresh the module preview, you should now see data similar to the below image:


https://localhost:4000/modules?
Not secure
https://localhost:4000/modules?type=product-feature&theme=spring&actionMock=pro...

Retro Horn-Rimmed Keyhole Sunglasses

High-quality with the perfect blend of timeless classic and modern technology with hint of old school glamor.

$ 15

MODIFYING STARTER KIT MODULES

You cannot modify an existing Starter Kit module directly; however, you can clone it and change the source as you wish, but the module name must be different. Module views can be extended to avoid cloning modules, but in some cases cloning might be necessary. See section later in the document on module view extensions.

Modifying an existing module from the Starter Kit is as simple as running the CLI command 'yarn msdyn365 clone STARTER_KIT_MODULE_NAME NEW_MODULE_Name'.

Example: '**yarn msdyn365 clone content-block super-content-block**'

Once the command is run you should see the source code for the module under the 'src/modules' directory. At this point you can modify the module as you wish. You may also need to fix up some references in the module.

If you no longer want to keep the modified module (or any other under the 'src') you can just delete it, however if you upload your customization package with missing modules they will be gone and pages that have those modules configured will still render but without that module.

Ensure you test your site well if removing a module.

Note: Module dependencies are not automatically pulled down when you clone a module. Before you build the module, you must run Yarn and fix any missing dependencies.

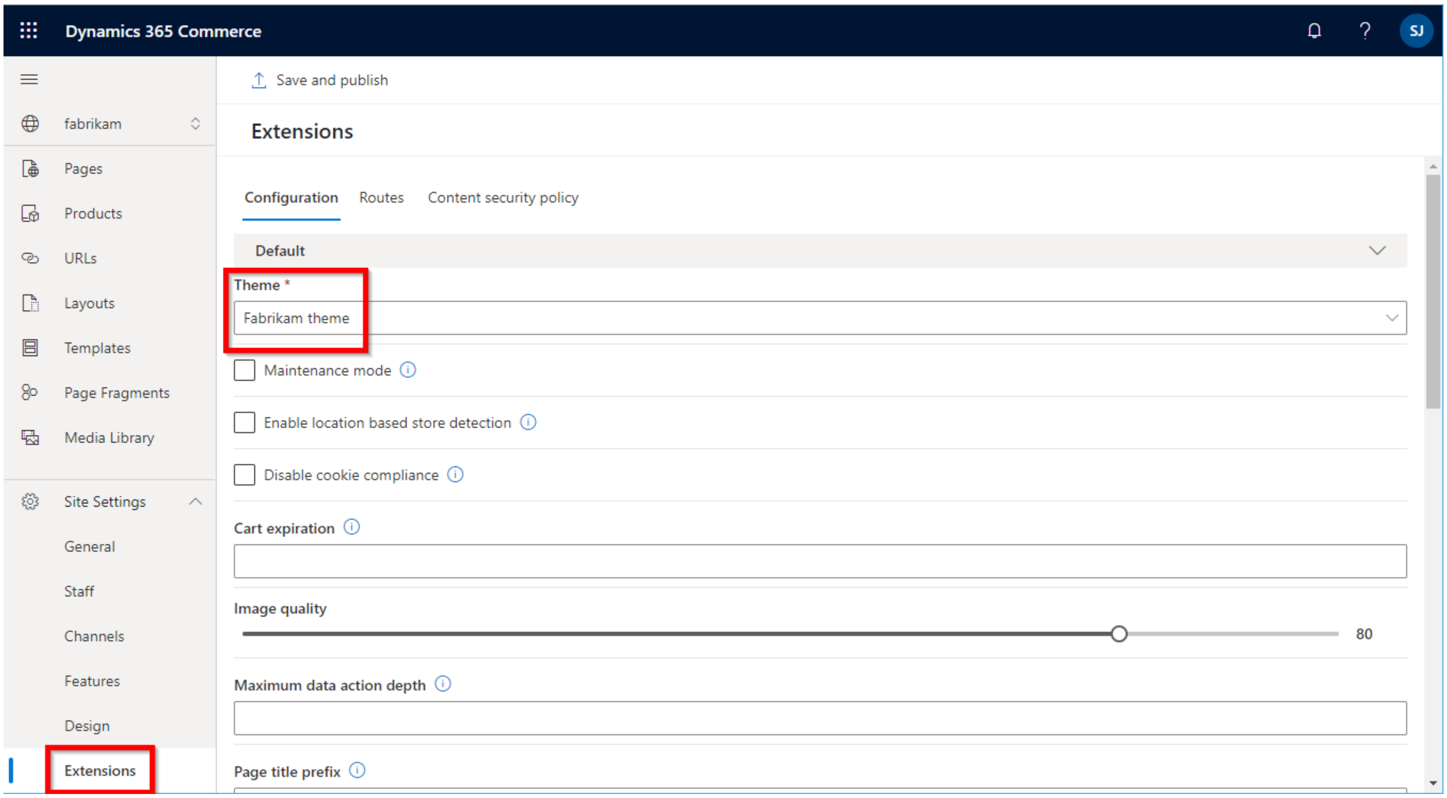
CREATING A CUSTOM THEME

Dynamics 365 e-Commerce allows you to apply a SCSS theme to your whole site.

This section will show you how to create a new theme. Once a theme is created and uploaded to your production site, the site builder tool will allow you to set the theme on your site, within a master/shared template or on a single page. When an e-Commerce page is rendered, the appropriate theme will be applied, giving all the modules on that page a consistent look and feel.

Themes can also contain store starter kit module view and config extensions which allow a theme to change any modules view reducing the need to clone a module to make modifications.

The below screen shot shows how a theme is selected for a site.



Theme *

Fabrikam theme

☐ Maintenance mode

☐ Enable location based store detection

☐ Disable cookie compliance

Cart expiration

Image quality 80

Maximum data action depth

Page title prefix

GENERAL INSTRUCTIONS IN CREATING A CUSTOM THEME

- Create a new theme using the CLI command "yarn msdyn365 add-theme NEW_THEME_NAME" which will create a theme in the "/src/themes/" folder.
- Under the styles directory you will find the SCSS entry point file for the theme which uses name pattern "THEME_NAME.theme.scss".
- Themes are created as special modules and contain a definition file which includes the theme friendly name and description, as well as a template react component.
- There is no limit to the number of ".scss" files your theme may contain.
- Your theme entry point may import other ".scss" files using relative paths.

BEST PRACTICES

- Starter Kit modules are built using Bootstrap 4 classes, so the recommended scss framework to include in any theme is either **Bootstrap 4** or **Bootstrap 4 RTL**.
- Starter Kit modules and components are built using Font Awesome glyphicons, so **FontAwesome** should be included in the scss.

RECOMMENDED STRUCTURE

Import or define the following:

1. Fonts and glyph icons
2. Mixins and functions
   - Bootstrap: dependencies, excluding components and utilities
   - Shared components: dependencies, excluding components and utilities
   - Custom theme mixins and functions
3. Theme variables
   - Custom theme variables
   - Bootstrap: theme variable defaults (fallbacks)
4. Components and modules scss
   - Shared components: components
   - Custom components and modules
5. Utilities
   - Bootstrap, shared component, and custom utilities

THEMING MODULES

Starter Kit modules allow targeting by any theme by defining a class name matching a module name. This class name should be the first class name applied to the outermost element rendered by the React component. Developers may allow for more granular themability options by providing additional class names on elements or features of a module that may be targetable by custom themes.

See Bootstrap 4 theming guide for further details on Bootstrap: https://getbootstrap.com/docs/4.0/getting-started/theming/

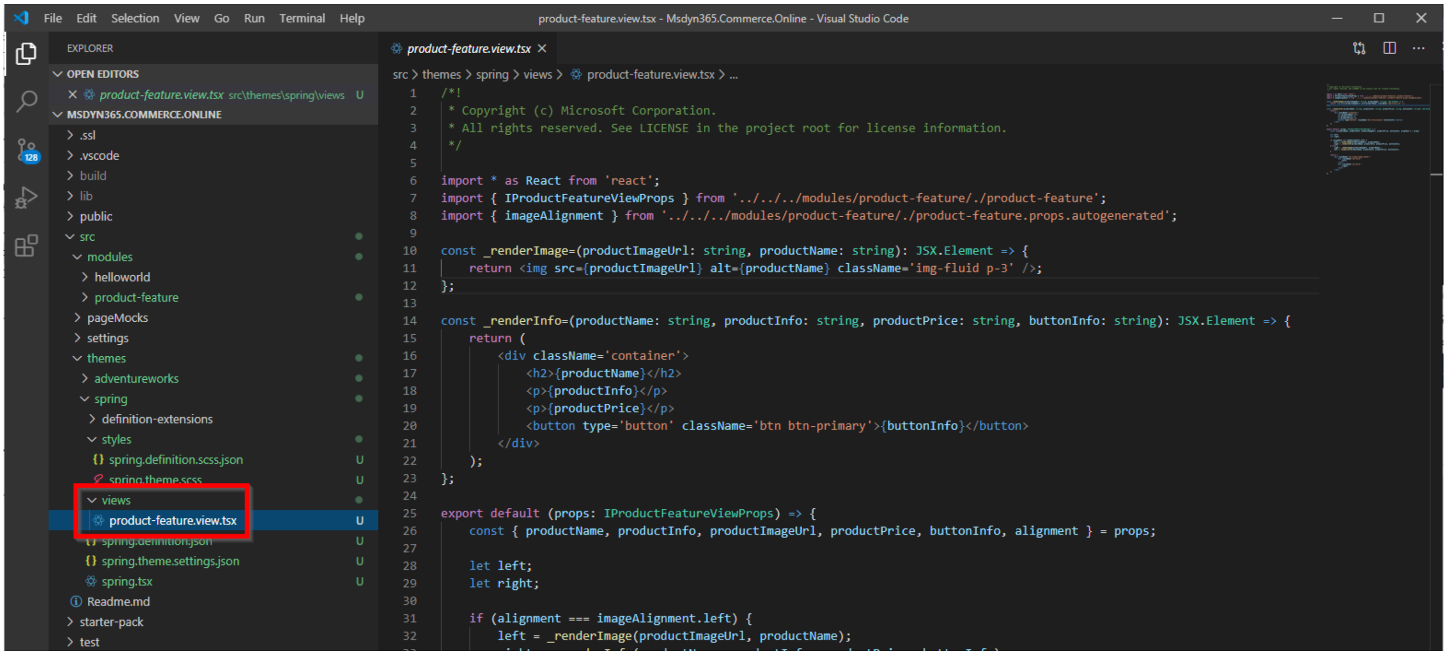
MODULE VIEW EXTENSIONS

Themes can group together multiple module 'view' changes without having to clone the original module. This can include your own module or starter kit modules.

Earlier we created a theme called "Spring", we can create a view extension for our module (product-feature) we created earlier in our 'spring' theme execute the following command:

```
yarn msdyn365 add-view-extension spring product-feature
```

Notice a new file was added to the **".../src/themes/spring/views**" directory with the following naming pattern "MODULE_NAME.view.tsx". Example you should see ".../src/themes/spring/views/product-feature.view.tsx". This is a copy of the module view file and notice the import path was auto corrected.

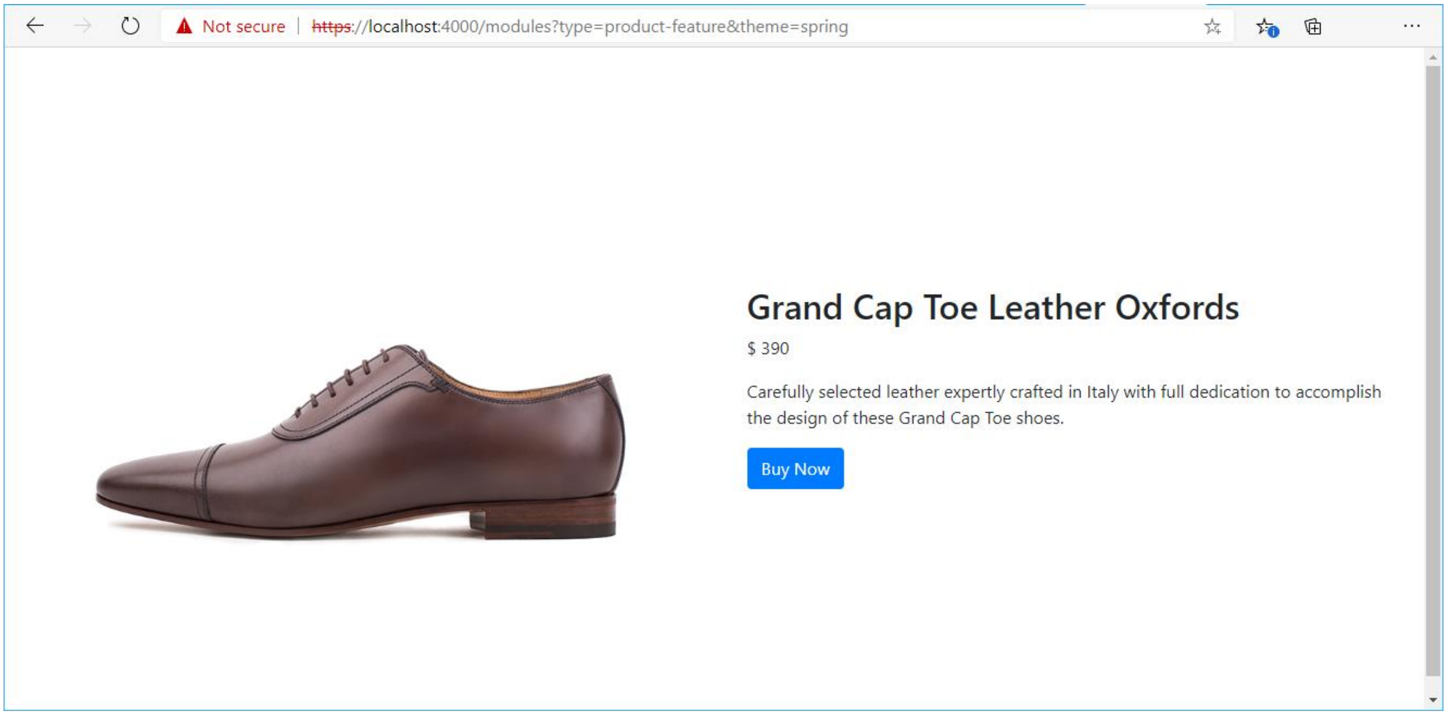


```
/*!
 * Copyright (c) Microsoft Corporation.
 * All rights reserved. See LICENSE in the project root for license information.
 */

import * as React from 'react';
import { IProductFeatureViewProps } from '../../../modules/product-feature/./product-feature';
import { imageAlignment } from '../../../modules/product-feature/./product-feature.props.autogenerated';

const _renderImage=(productImageUrl: string, productName: string): JSX.Element => {
    return <img src={productImageUrl} alt={productName} className='img-fluid p-3' />;
};

const _renderInfo=(productName: string, productInfo: string, productPrice: string, buttonInfo: string): JSX.Element => {
    return (
        <div className='container'>
            <h2>{productName}</h2>
            <p>{productInfo}</p>
            <p>{productPrice}</p>
            <button type='button' className='btn btn-primary'>{buttonInfo}</button>
        </div>
    );
};

export default (props: IProductFeatureViewProps) => {
    const { productName, productInfo, productImageUrl, productPrice, buttonInfo, alignment } = props;

    let left;
    let right;

    if (alignment === imageAlignment.left) {
        left = _renderImage(productImageUrl, productName);
```

Make a minor change to the HTML below to re-order the productInfo and productPrice. between the highlighted lines below. and save the file.

```
/*!
 * Copyright (c) Microsoft Corporation.
 * All rights reserved. See LICENSE in the project root for license information.
 */

import * as React from 'react';
import { IProductFeatureViewProps } from '../../../modules/product-feature/./product-feature';
import { imageAlignment } from '../../../modules/product-feature/./product-
feature.props.autogenerated';

const _renderImage=(productImageUrl: string, productName: string): JSX.Element => {
    return <img src={productImageUrl} alt={productName} className='img-fluid p-3' />;
};

const _renderInfo=(productName: string, productInfo: string, productPrice: string, buttonInfo: string): JSX.Element => {
    return (
        <div className='container'>
            <h2>{productName}</h2>
            <p>{productPrice}</p>
            <p>{productInfo}</p>
            <button type='button' className='btn btn-primary'>{buttonInfo}</button>
        </div>
    );
};

export default (props: IProductFeatureViewProps) => {
    const { productName, productInfo, productImageUrl, productPrice, buttonInfo, alignment } = props;

    let left;
    let right;

    if (alignment === imageAlignment.left) {
        left = _renderImage(productImageUrl, productName);
        right = _renderInfo(productName, productInfo, productPrice, buttonInfo);
    } else {
        right = _renderImage(productImageUrl, productName);
        left = _renderInfo(productName, productInfo, productPrice, buttonInfo);
    }

    return (
        <div className='row align-items-center'>
            <div className='col-sm-6'>
                {left}
            </div>
            <div className='col-sm-6'>
                {right}
            </div>
        </div>
    );
};
```

Run “yarn start” from the command prompt window and navigate to https://localhost:4000/modules?type=product-feature&theme=spring and you should see the changes to the module. Note the price now shows up above the product details in the below image.



Not secure
~~https~~://localhost:4000/modules?type=product-feature&theme=spring

Grand Cap Toe Leather Oxfords

$ 390

Carefully selected leather expertly crafted in Italy with full dedication to accomplish the design of these Grand Cap Toe shoes.

MODULE DEFINITION EXTENSIONS

To create a definition extension, create a “definition-extensions” directory under the theme directory if it does not exist and create a new file with the pattern “MODULE_NAME.definition.ext.json”. For example, create /src/themes/spring/definition-extensions/product-feature.definition.ext.json file. Paste in the following to add a new string config for a subtitle.

```
{
    "$type": "definitionExtension",
    "config": {
        "productSubTitle": {
            "type": "string",
            "friendlyName": "Product Sub-Title",
            "description": "Product placement sub title"
        }
    }
}
```

When you run “yarn start” a new *.autogenerated.ts file will be added to the same directory. You’ll need to do this to proceed to the next step.

You can now modify the view extension (\themes\spring\views\product-feature.view.tsx) to use the new property. Below shows an example with the changes highlighted to consume the new config via the props.config object

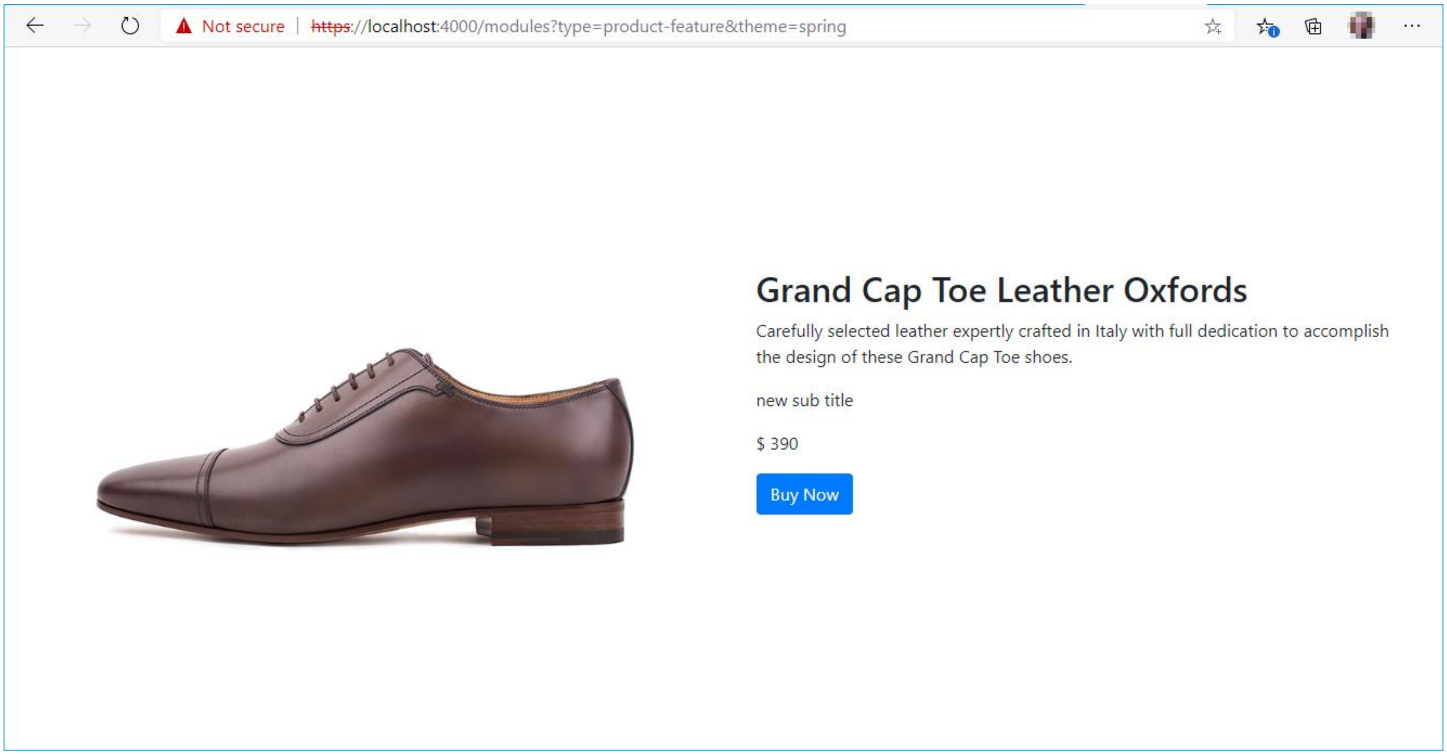
```
/*!
 * Copyright (c) Microsoft Corporation.
 * All rights reserved. See LICENSE in the project root for license information.
 */

import * as React from 'react';
import { IProductFeatureViewProps } from '../../../modules/product-feature/./product-feature';
import { imageAlignment } from '../../../modules/product-feature/./product-
feature.props.autogenerated';
import { IProductFeatureProps } from '../definition-extensions/product-
feature.ext.props.autogenerated';

const _renderImage=(productImageUrl: string, productName: string): JSX.Element => {
    return <img src={productImageUrl} alt={productName} className='img-fluid p-3' />;
};

const _renderInfo=(productName: string, productInfo: string, productPrice: string, buttonInfo:
string, subTitle?: string): JSX.Element => {
    return (
        <div className='container'>
            <h2>{productName}</h2>
            <p>{productInfo}</p>
            <p>{subTitle}</p>
            <p>{productPrice}</p>
            <button type='button' className='btn btn-primary'>{buttonInfo}</button>
        </div>
    );
};
```

```
export default (props: IProductFeatureViewProps & IProductFeatureProps<{}>) => {
    const { productName, productInfo, productImageUrl, productPrice, buttonInfo, alignment } =
props;
    const { productSubTitle } = props.config;

    let left;
    let right;

    if (alignment === imageAlignment.left) {
        left = _renderImage(productImageUrl, productName);
        right = _renderInfo(productName, productInfo, productPrice, buttonInfo, productSubTitle);
    } else {
        right = _renderImage(productImageUrl, productName);
        left = _renderInfo(productName, productInfo, productPrice, buttonInfo, productSubTitle);
    }

    return (
        <div className='row align-items-center'>
            <div className='col-sm-6'>
                {left}
            </div>
            <div className='col-sm-6'>
                {right}
            </div>
        </div>
    );
};
```

Next you can add additional mock data to the original module under the \src\modules\product-feature\mocks\product-feature.json file:

```
{
  "id": "R1Module1",
  "config": {
    "imageAlignment": "left",
    "productTitle": "Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses",
    "productDetails": "High-
quality and pioneered with the perfect blend of timeless classic and modern technology with hint o
f old school glamor.",
    "productImage": {
      "src": "https://bit.ly/33cMGxr",
      "altText": "Retro Horn Rimmed Keyhole Nose Bridge Round Sunglasses"
    },
    "buttonText": "Buy Now",
    "productIds": "68719498121",
    "productSubTitle": "new sub title"
  },
  "data": {
    "actionResponse": {
      "text": "Sample Action Response"
    }
  },
  "typeName": "product-feature"
}
```

Running “yarn start” and previewing the module with the theme should now render the extra “productSubTitle” config.



~~https~~://localhost:4000/modules?type=product-feature&theme=spring

Grand Cap Toe Leather Oxfords

Carefully selected leather expertly crafted in Italy with full dedication to accomplish the design of these Grand Cap Toe shoes.

new sub title

$ 390

BUILDING A THEME BASED OFF STARTER KIT “FABRIKAM” THEME

A great place to start with theme development is to leverage the “**Fabrikam**” theme code which was downloaded along with the SDK to your development environment. This section will show how to make a copy of this theme and then changes can be made to the CSS which can change the look and feel of the starter kit modules and default CSS.

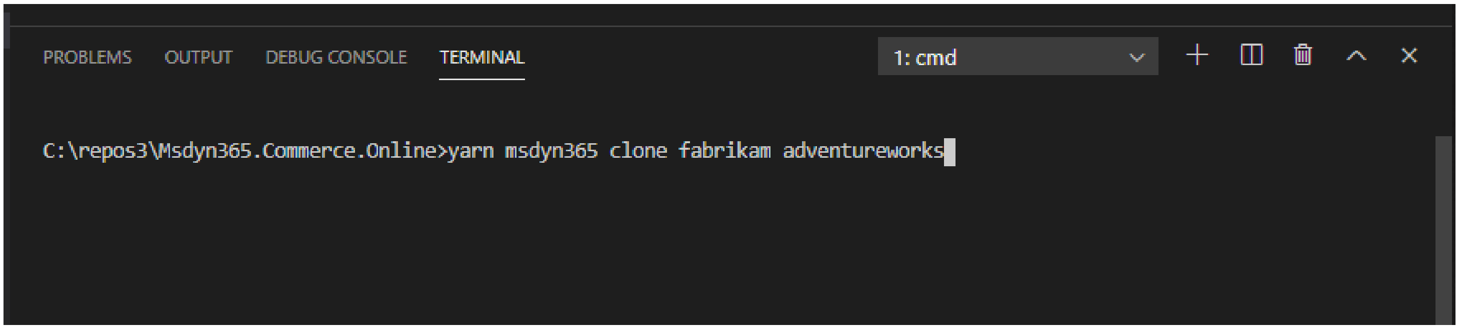
As shown in a previous section, the online SDK provides a CLI command to **clone** a module, and we’ve mentioned themes are a type of module, so we’ll use this to make a copy of the Fabrikam theme that is included in the starter kit.

If interested the source code for the starter kit Fabrikam theme module can be found here:

“c:\repos\Msdyn365.Commerce.Online\node_modules\@msdyn365-commerce-modules\fabrikam-design-kit\src\modules\fabrikam” directory.

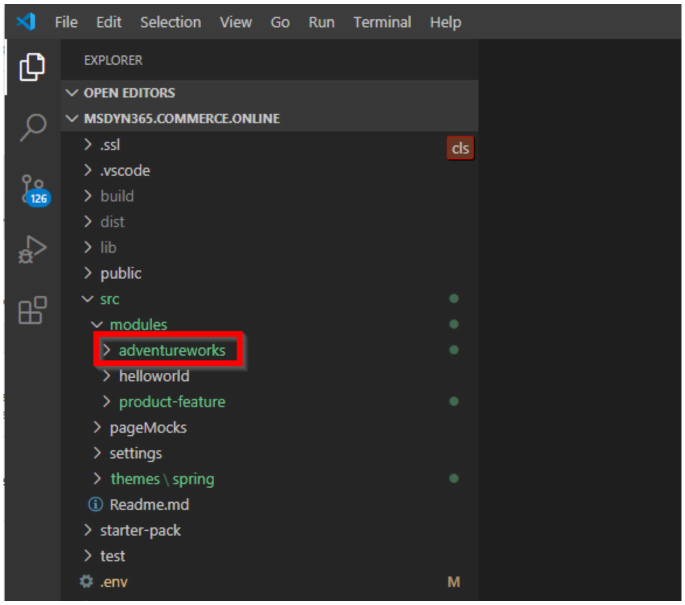
From within the Visual Studio Code terminal window stop the Node server if it is running by hitting Ctrl-C and then type the following command to clone the “Fabrikam” theme module into a new one called “adventureworks”.

```
yarn msdyn365 clone fabrikam adventureworks
```

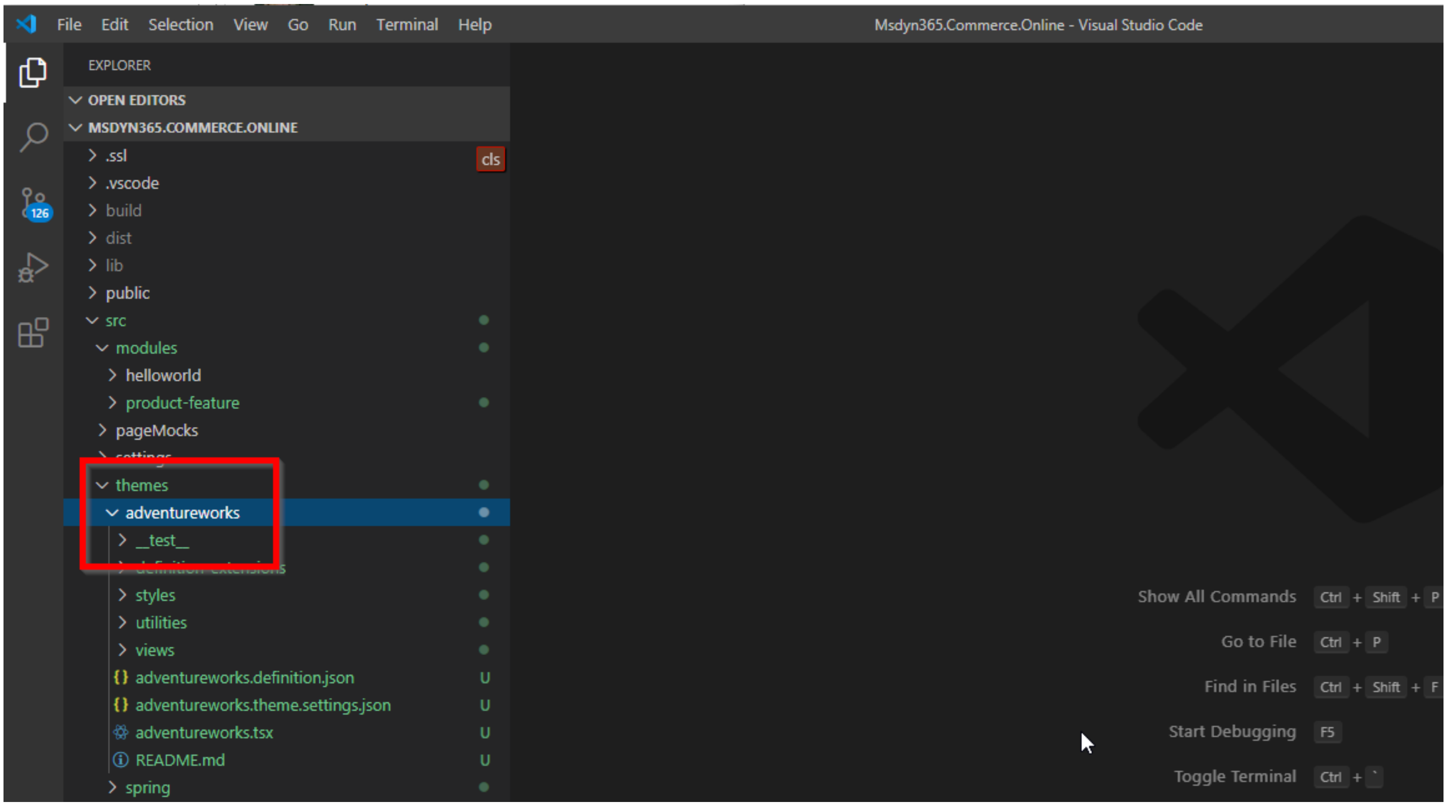


```
C:\repos3\Msdyn365.Commerce.Online>yarn msdyn365 clone fabrikam adventureworks
```

You should now see the theme module under the “src/modules” directory as shown below.



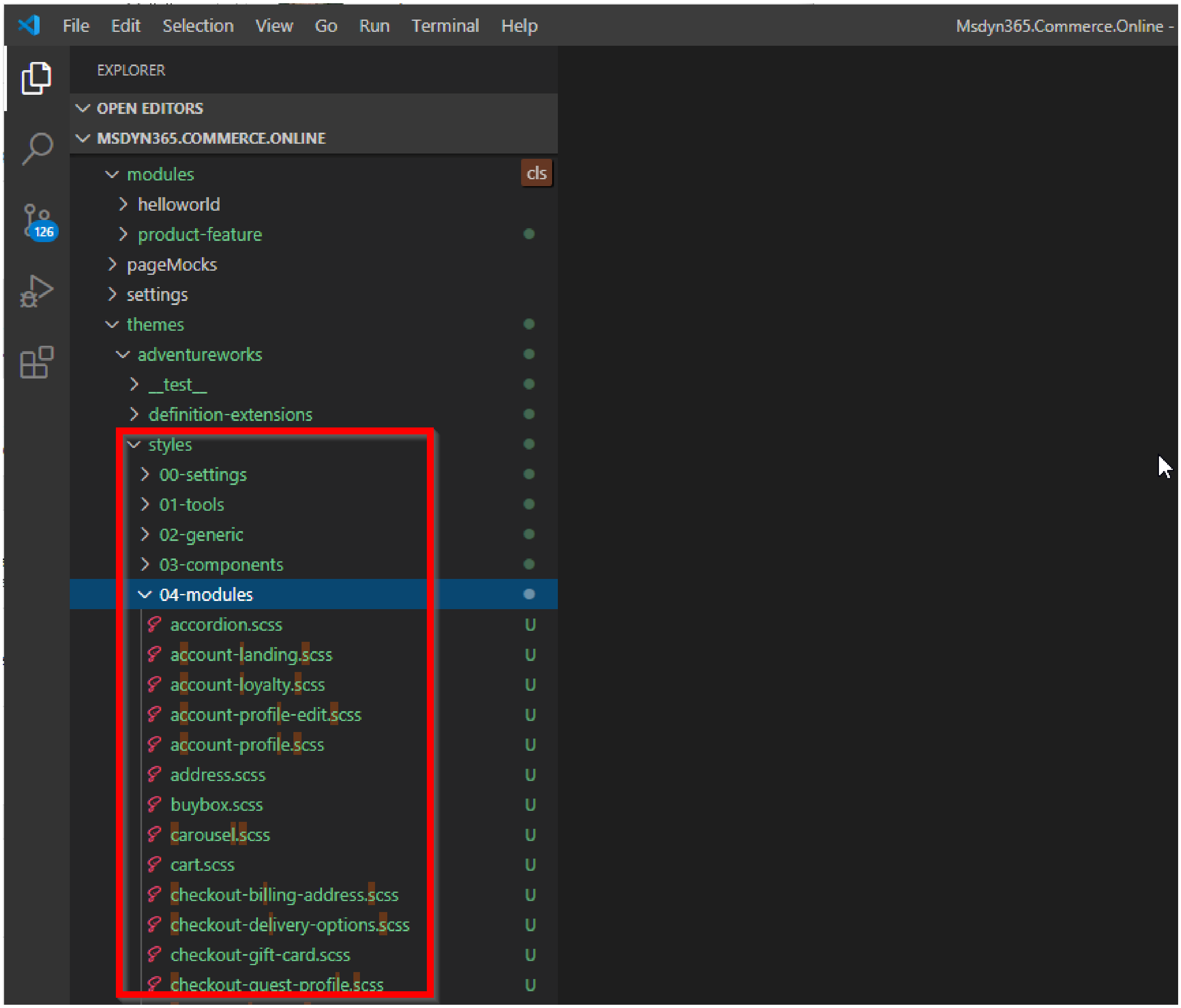
We have to make one more change, theme modules are treated a bit differently than regular modules and need to be under the themes directory. You should be able to drag the “adventureworks” theme directory and drop it under the themes directory as shown below.



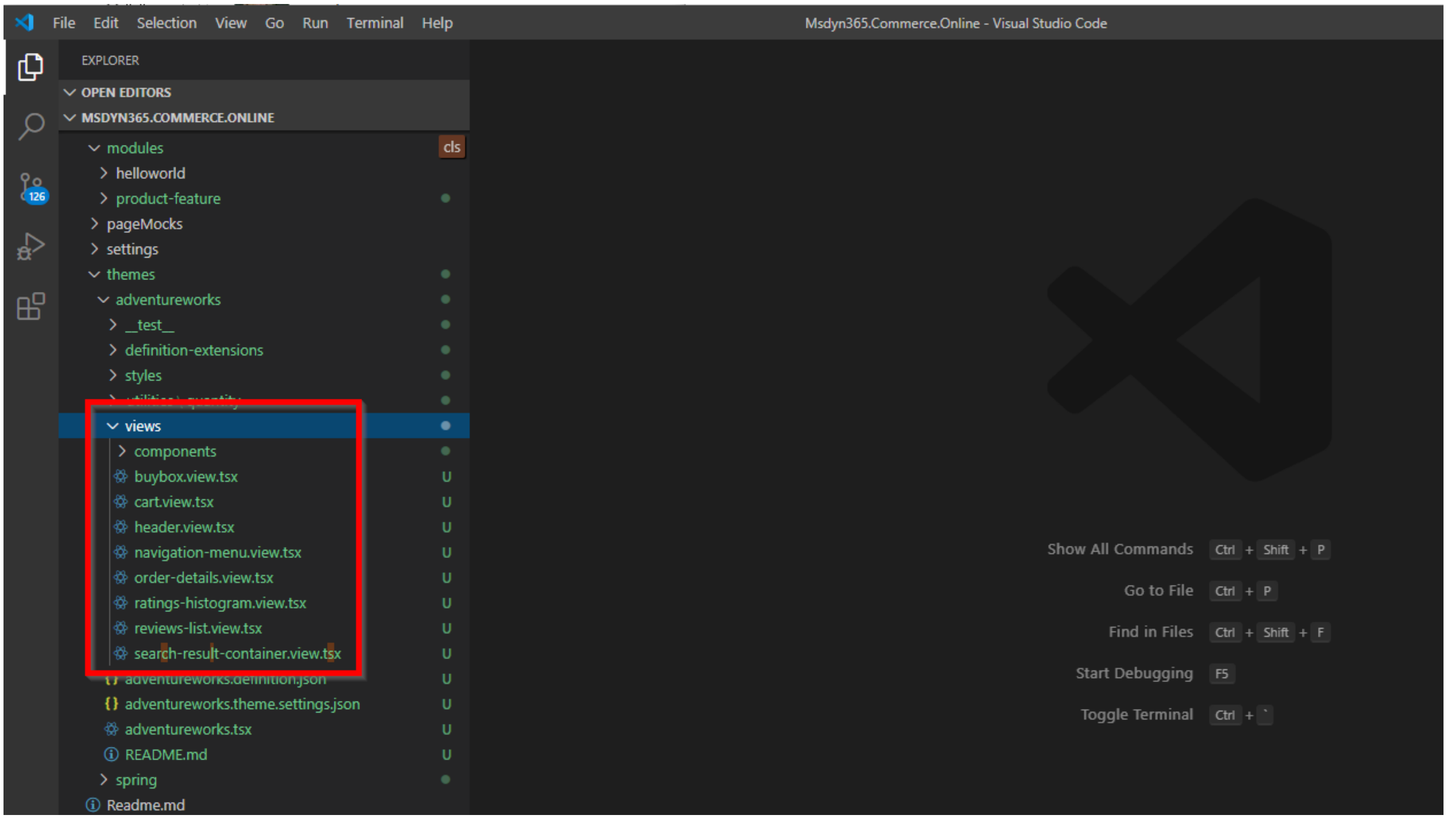
Next, open the new adventureworks.definition.json file and modify the friendly name and description, this is the name that will show up in the site builder tool when configurint themes.

```
{
    "$type": "themeModule",
    "friendlyName": "Adventure Works",
    "name": "adventureworks",
    "description": "This is the Adventure Works theme."
}
```

We now have a new theme based on the Fabrikam theme. If you look inside the theme files, you'll notice the Fabrikam theme contains SCSS files for each module under the "styles" directory.



Themes can change a modules view (layout / html), this can include starter kit modules as well as custom modules. The “views” directory contains module view overrides. Notice the Fabrikam theme has several starter kit module view overides already.



You can also extend a starter kit modules configuration properties inside of a theme, these are done with module defintion extension files added to the “definition-extensions” directory, you’ll see a few have already been changed in the Fabrikam theme.

You can modify these files as needed and under the styles directory you can modify any of the default SCSS including module SCSS..

TESTING YOUR CUSTOM THEME

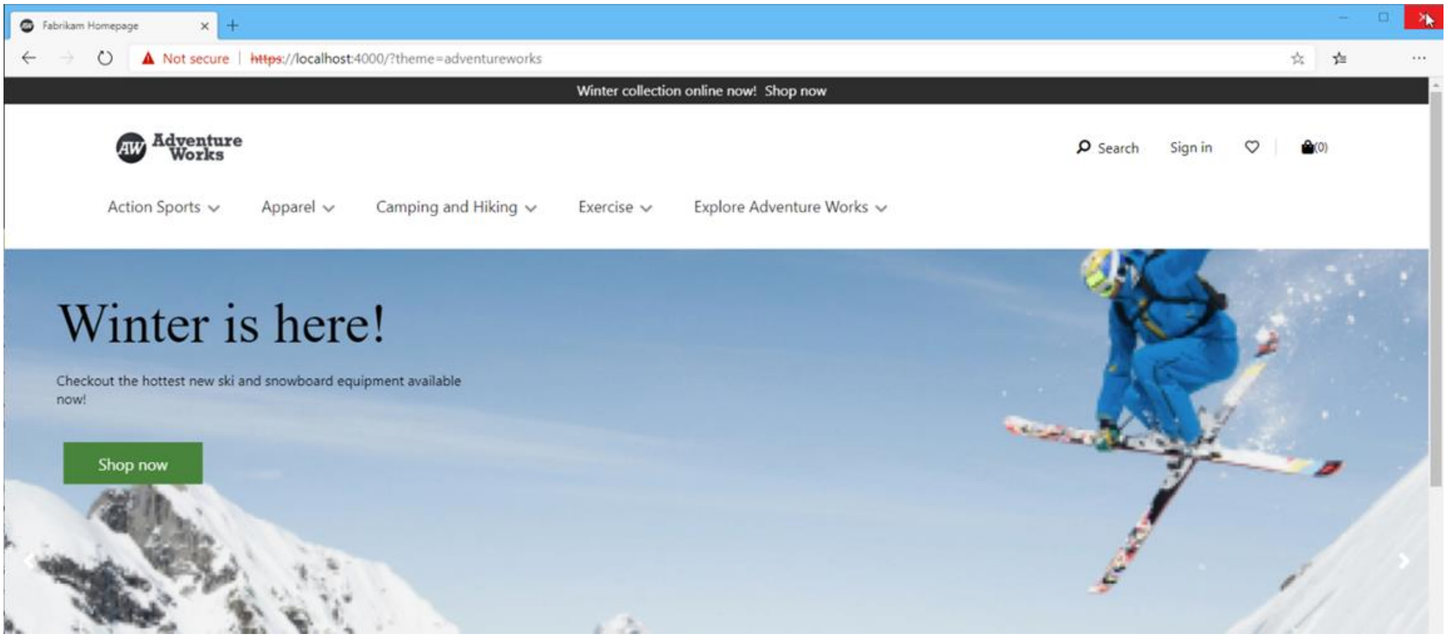
If you setup the .env file to point to your host site, you can also see your theme rendering against your live site. To do this first run "yarn start" from the terminal window.

You can now navigate in a Browser to see our live site rendered in the local development environment.

Try https://localhost:4000 , this should render your live site using the default Fabrikam theme.

To render using our new theme use this URL: https://localhost:4000/?theme=adventureworks

For now the theme should look the same as Fabrikam.

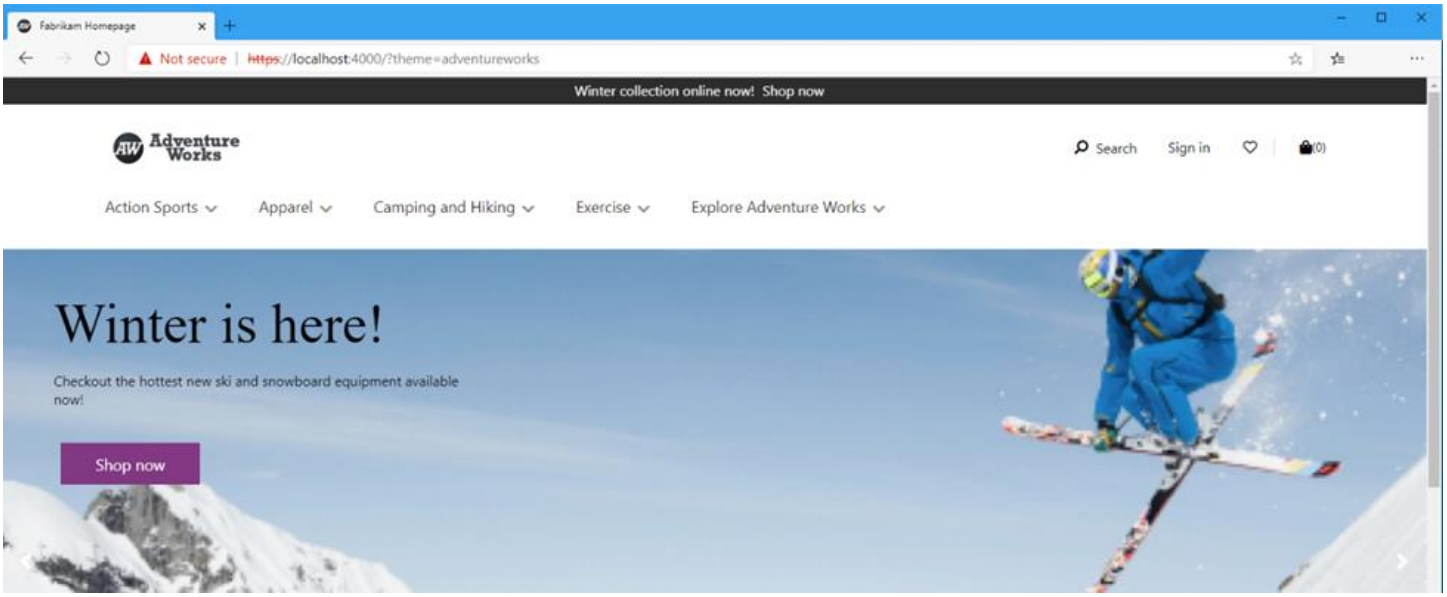

Fabrikam Homepage
Not secure | https://localhost:4000/?theme=adventureworks
Winter collection online now! Shop now
Adventure Works
Search
Sign in
(0)
Action Sports
Apparel
Camping and Hiking
Exercise
Explore Adventure Works
Winter is here!
Checkout the hottest new ski and snowboard equipment available now!
Shop now

If you now make theme changes you can see them on your online site.

For example, change the primary color in the /src/themes/adventureworks/styles/00-settings/colors and save the file.

```
…
$msv-primary: rgb(129, 58, 131);
$msv-primary-hover: rgb(63, 44, 99);
…
```

Notice in the below image how the buttons on the site change to match the color changes.



Winter is here!

Checkout the hottest new ski and snowboard equipment available now!

MODULE DEFINITION EXTENSIONS

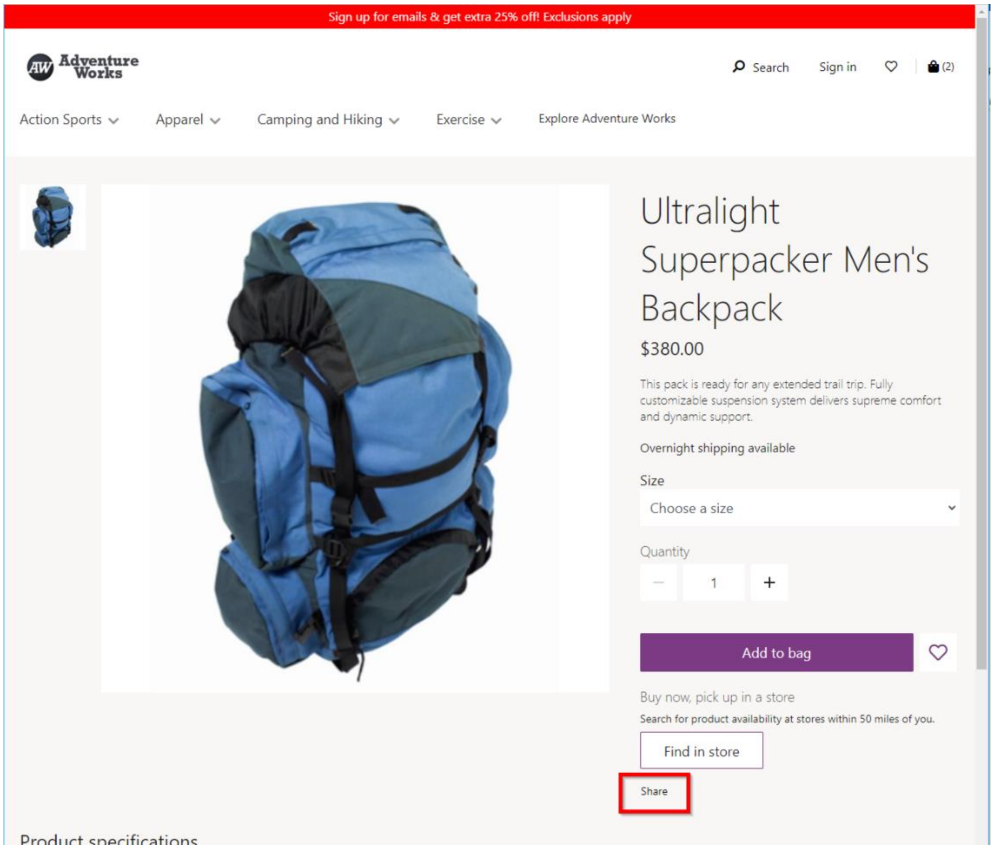
To show another module definition extension in action, we'll make a small change to the buybox starter kit module.

The below example shows how to modify the buybox.view.tsx to add a share link to send the product as an email. Notice it already exists under the "/src/themes/adventureworks/views" directory, however if it wasn't or you want to extend a different module's view for your theme, a CLI command is available (example: yarn msdyn365 add-module-view-extension adventureworks buybox).

Open the file and make the following highlighted changes.

```
/*---------------------------------------------------------------------------------------------
 *  Copyright (c) Microsoft Corporation. All rights reserved.
 *  Licensed under the MIT License. See License.txt in the project root for license information.
 *--------------------------------------------------------------------------------------------*/
import * as React from 'react';

import {
    IBuyboxAddToCartViewProps,
    IBuyboxAddToWishlistViewProps,
    IBuyboxCallbacks,
    IBuyboxData,
    IBuyboxFindInStoreViewProps,
    IBuyboxProductConfigureDropdownViewProps,
    IBuyboxProductConfigureViewProps,
    IBuyboxProductQuantityViewProps,
    IBuyboxProps,
    IBuyboxState,
    IBuyboxViewProps
 } from '@msdyn365-commerce-modules/buybox';

import { Module, Node } from '@msdyn365-commerce-modules/utilities';

import { IBuyboxProps as IBuyboxExtentionProps, IBuyboxResources as IBuyboxExtentionResources } fr
om '../definition-extensions/buybox.ext.props.autogenerated';
import { Quantity } from '../utilities/quantity';

const BuyboxView: React.FC<IBuyboxViewProps & IBuyboxExtentionProps<IBuyboxData>> = props => {
    const { ModuleProps, MediaGalleryContainerProps, ProductInfoContainerProps, addToCart, addToWi
shlist, configure, description, findInStore, quantity, price, title, rating, callbacks, state, res
ources } = props;

    return (
        <Module {...ModuleProps}>
            <Node {...MediaGalleryContainerProps}>
                {props.mediaGallery}
            </Node>
            <Node {...ProductInfoContainerProps}>
                {title}
                {price}
                {description}
                { _renderTextBlock(props.slots.textBlocks)}
                {rating}
                {configure && _renderConfigure(configure)}
                {quantity && _renderQuantity(quantity, callbacks, props, state, resources)}
                { _renderCartAndWishlistDiv(addToCart, addToWishlist)}
                {findInStore && _renderFindInStore(findInStore)}
                {_renderShareLink(props.data?.product?.result?.Name, props.context.request.url.requestUrl.href)}
            </Node>
        </Module>
    );
};

const _renderShareLink = (productName?: String, productLInk?: String): JSX.Element | undefined => {
    if (!productName || !URL) {
        return undefined;
    }

    const hrefContent=`mailto:?body=Check out this product ${productName}!\n ${productLInk}\n\n &subject=I want to recommend this product from Adventure Works!`;

    return (
        <a className='a-link-normal email' title='Share via e-mail' href={hrefContent}>Share</a>
    );
};

const _renderTextBlock = (textBlocks: React.ReactNode[]): JSX.Element | undefined => {
    if (!textBlocks || textBlocks.length === 0) {
        return undefined;
    }

    return (
        <>
            {textBlocks[0]}
        </>
    );
};

const _renderCartAndWishlistDiv = (addToCart?: IBuyboxAddToCartViewProps, addToWishlist?: IBuyboxAddToWishlistViewProps): JSX.Element | undefined => {
    if (!addToCart && !addToWishlist) {
        return undefined;
    }

    return (
        <div className='product-add-to-cart'>
            {addToCart && addToCart.errorBlock}
            {addToWishlist && addToWishlist.errorBlock}
            <div className='buttons'>
                {addToCart && addToCart.button}
                {addToWishlist && addToWishlist.button}
            </div>
        </div>
    );
};

const _renderConfigure = (configure: IBuyboxProductConfigureViewProps): JSX.Element => {
    const { ContainerProps, dropdowns } = configure;
    return (
        <Node {...ContainerProps}>
            {dropdowns.map(_renderConfigureDropdown)}
        </Node>
    );
};

const _renderConfigureDropdown = (dropdown: IBuyboxProductConfigureDropdownViewProps): JSX.Element
 => {
    const { ContainerProps, LabelContainerProps, heading, errors, select } = dropdown;

    return (
        <Node {...ContainerProps}>
            <Node {...LabelContainerProps}>
                {heading}
                {errors}
            </Node>
            {select}
        </Node>
    );
};

const _renderFindInStore = (findInStore: IBuyboxFindInStoreViewProps): JSX.Element => {
    const { ContainerProps, storeSelector, heading, description, errors, button } = findInStore;

    return (
        <Node {...ContainerProps}>
            {storeSelector}
            {heading}
            {description}
            {errors}
            {button}
        </Node>
    );
};

const _renderQuantity = (quantityComponent: IBuyboxProductQuantityViewProps, callbacks: IBuyboxCal
lbacks, props: IBuyboxProps<IBuyboxData>, state: IBuyboxState, extentionResources: IBuyboxExtentio
nResources): JSX.Element => {
    const { ContainerProps, LabelContainerProps, heading, errors } = quantityComponent;

    const {
        resources,
        context: {
            app: {
                config: {
                    maxQuantityForCartLineItem
                }
            }
        }
    } = props;

    const
    {
        quantity,
    } = state;

    const onChange = (newValue: number) => {
        if (callbacks.updateQuantity) {
            callbacks.updateQuantity(newValue);
        }
    };

    return (
        <Node {...ContainerProps}>
            <Node {...LabelContainerProps}>
                {heading}
                {errors}
            </Node>

            <Quantity
                id='ms-buybox__product-quantity-input'
                max={maxQuantityForCartLineItem || 10}
                currentCount={quantity}
                onChange={onChange}
                inputQuantityAriaLabel={resources.inputQuantityAriaLabel}
                decrementButtonAriaLabel={extentionResources.decrementButtonAriaLabel}
                incrementButtonAriaLabel={extentionResources.incrementButtonAriaLabel}
            />
        </Node>
    );
};

export default BuyboxView;
```

The below output shows what the module change will look like.



Ultralight Superpacker Men's Backpack

$380.00

This pack is ready for any extended trail trip. Fully customizable suspension system delivers supreme comfort and dynamic support.

Overnight shipping available

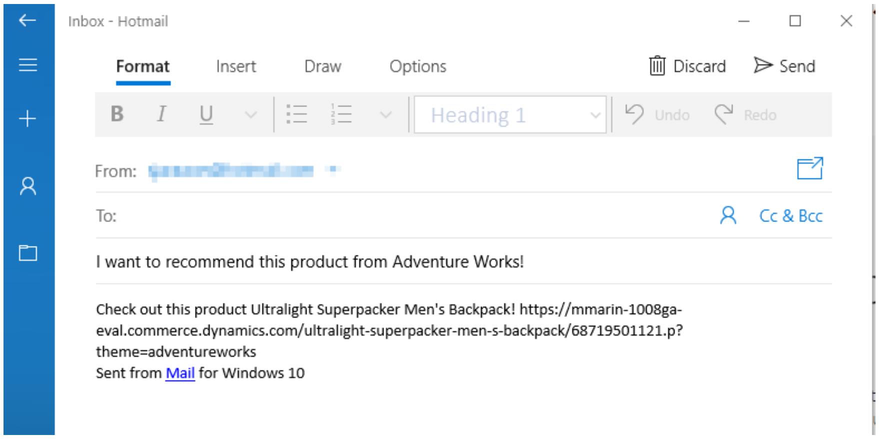
Size: Choose a size

Quantity: 1

Buy now, pick up in a store

Search for product availability at stores within 50 miles of you.

Selecting the link will bring up the default mail app with a new e-mail including the product name and link.



From: [redacted]

To:

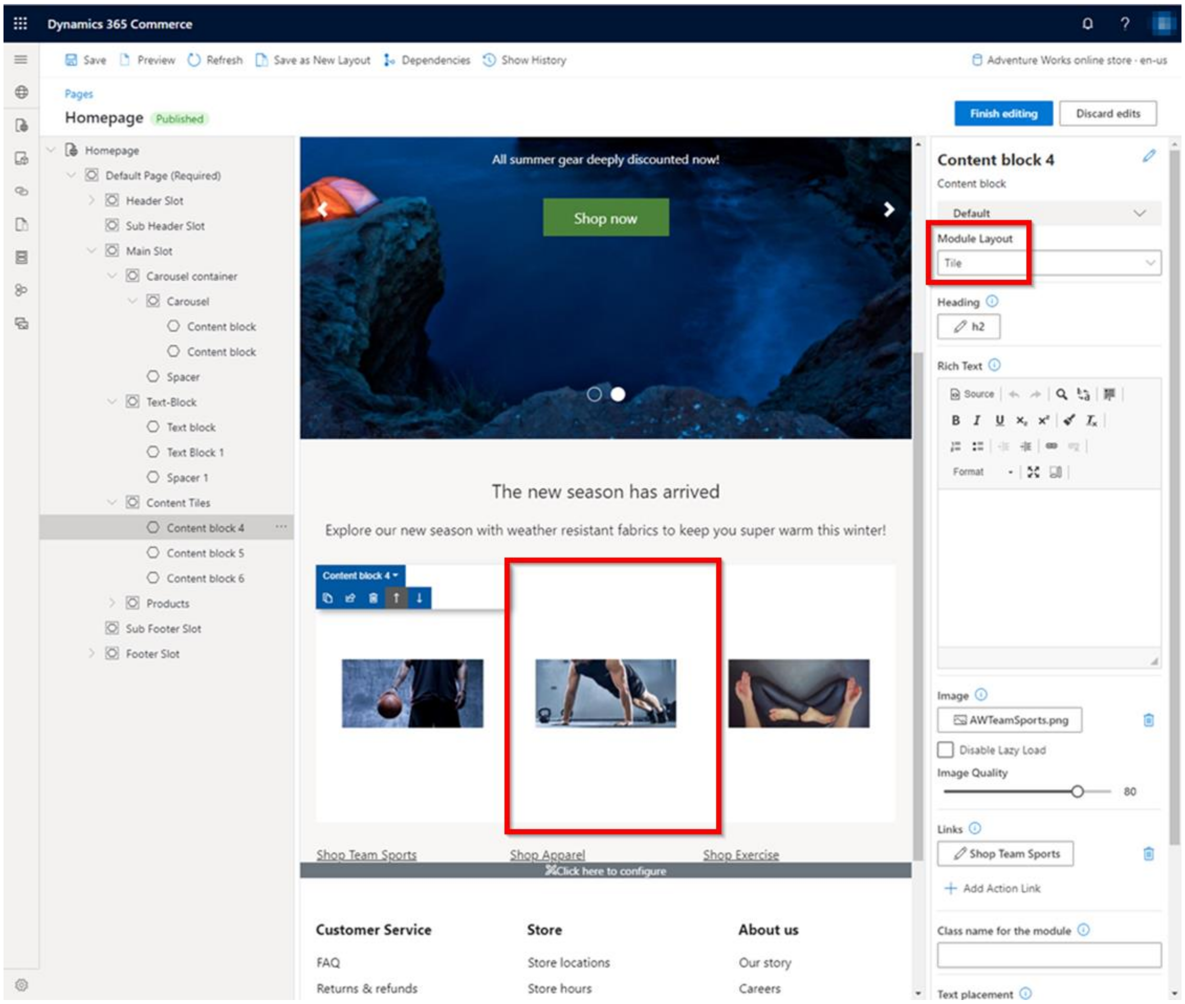
I want to recommend this product from Adventure Works!

Check out this product Ultralight Superpacker Men's Backpack! https://mmarin-1008ga-eval.commerce.dynamics.com/ultralight-superpacker-men-s-backpack/68719501121.p?theme=adventureworks
Sent from Mail for Windows 10

THEME SETTINGS

Theme designers can specify various layouts for each module and can control specific layout options for images that the module renders. The layouts are specified in the THEME_NAME.theme.settings.json file located in the root of the theme folder. For example, you should see a src/themes/adventureworks/adventureworks.theme.settings.json file.

To see this in practice, earlier in Unit 2 we swapped out some marketing images on our homepage. Notice however, since the images were a different aspect ratio, they are rendering additional white space around the image (as shown in image below).



All summer gear deeply discounted now!

The new season has arrived

Explore our new season with weather resistant fabrics to keep you super warm this winter!

Content block 4

Content block

Default

Module Layout: Tile

Heading: h2

Rich Text

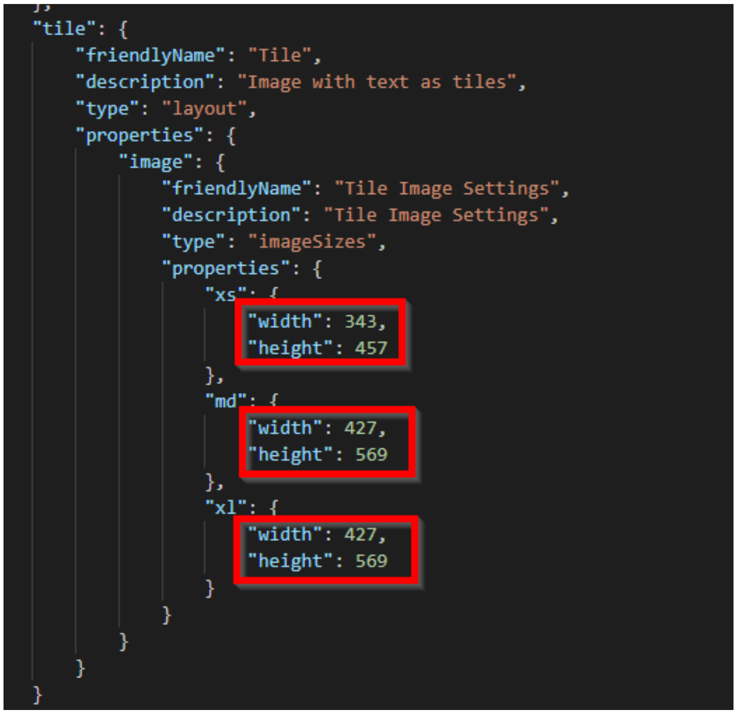
Image: AWTeamSports.png

☐ Disable Lazy Load

Image Quality: 80

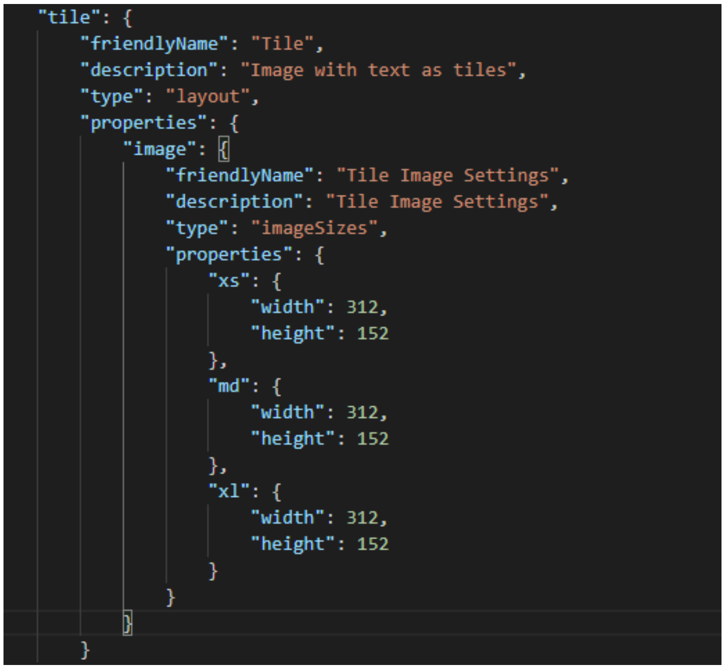
Links: Shop Team Sports

This is due to the auto image resizer using the image settings provided in the Fabrikam theme. If you open adventureworks.theme.settings.json file you'll find the following definition for the "content-block" module "Tile" layout:



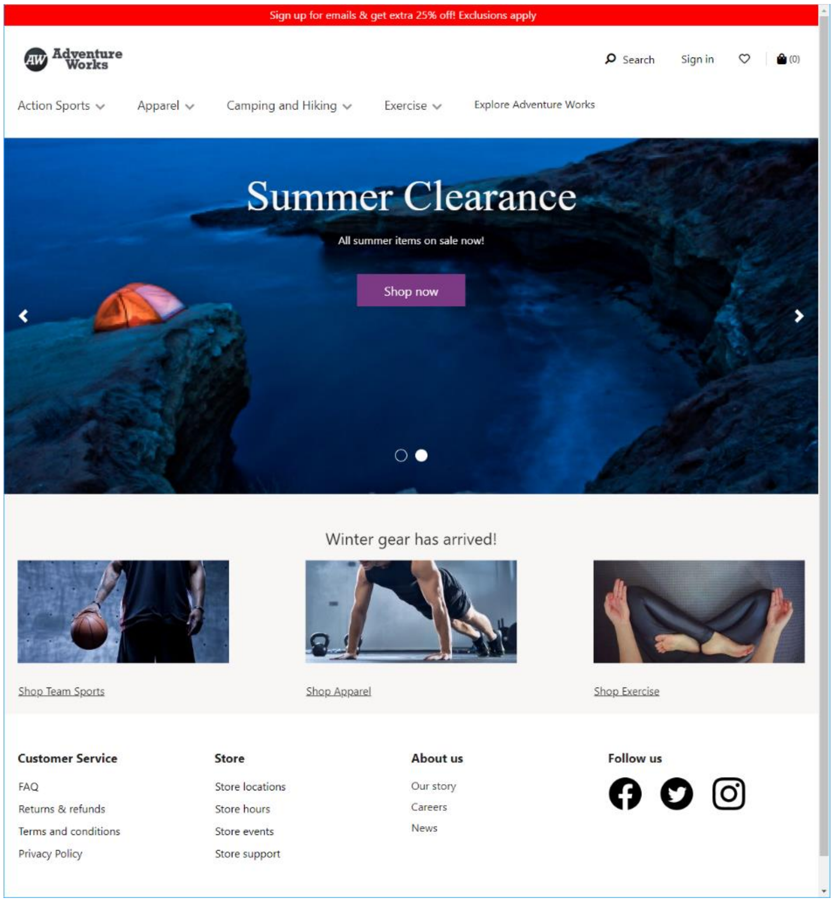
```
"tile": {
    "friendlyName": "Tile",
    "description": "Image with text as tiles",
    "type": "layout",
    "properties": {
        "image": {
            "friendlyName": "Tile Image Settings",
            "description": "Tile Image Settings",
            "type": "imageSizes",
            "properties": {
                "xs": {
                    "width": 343,
                    "height": 457
                },
                "md": {
                    "width": 427,
                    "height": 569
                },
                "xl": {
                    "width": 427,
                    "height": 569
                }
            }
        }
    }
}
```

Notice the image width and heights provided above are for a larger image with a different aspect ratio. The ones we provided in the assets zip file were 312x152. So we can fix up the sizes here as shown below.



```
"tile": {
    "friendlyName": "Tile",
    "description": "Image with text as tiles",
    "type": "layout",
    "properties": {
        "image": {
            "friendlyName": "Tile Image Settings",
            "description": "Tile Image Settings",
            "type": "imageSizes",
            "properties": {
                "xs": {
                    "width": 312,
                    "height": 152
                },
                "md": {
                    "width": 312,
                    "height": 152
                },
                "xl": {
                    "width": 312,
                    "height": 152
                }
            }
        }
    }
```

You will need to deploy this file to see the proper rendering, debugging on the local dev environment will not leverage this file. See the next section on deploying the extensions. Once deployed the home page will now look like the below.



Sign up for emails & get extra 25% off! Exclusions apply

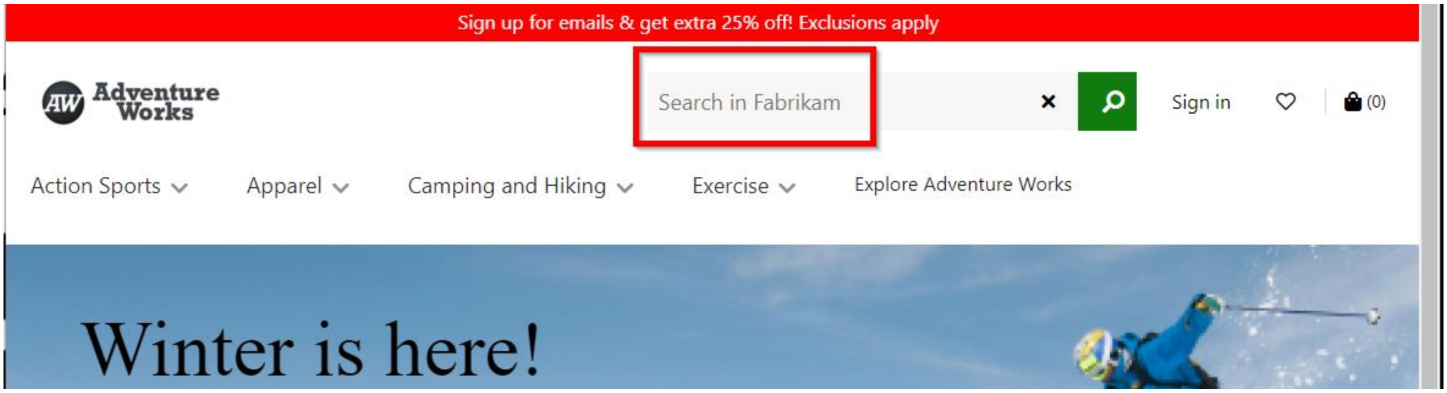
Summer Clearance

All summer items on sale now!

Winter gear has arrived!

CHANGING THE STARTER KIT MODULE RESOURCE STRINGS

You may find situations where you want to change a resource string that was included as part of the starter kit. For example the placeholder search text has been hard coded to “Search in Fabrikam”.



Module resource are stored in a global.json file and can be found under the C:\repos\Msdyn365.Commerce.Online\node_modules\@msdyn365-commerce-modules\resources\src\resources\modules directory.

If you open this file and search for searchplaceholderText you should find the resource as shown below.

```
  "searchplaceholderText": {
        "value": "Search in Fabrikam",
        "_value.comment": "Text for search place holder"
    },
```

We can easily override this setting by creating a new directory under our src directory “resources” and then “modules” under that and adding a new file global.json. For example, “C:\repos\Msdyn365.Commerce.Online\src\resources\modules\global.json”.

We can now override any values, see below example to change the text to “Search”.

```
{
    "@msdyn365-commerce-modules.searchplaceholderText": {
      "value": "Search"
      "comment": "Text for search place holder",
    }
}
```

BUILDING AND DEPLOYING E-COMMERCE EXTENSION PACKAGE

To deploy your new/modified customizations of modules, data actions and/or themes onto a test or production server, a customization package must be created, then uploaded via LCS.

In this section, we will cover how to create a configuration package and upload the package via LCS and deploy to release servers.

CREATE A CONFIGURATION PACKAGE

To package your local customizations, open a Command Prompt and navigate to the root folder of the installed SDK and use the '**yarn msdyn365 pack**" command, below is an example:

```
C:\repos\Project.Rushmore>yarn msdyn365 pack
```

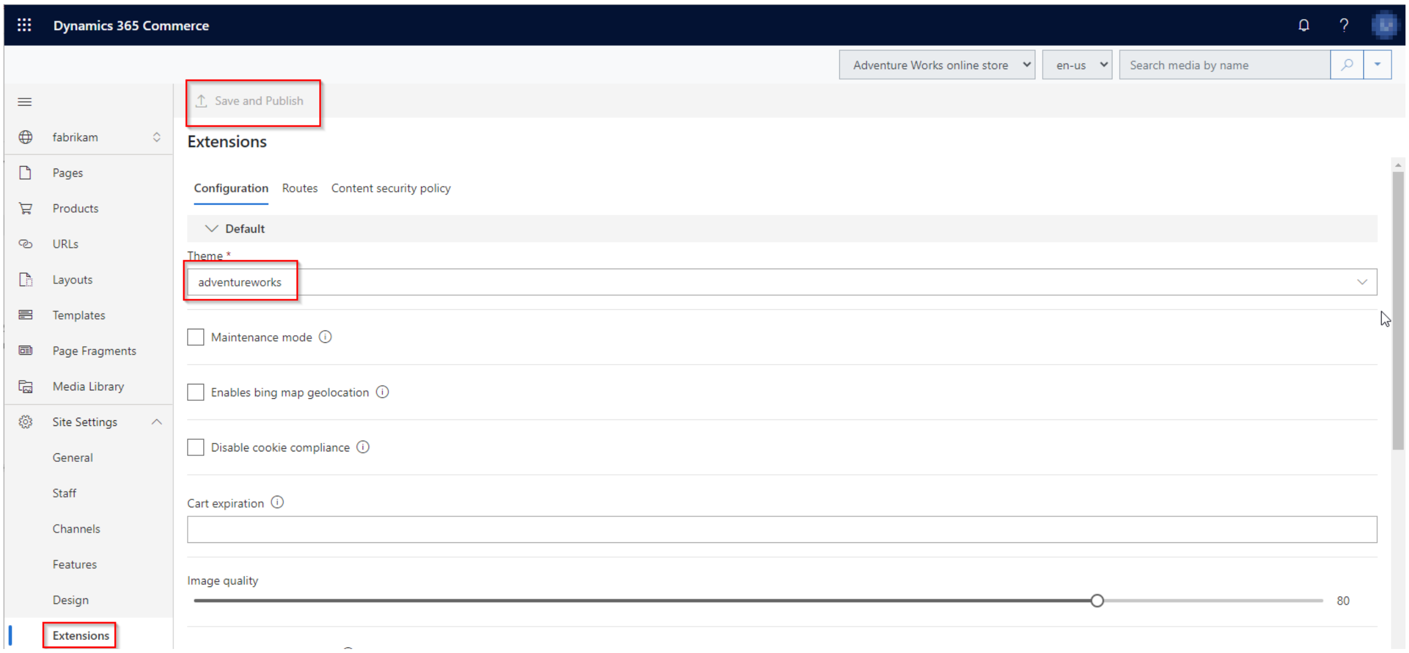
This could take a couple of minutes to complete and there should now be a new package file (.zip) in the same directory. This file will then be uploaded to LCS.

It will then create a zip file with a name similar to "Msdyn365.Commerce.Online-1.0.0.zip" under the SDK root directory (Example: c:\repose\Msdyn365.Commerce.Online).

This file is now ready to upload and deploy via LCS, follow the Package configurations and deploy them to an online environment doc to do this.

Once deployed follow the below steps to set the new theme for the site in the Site Builder tool.

- Select "Extensions" page under the "Site Settings"
- Select "adventureworks" under the Theme drop down
- Select "Save and Publish"
- **NOTE**: there is currently a bug that the "Save and publish" won't be enabled due to a missing property in Content security policy, enable the "Enable Report Only Mode" option.



Dynamics 365 Commerce

Adventure Works online store | en-us | Search media by name

Save and Publish

fabrikam

Pages

Products

URLs

Layouts

Templates

Page Fragments

Media Library

Site Settings

General

Staff

Channels

Features

Design

Extensions

Extensions

Configuration | Routes | Content security policy

Default

Theme *

adventureworks

□ Maintenance mode

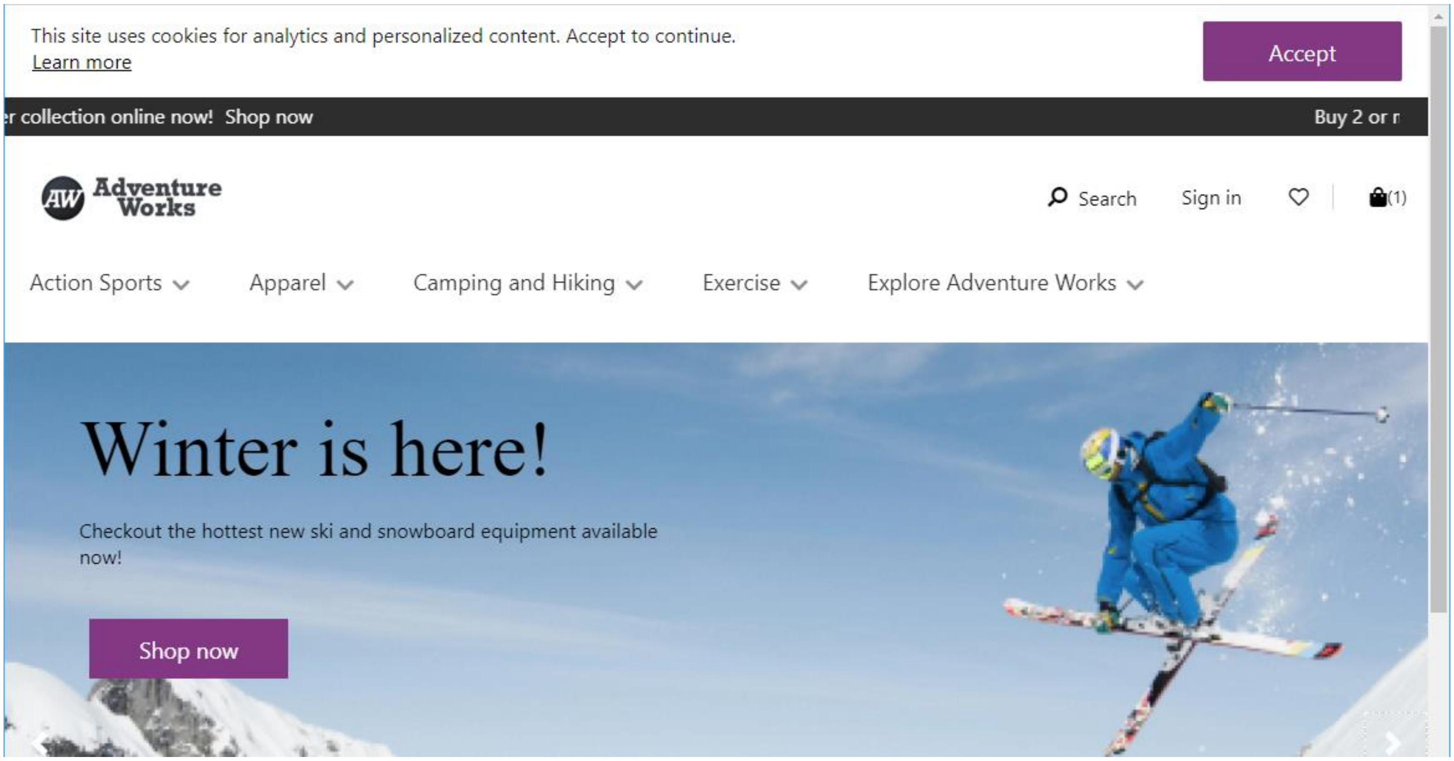
□ Enables bing map geolocation

□ Disable cookie compliance

Cart expiration

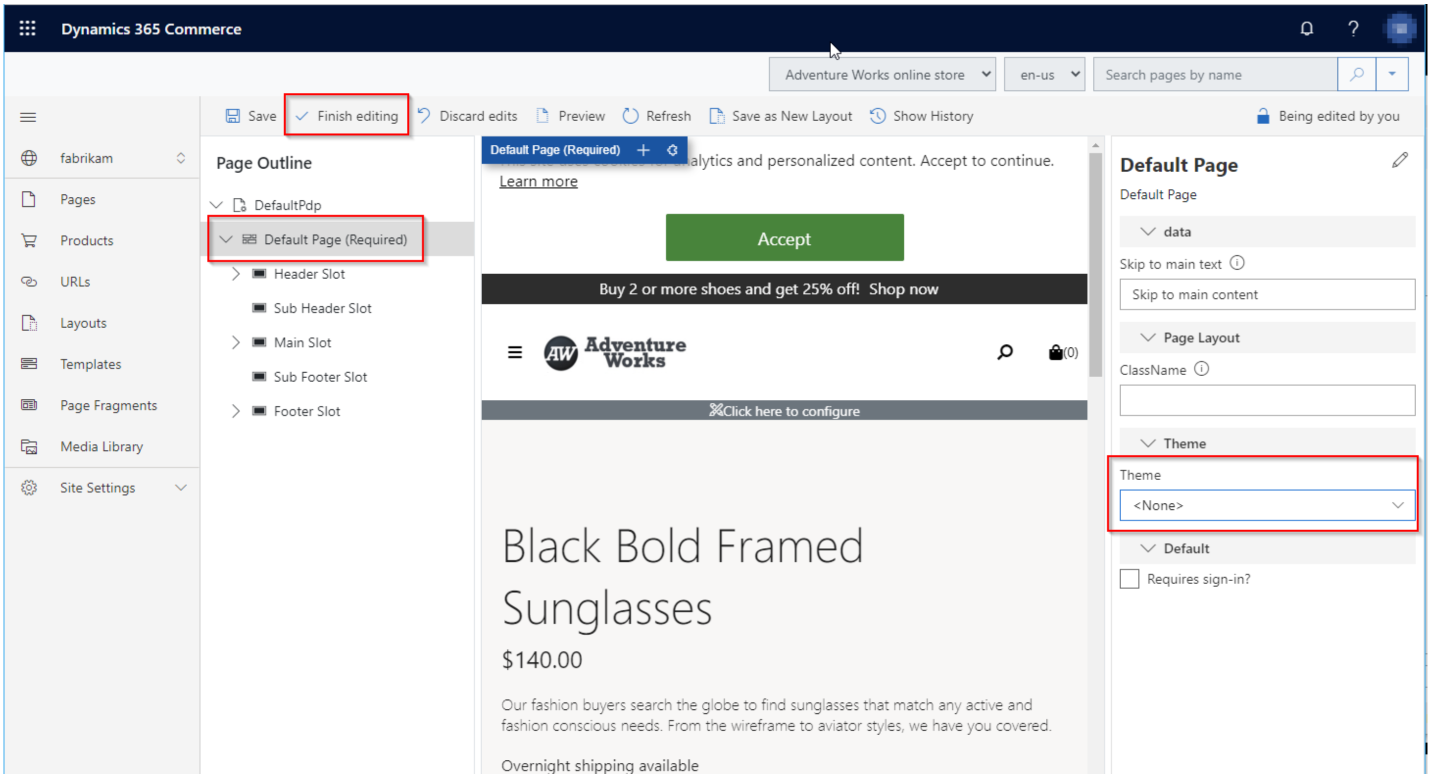
Image quality 80

If you now refresh the site you should see the new theme applied, notice the changed button colors:



This site uses cookies for analytics and personalized content. Accept to continue.
Learn more
Accept
r collection online now! Shop now
Buy 2 or r
Adventure Works
Search
Sign in
(1)
Action Sports
Apparel
Camping and Hiking
Exercise
Explore Adventure Works
Winter is here!
Checkout the hottest new ski and snowboard equipment available now!
Shop now

We also need to disable the PDP page theme selection it has been set to “Fabrikam” by default. Select the “Pages” tab and “DefaultPDP” page, click “Edit”. Now select the “Default Page” node in the tree and change the “Theme” to “<None>”. Click “Finish editing” and “Publish”.



...alytics and personalized content. Accept to continue.

Buy 2 or more shoes and get 25% off! Shop now

Black Bold Framed Sunglasses

$140.00

Our fashion buyers search the globe to find sunglasses that match any active and fashion conscious needs. From the wireframe to aviator styles, we have you covered.

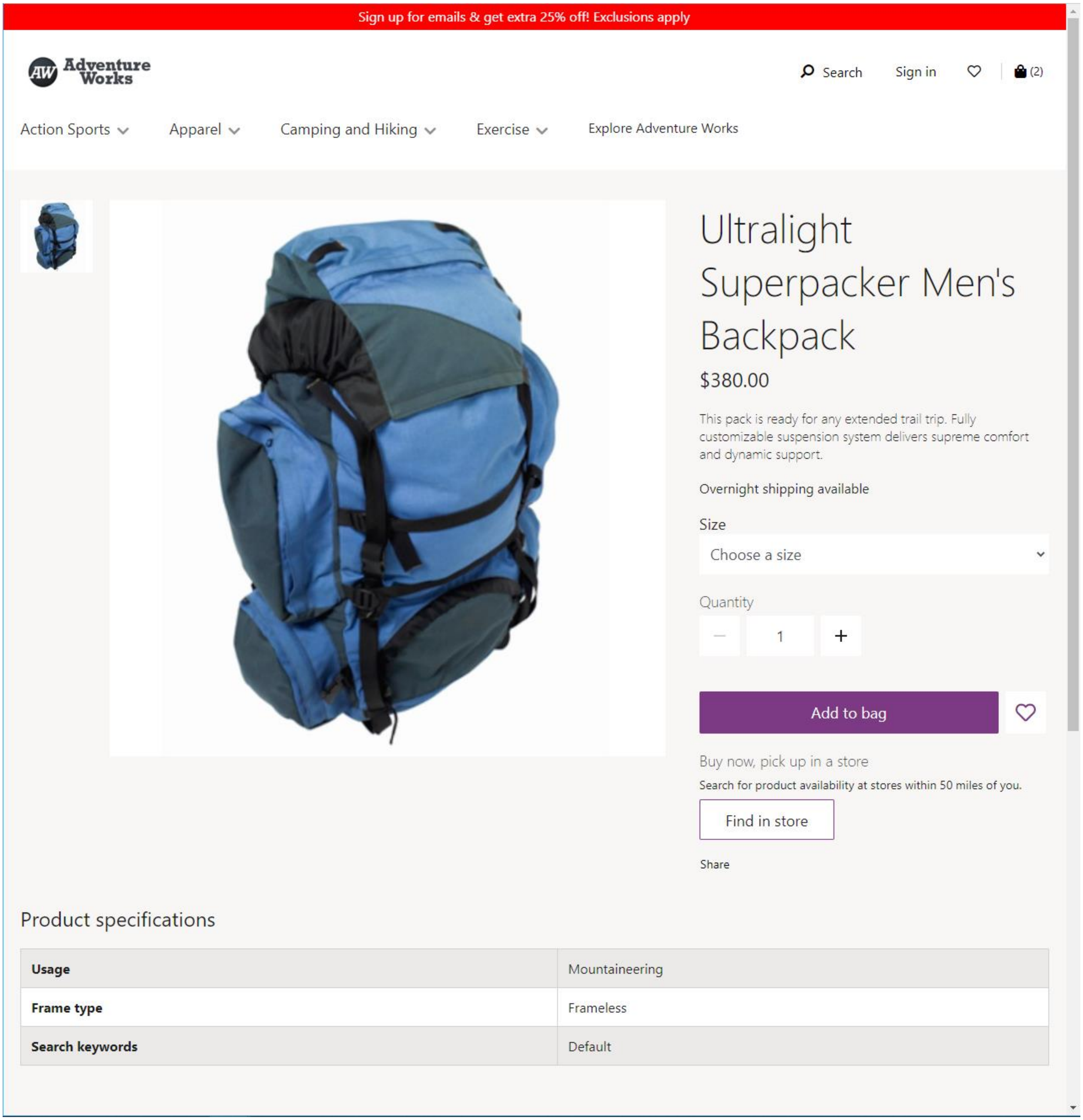
Overnight shipping available

Skip to main text: Skip to main content

Theme: <None>

□ Requires sign-in?

Also, if you navigate to a PDP page you can see the buy box module view extension changes as shown below.



Sign up for emails & get extra 25% off! Exclusions apply

Ultralight Superpacker Men's Backpack

$380.00

This pack is ready for any extended trail trip. Fully customizable suspension system delivers supreme comfort and dynamic support.

Overnight shipping available

Buy now, pick up in a store

Search for product availability at stores within 50 miles of you.

Product specifications

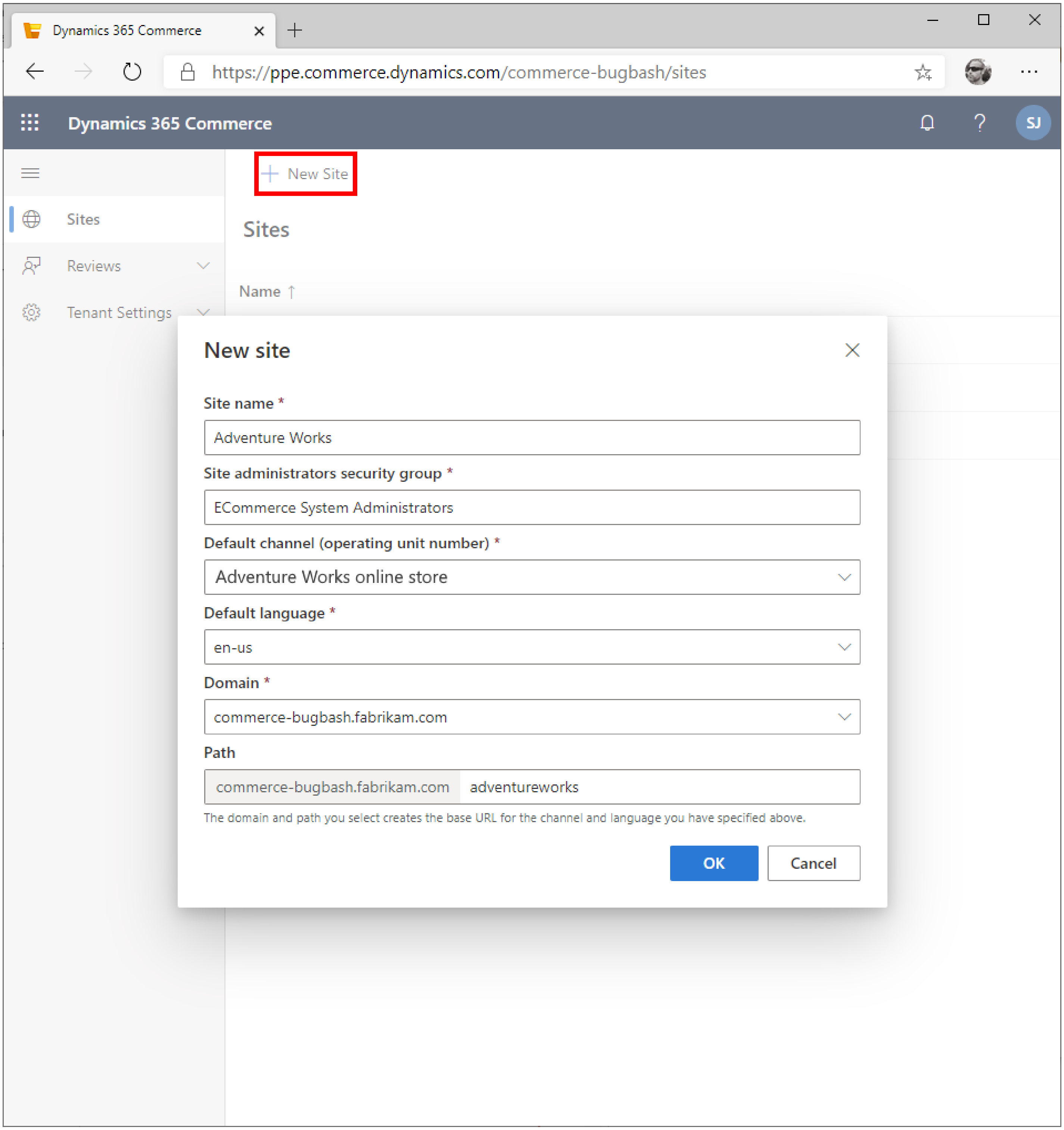
Usage	Mountaineering
Frame type	Frameless
Search keywords	Default

This completes unit 3, for more advanced extensibility topics visit the online SDK documentation.

APPENDIX A – CREATING A NEW E-COMMERCE SITE

In the scenario where you need to create a brand-new e-Commerce site inside of the Site Builder tool, you will notice, no starter pages, templates, images, … are included. Starter pages are only included with the first run experience of the "Fabrikam" site as configured in Unit 2. This section provides detailed instructions on creating a new site and the starter pages that go with it.

The home page of the Site Builder tool should look like the below screen shot. Select the "**New Site**" button to create a new site.



New site

Site name *

Adventure Works

Site administrators security group *

ECommerce System Administrators

Default channel (operating unit number) *

Adventure Works online store

Default language *

en-us

Domain *

commerce-bugbash.fabrikam.com

Path

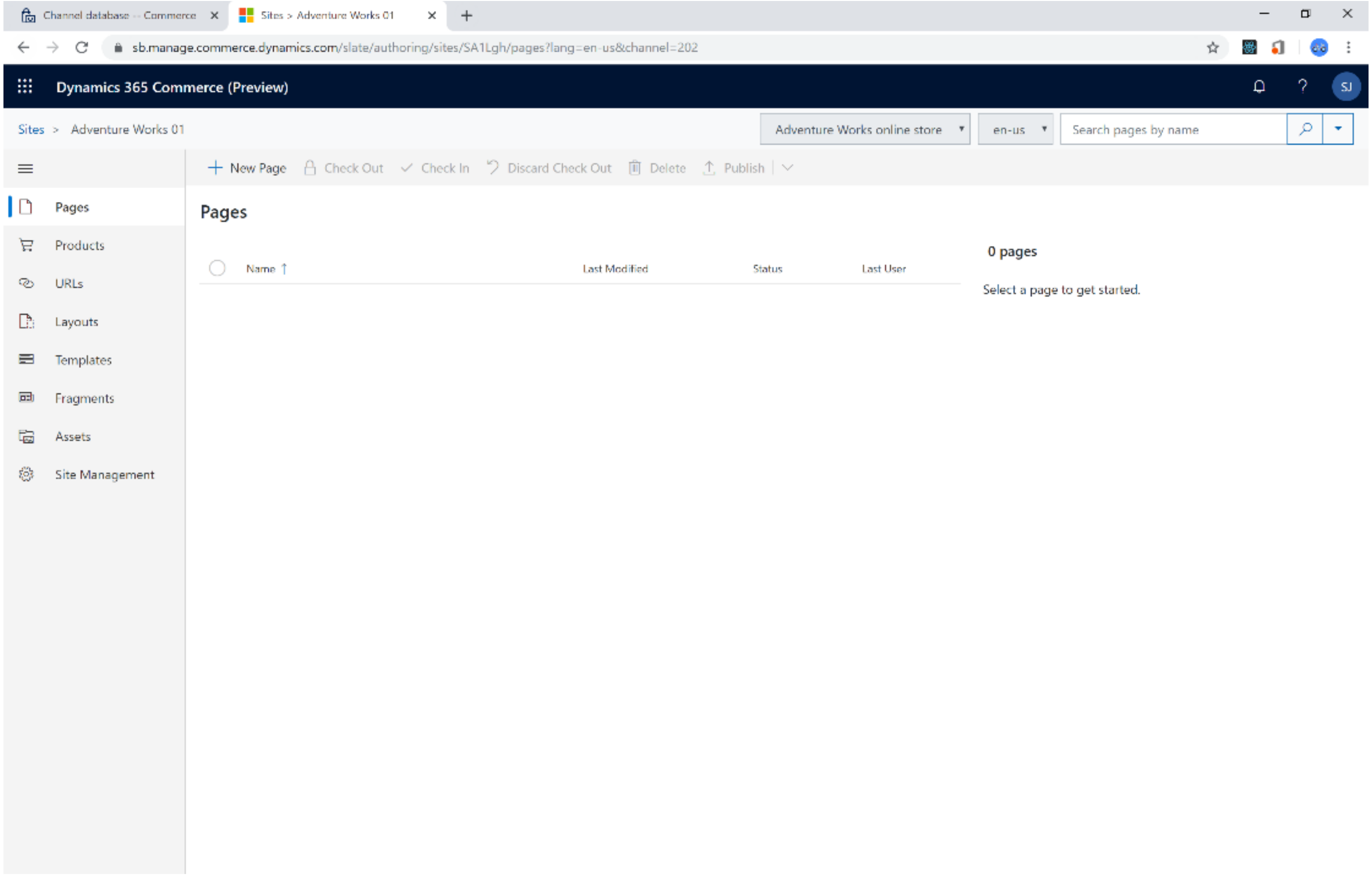
commerce-bugbash.fabrikam.com adventureworks

The domain and path you select creates the base URL for the channel and language you have specified above.

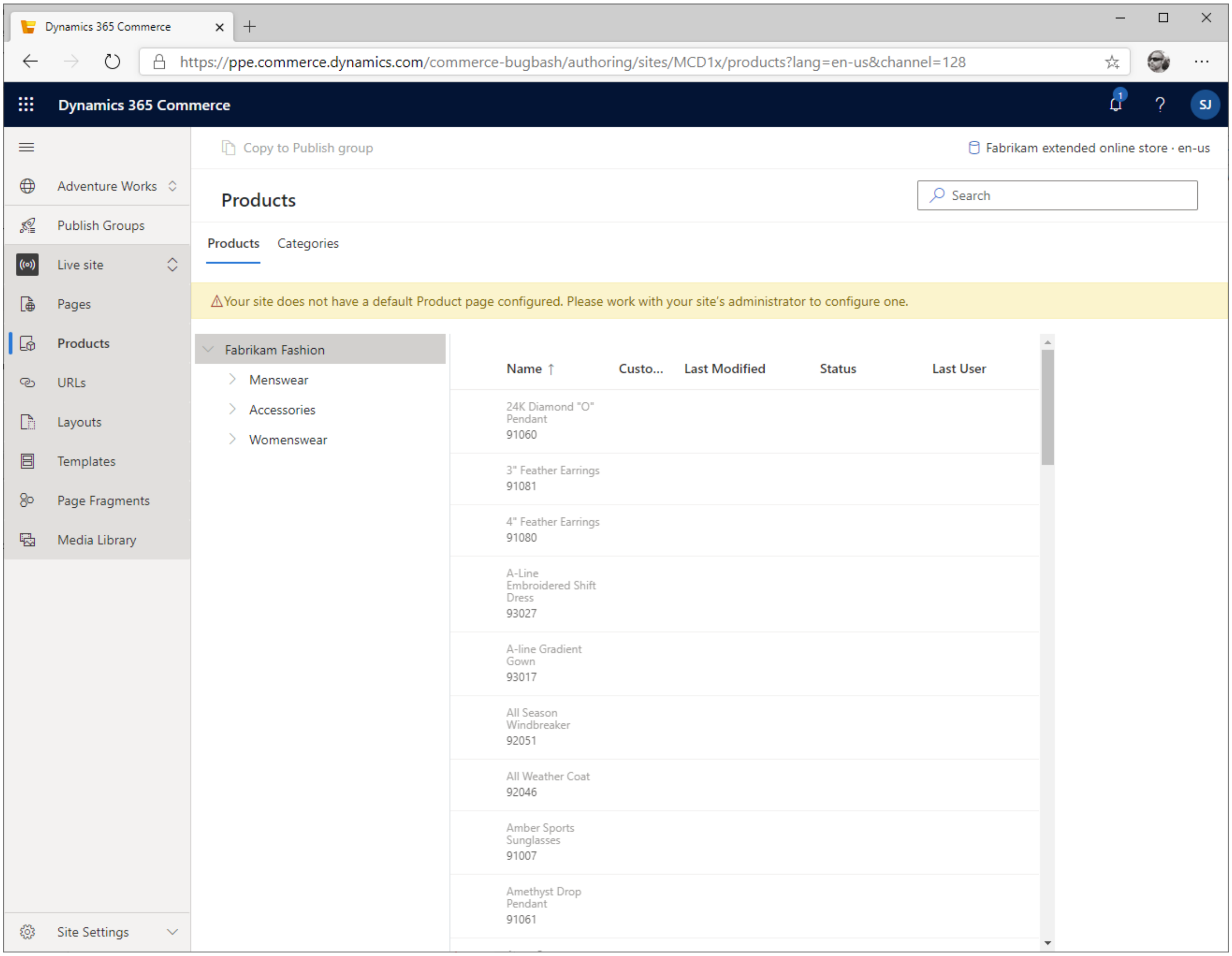
Next you need to fill out some site information

- **Site name** – friendly name for your site
- **Site administrator's security group** - this is the AD account that will have admin access for this group
- Select the **default channel** – this is the new online channel created earlier in this guide
- Set a **default language**
- Select a **domain name** (these were provided when the e-Commerce component was provisioned in LCS)
- **Path** – add a path for the site for the selected domain

You should now be taken to the new site, which is blank (no layout/pages/…).


Channel database -- Commerce
Sites > Adventure Works 01
sb.manage.commerce.dynamics.com/slate/authoring/sites/SA1Lgh/pages?lang=en-us&channel=202
Dynamics 365 Commerce (Preview)
SJ
Sites > Adventure Works 01
Adventure Works online store
en-us
Search pages by name
New Page
Check Out
Check In
Discard Check Out
Delete
Publish
Pages
Products
URLs
Layouts
Templates
Fragments
Assets
Site Management
Pages
Name ↑
Last Modified
Status
Last User
0 pages
Select a page to get started.

To confirm our channel is setup correctly, select the “**Products**” node on the left (ignore the error message for now), you should see products in it list that you added to the channel navigation hierarchy and assortment.



⚠Your site does not have a default Product page configured. Please work with your site's administrator to configure one.

Name ↑	Custo...	Last Modified	Status	Last User
24K Diamond "O" Pendant 91060				
3" Feather Earrings 91081				
4" Feather Earrings 91080				
A-Line Embroidered Shift Dress 93027				
A-line Gradient Gown 93017				
All Season Windbreaker 92051				
All Weather Coat 92046				
Amber Sports Sunglasses 91007				
Amethyst Drop Pendant 91061				

APPENDIX B – CREATE SITE CONTENT PAGES

OVERVIEW

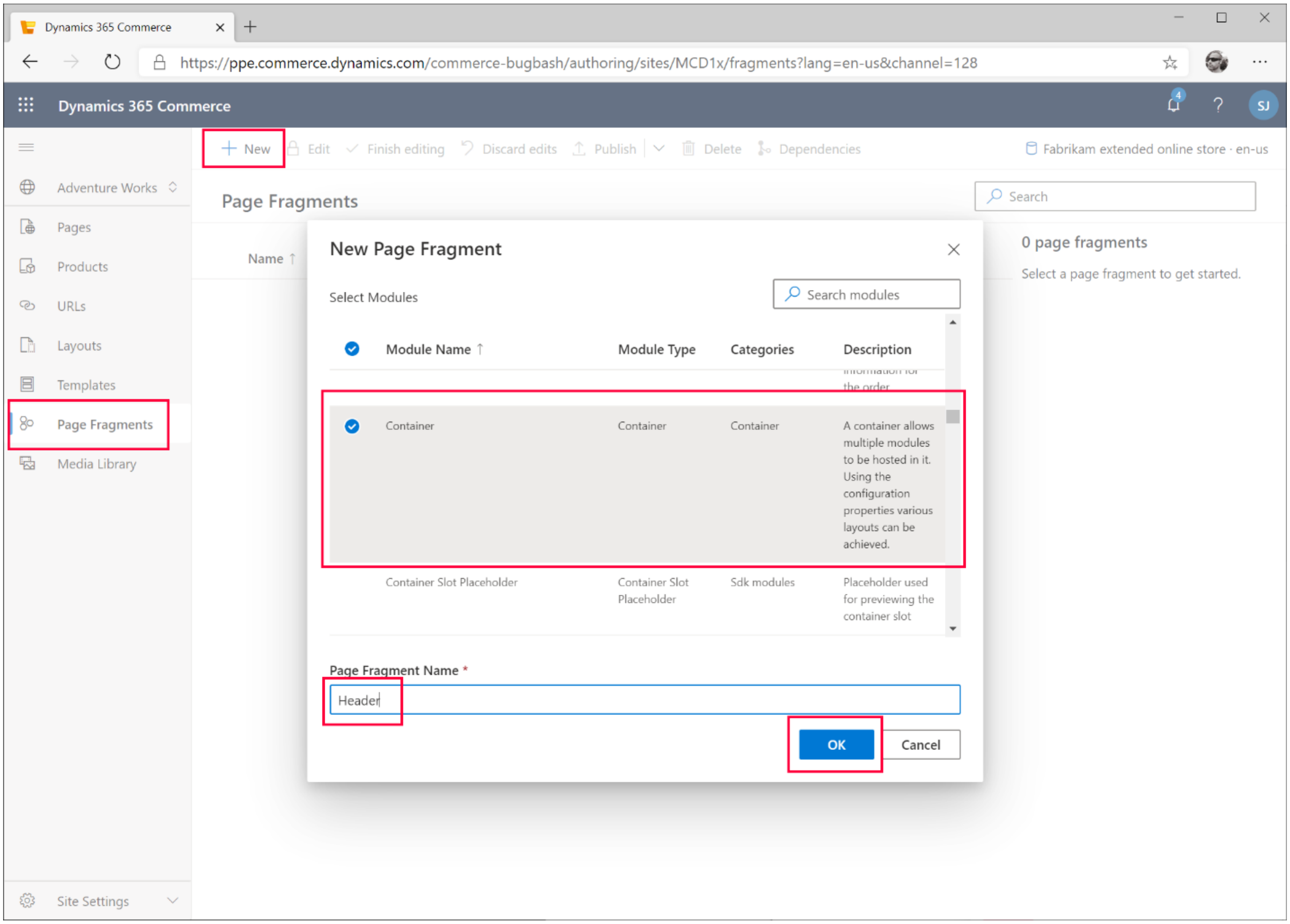
In the following sections we will show how to hand craft various pages that make up an e-Commerce site. These pages can also be found pre-created as part of the Store Starter Kit. We will walk through the general setup to get a home page, category page and product details page created. There are many other pages needed for a full site configuration and looking at the starter kit is a great way to see and learn how to create those.

We will start with creating a branded header and footer for each page. We will create these as “fragments” which allows you to create a re-usable module (or set of modules) that can be placed on templates or pages. This will allow us to create/configure in a single location and not on every page.

CREATE A HEADER FRAGMENT

Select the "**Fragments**" on the left followed by the "**New"** button.

We want this fragment to be made up of the "Header" module. This module is considered a "container" module, which defines various slots (think of slots as UI regions) where it will layout parts of the header. These parts include a logo, category hierarchy and search box. The Header module is designed to hous e a navigation menu and search, however we want our header to be more flexible and contain a cookie compliance notice and a promotional banner. So, we will make the parent of our fragment a generic "**Container**" module. Select "Container" from the list of modules and provide a fragment name such as "**Header**" and click the "**OK**" button.



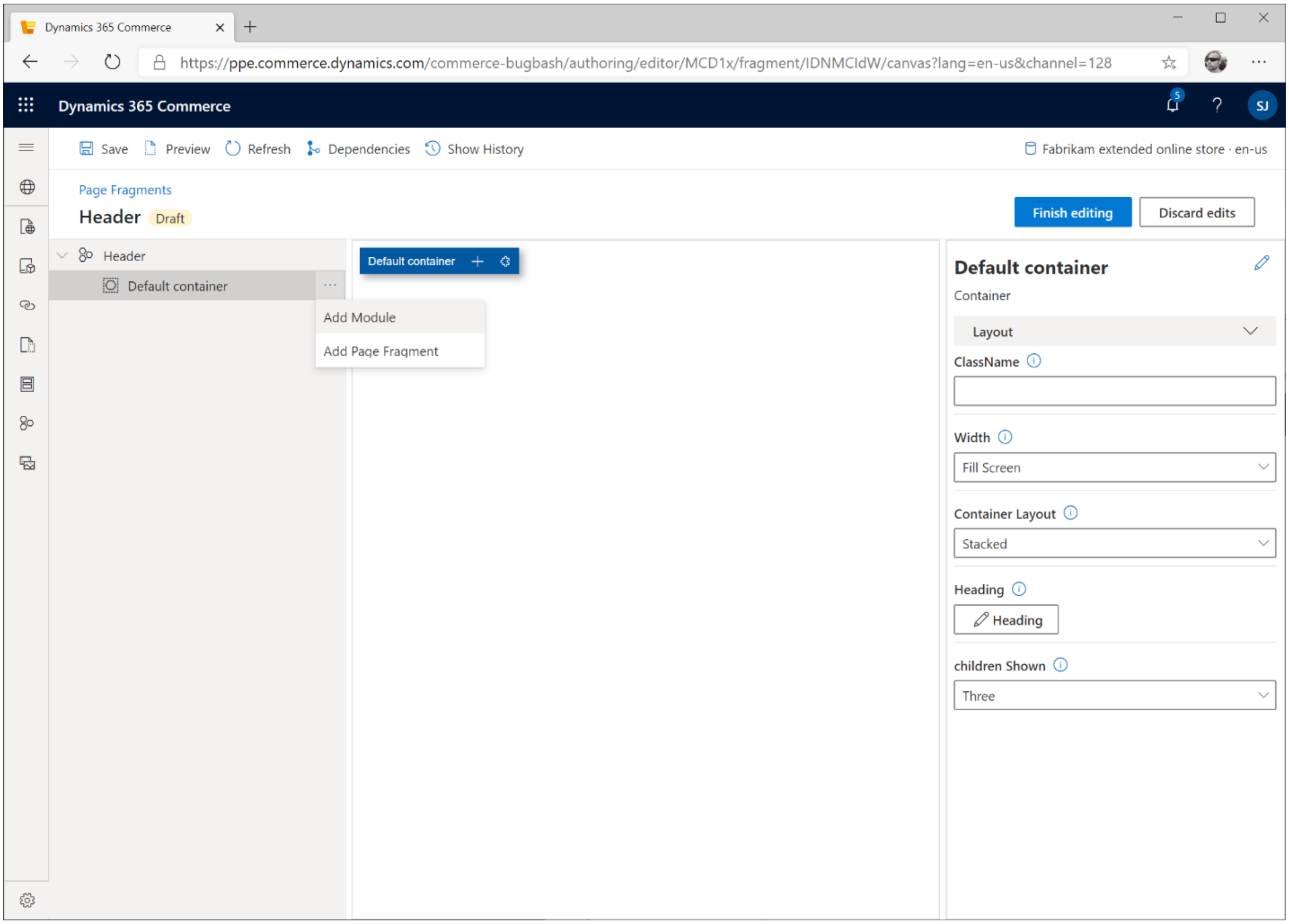
Module Name ↑	Module Type	Categories	Description
			information for the order
Container	Container	Container	A container allows multiple modules to be hosted in it. Using the configuration properties various layouts can be achieved.
Container Slot Placeholder	Container Slot Placeholder	Sdk modules	Placeholder used for previewing the container slot

Page Fragment Name *

Header

You should now be in the fragment editor, notice a tree view of the Fragment structure on the left, a preview in the middle and a properties column on the right. Note: you can select the top left 'hamburger' menu to collapse the left side tabs to get more room on the page.

The top level of the tree represents our fragment and the "Default container" is our container module. Select the "Default container" module and notice module configuration properties show up on the left side, we'll leave these as their default values. Select the "**…**" button and then "**Add Module**".



Add a total of 3 modules to the “Default container”

- Cookie consent
- Promo banner
- Header

Select the “Cookie consent” and add the following configurations

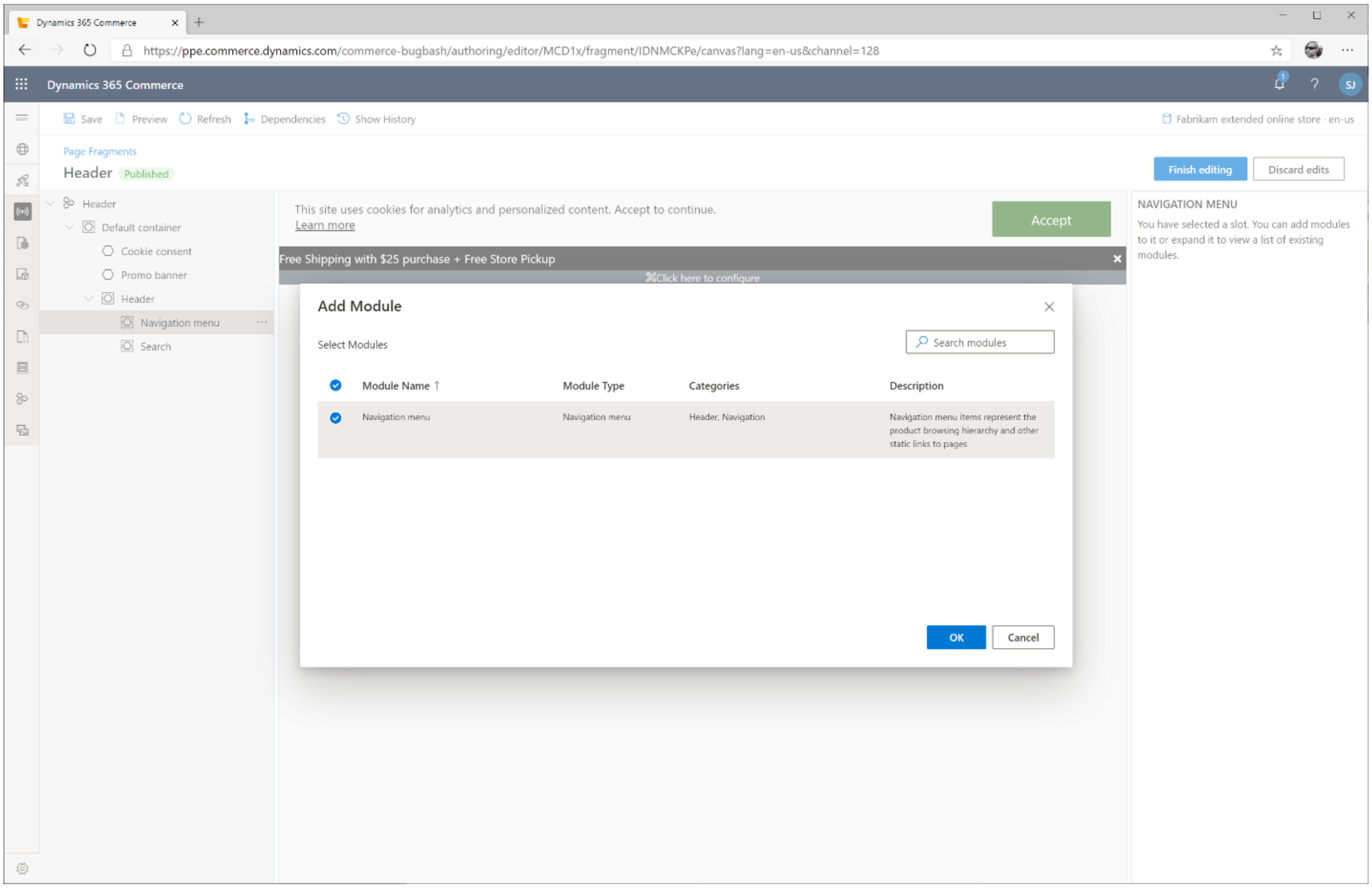
- “Rich text” configuration: “This site uses cookies for analytics and personalized content. Accept to continue.”
- “Links” -> select “Add Action Link” then select the new “Action Link” button and add “Learn more” to the “Link Text”, we’ll leave the “Link target” empty for now since we don’t have a page built to link it to.

Select the “Promo Banner” and configure the following

- Select “Add Message” button and add the following message “Free Shipping with $25 purchase + Free Store Pickup”

The “Header” module contains two “slots” that can be configured. Within each slot you can add other modules or fragments. We’ll be adding other modules into each slot. Select the “Navigation menu” slot and then the “…”

You should see the “Add Module” dialog box, this will only show modules that are ‘allowed’ to be added to the selected slot. (Note: when modules are created, the Module developer has the ability to define which modules are allowed with each slot, they can also choose to allow ‘any’ module).



This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

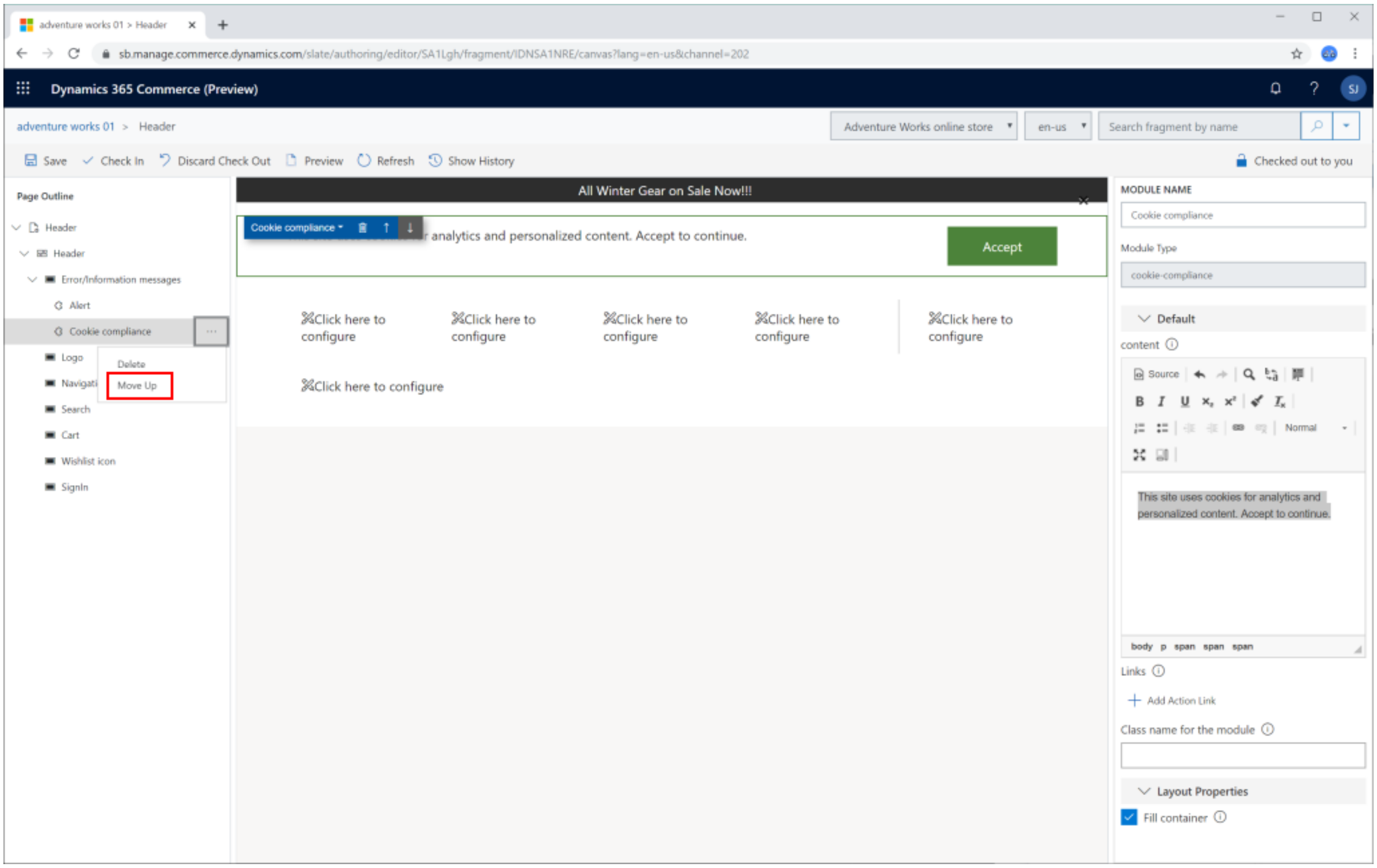
Free Shipping with $25 purchase + Free Store Pickup

You have selected a slot. You can add modules to it or expand it to view a list of existing modules.

✓	Module Name ↑	Module Type	Categories	Description
✓	Navigation menu	Navigation menu	Header, Navigation	Navigation menu items represent the product browsing hierarchy and other static links to pages

Select the “**Navigation menu**” module and click the “**OK**” button. This module is used to show a message at the top of a page. It can be configured to have a dismiss button if desired.

Select the “**Search**” slot and then “…” followed by “**Add Module**” and select the “Search” module.



r analytics and personalized content. Accept to continue.

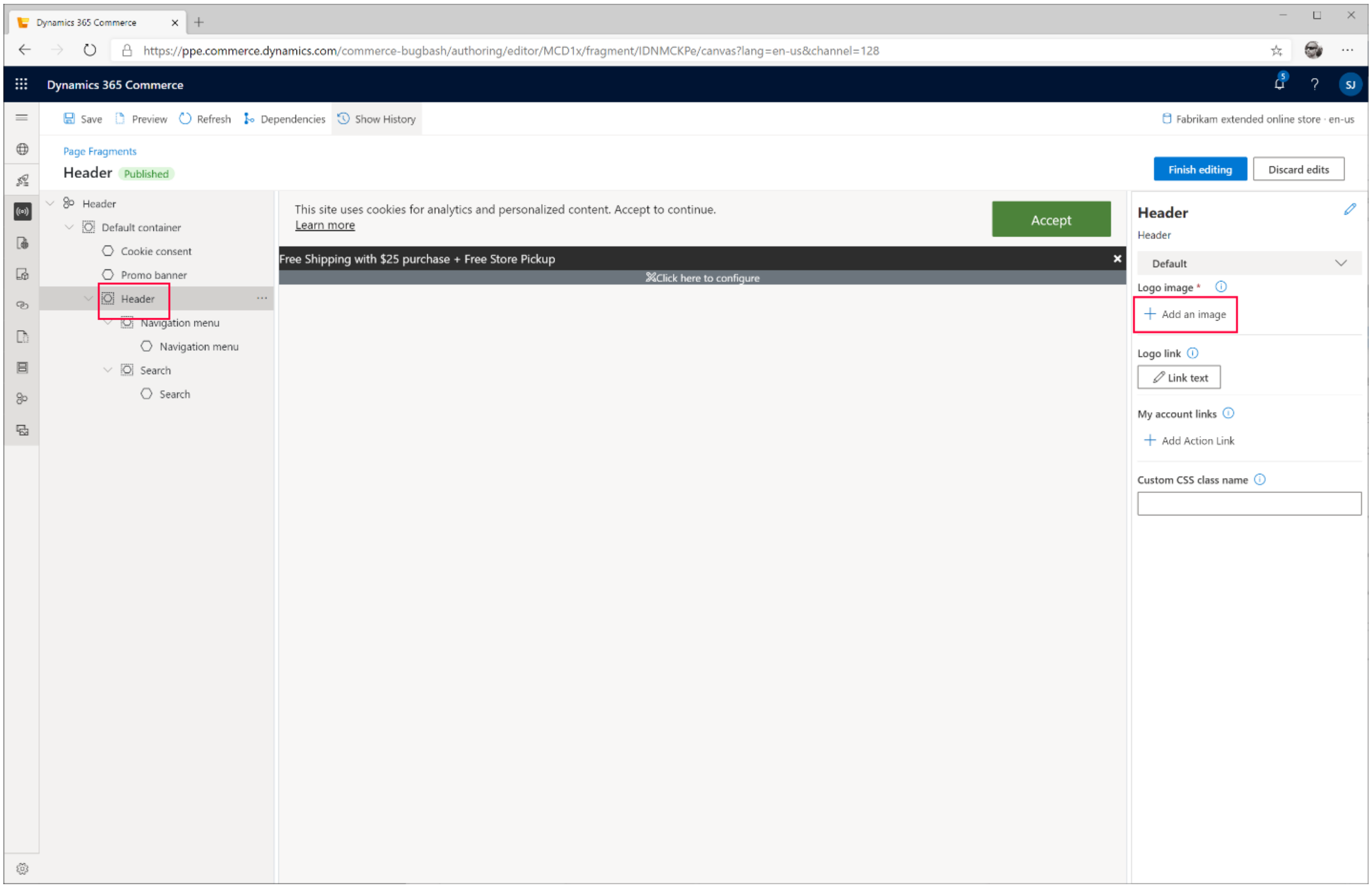
MODULE NAME: Cookie compliance

Module Type: cookie-compliance

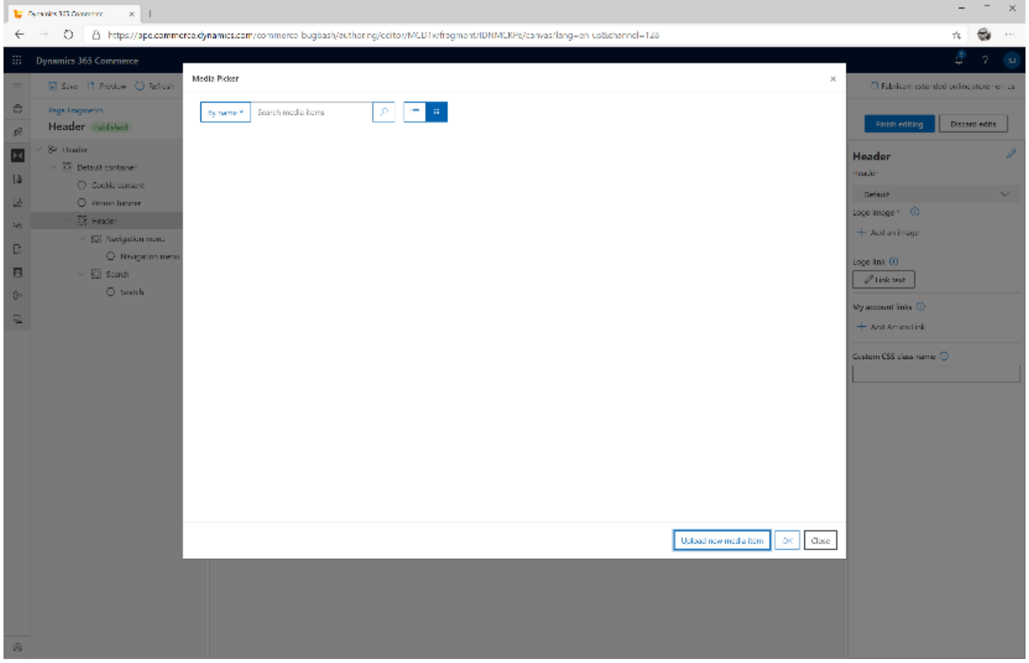
content: This site uses cookies for analytics and personalized content. Accept to continue.

☑ Fill container

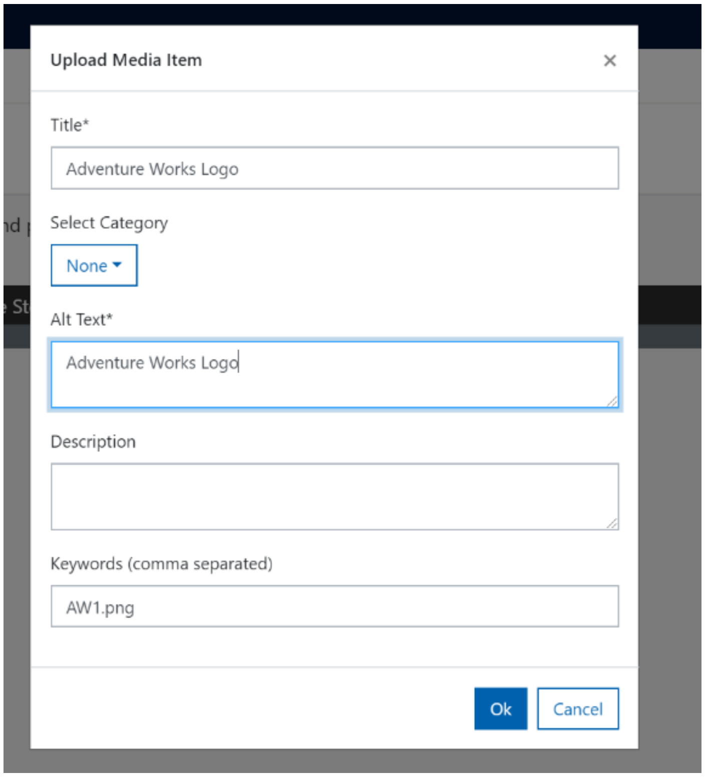
Select the “Header” module and “Add an image” to upload a logo.



The “Media Picker” dialog will appear, select the “**Upload new media item**” button and upload a logo.


Dynamics 365 Commerce
https://ape.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/fragment/IDNMCKPe/canvas?lang=en-us&channel=128
Dynamics 365 Commerce
Save
Preview
Refresh
Page fragments
Header
Published
Header
Default container
Cookie consent
Promo banner
Header
Navigation menu
Navigation menu
Search
Search
Media Picker
By name
Search media items
Upload new media item
OK
Close
Fabrikam extended online store - en-us
Finish editing
Discard edits
Header
Header
Default
Logo image *
Add an image
Logo link
Link text
My account links
Add Action Link
Custom CSS class name

Add a title and “Alt Text” and click “OK”. Next select the new image and click ok.



Upload Media Item

Title*: Adventure Works Logo

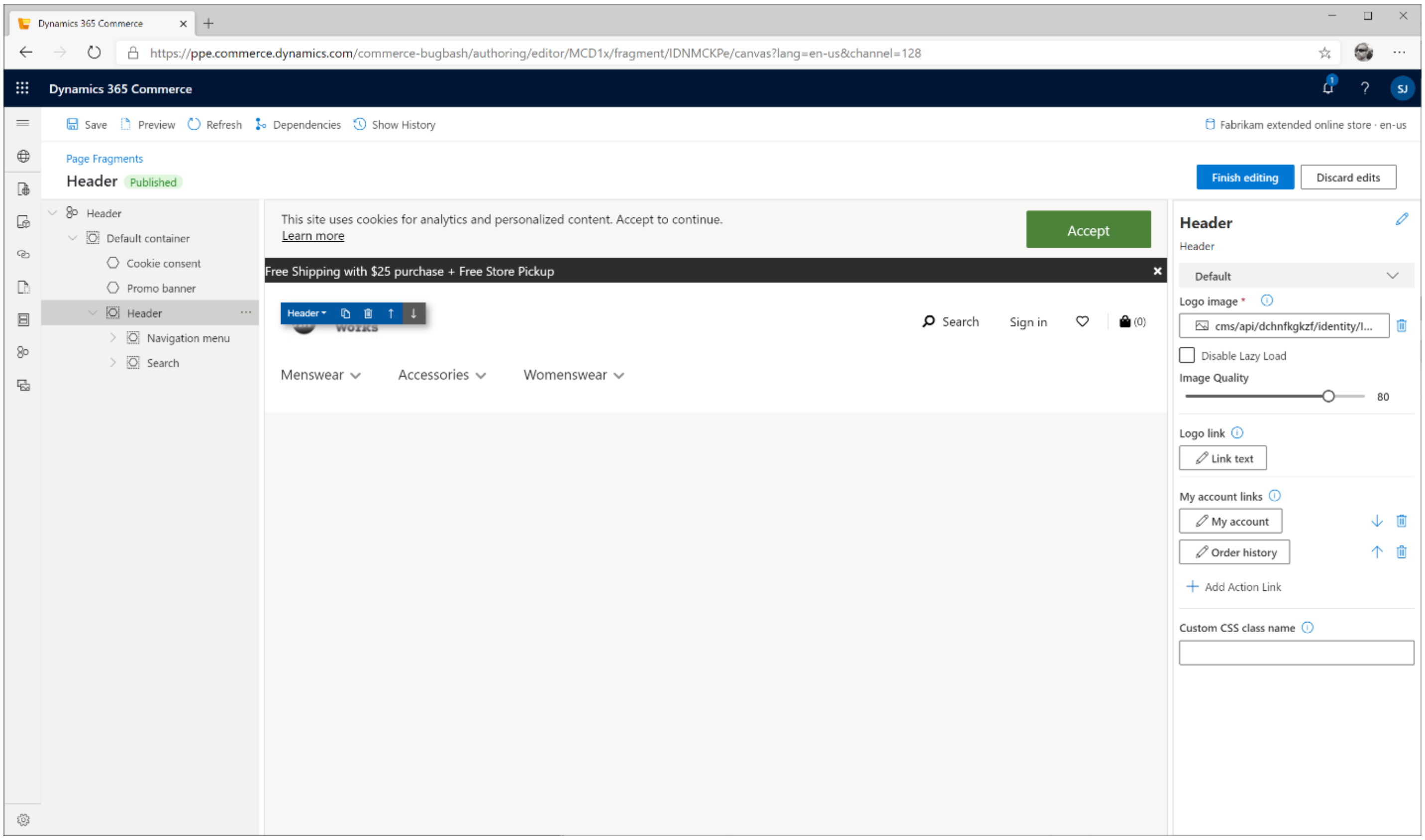
Select Category: None

Alt Text*: Adventure Works Logo

Description:

Keywords (comma separated): AW1.png

Under the “My account links”, add 2 one for “My account” and another for “Order history”, we’ll leave the links empty for now until we have pages to link those to.



Dynamics 365 Commerce
https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/fragment/IDNMCKPe/canvas?lang=en-us&channel=128
Dynamics 365 Commerce
SJ
Save
Preview
Refresh
Dependencies
Show History
Fabrikam extended online store · en-us
Page Fragments
Header
Published
Finish editing
Discard edits
Header
Default container
Cookie consent
Promo banner
Header
Navigation menu
Search
This site uses cookies for analytics and personalized content. Accept to continue.
Learn more
Accept
Free Shipping with $25 purchase + Free Store Pickup
Header
Search
Sign in
(0)
Menswear
Accessories
Womenswear

Header

Header

Default

Logo image * cms/api/dchnfkgkzf/identity/I...

□ Disable Lazy Load

Image Quality 80

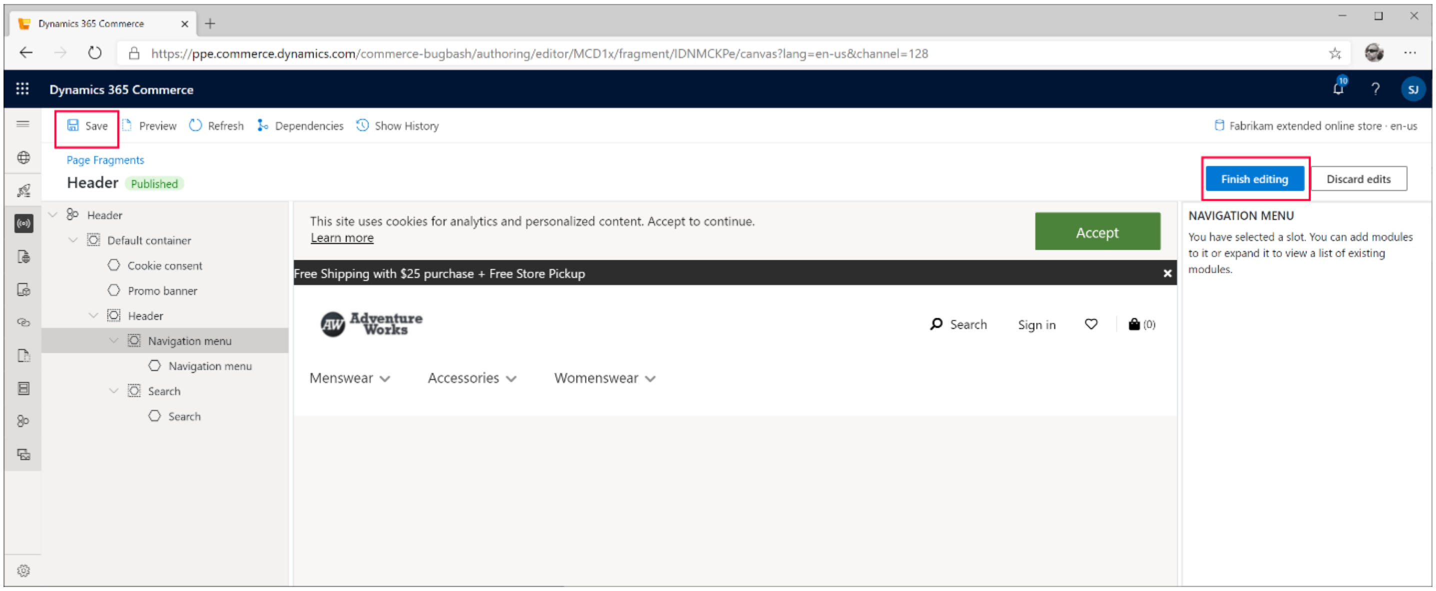
Logo link: Link text

My account links: My account, Order history

Add Action Link

Custom CSS class name

Now select “Save” and the image will be updated in the rendered view. We are now finished so select “Finish editing” button, you can optionally provide a note with the changes. You can also select the “Preview” button at any time which will open the fragment into a new tab.



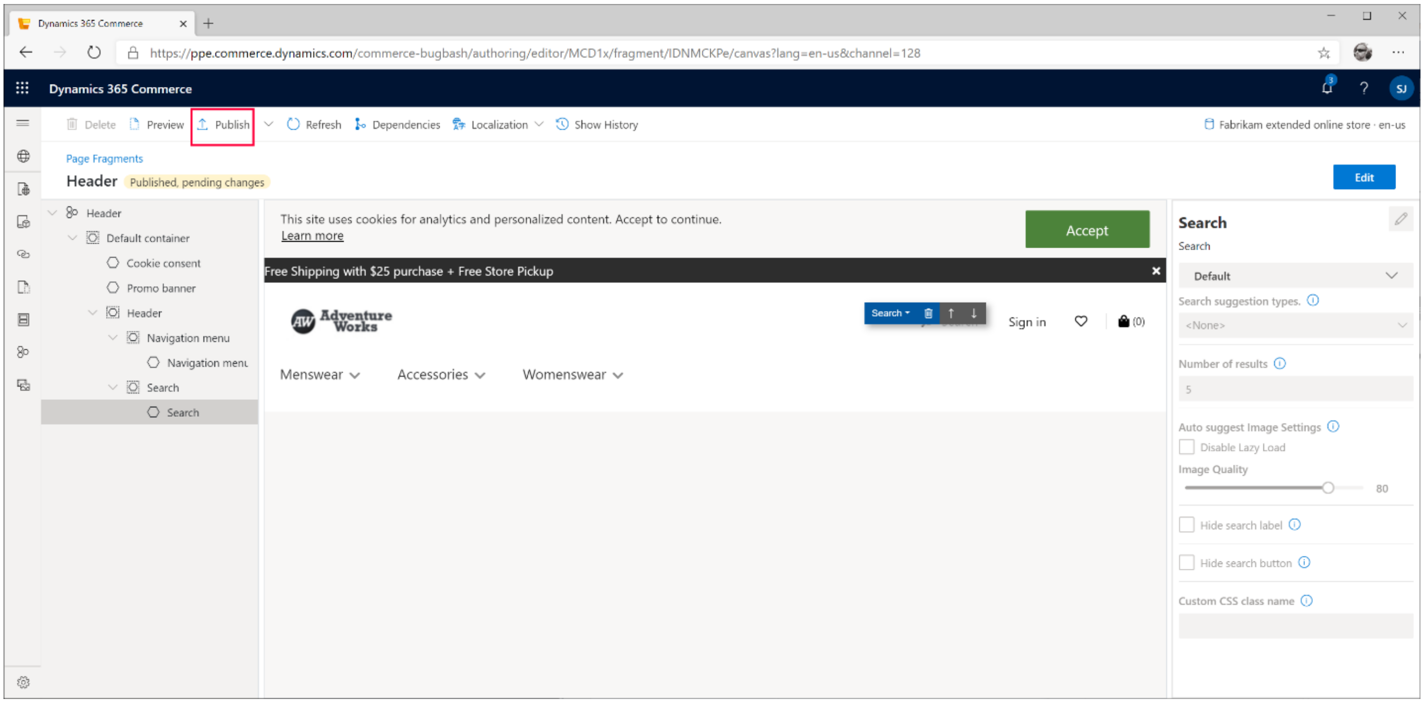
This site uses cookies for analytics and personalized content. Accept to continue. Learn more

Free Shipping with $25 purchase + Free Store Pickup

NAVIGATION MENU

You have selected a slot. You can add modules to it or expand it to view a list of existing modules.

We can now publish the fragment which makes it live by pressing the “Publish” button.



This site uses cookies for analytics and personalized content. Accept to continue. Learn more

Free Shipping with $25 purchase + Free Store Pickup

Search

Search

Default

Search suggestion types.

<None>

Number of results

5

Auto suggest Image Settings

□ Disable Lazy Load

Image Quality

80

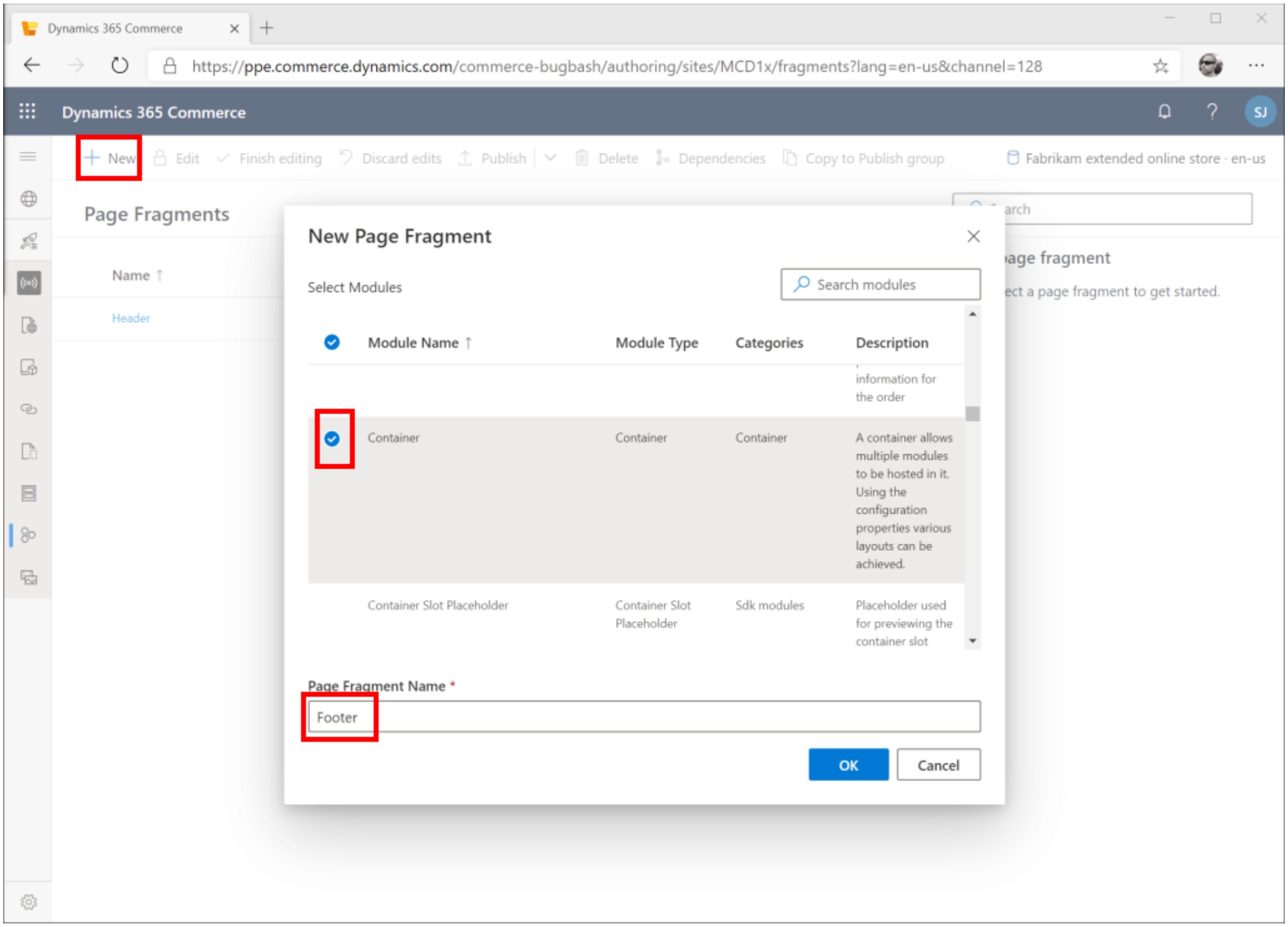
□ Hide search label

□ Hide search button

Custom CSS class name

CREATE A FOOTER FRAGMENT

We're now going to create a footer fragment. Ensure the "**Fragment**" tab is selected and click the "+ **New**" button to bring up the "New Page Fragment" dialog box. Select the "**Container**" module, which will be our root module for this fragment and provide a name such as "**Footer**", followed by pressing the "OK" button.

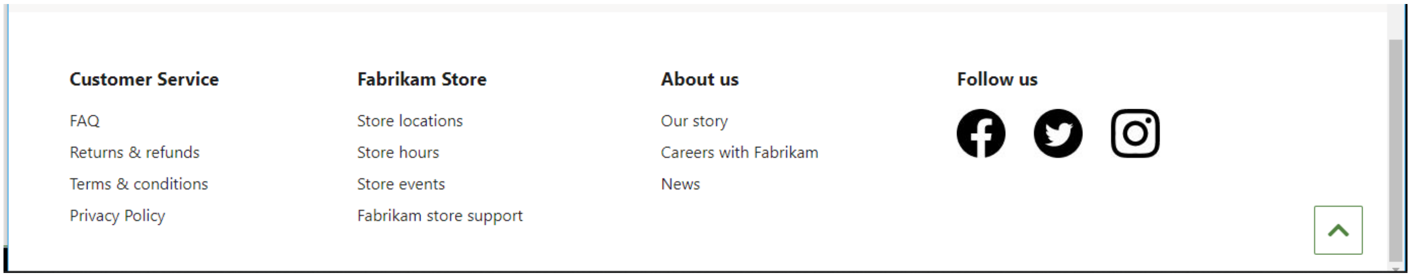


Module Name ↑	Module Type	Categories	Description
			information for the order
Container	Container	Container	A container allows multiple modules to be hosted in it. Using the configuration properties various layouts can be achieved.
Container Slot Placeholder	Container Slot Placeholder	Sdk modules	Placeholder used for previewing the container slot

Page Fragment Name *

Footer

We’ll be generating a footer that looks similar to the below image.



Customer Service

FAQ
Returns & refunds
Terms & conditions
Privacy Policy

Fabrikam Store

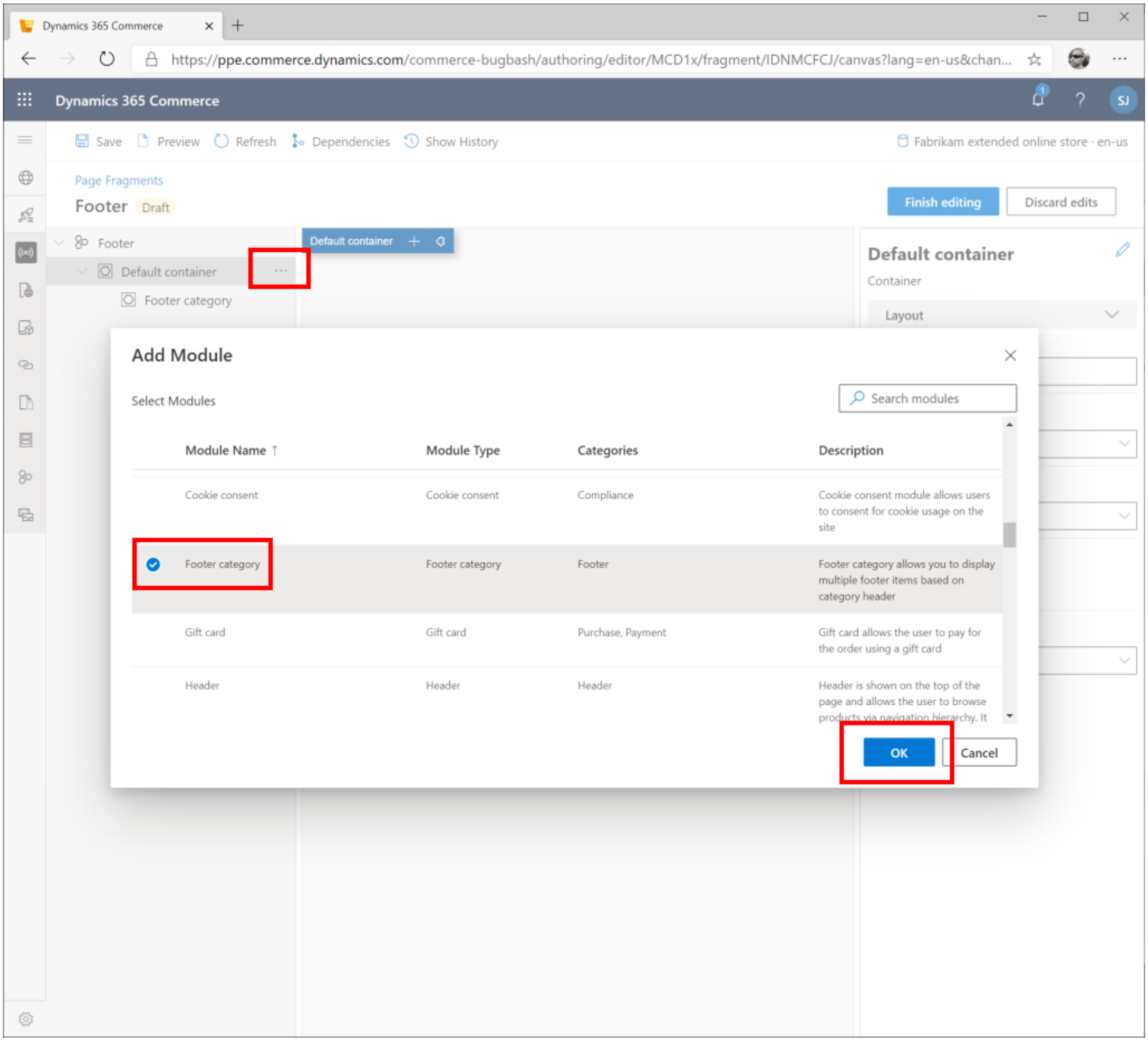
Store locations
Store hours
Store events
Fabrikam store support

About us

Our story
Careers with Fabrikam
News

Follow us

To achieve this, the Footer module can render columns using the “Footer category” module. So for the 4 columns shown in the image above we’ll add five “Footer category” modules to the “Default container” module. Select “Default container” in the tree then the “…” button and “Add Module” button to bring up the “Add module” dialog box. Find the “Footer category” module and add it.



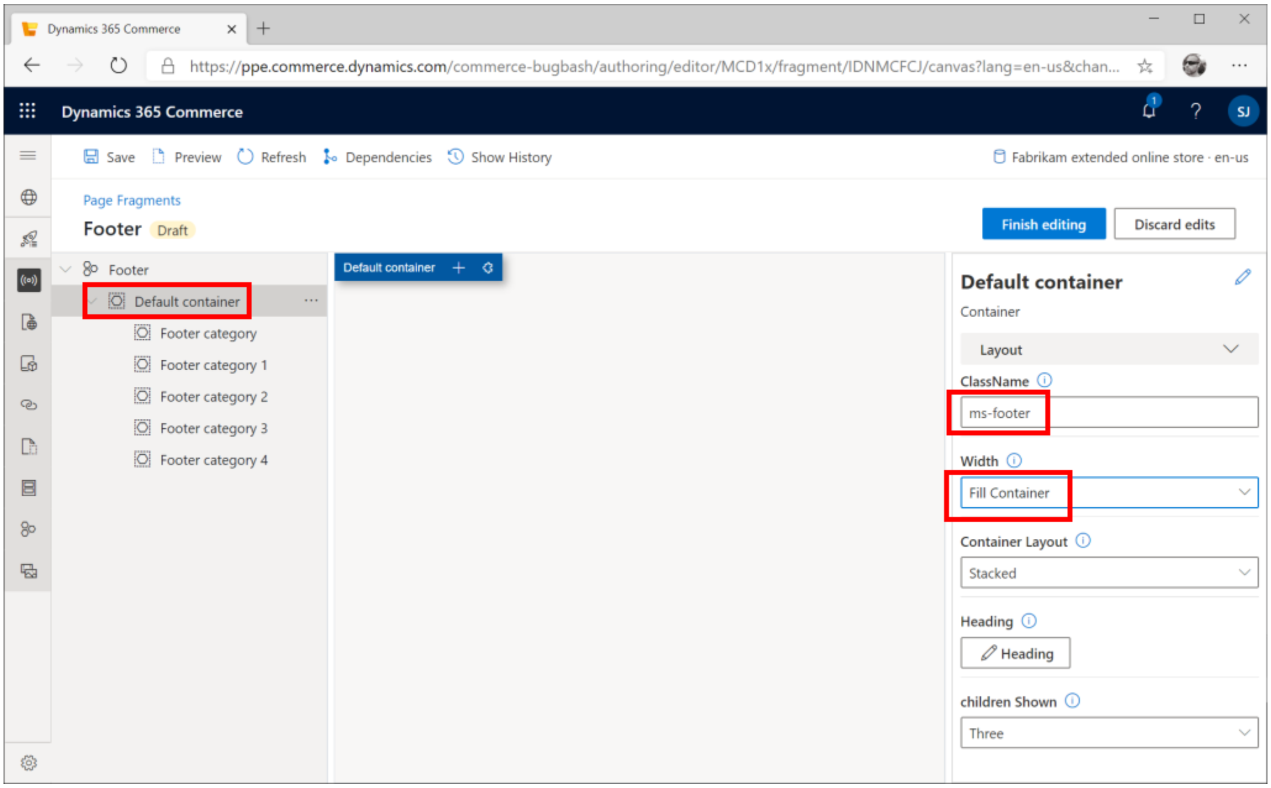
Add Module

Select Modules

	Module Name ↑	Module Type	Categories	Description
	Cookie consent	Cookie consent	Compliance	Cookie consent module allows users to consent for cookie usage on the site
●	Footer category	Footer category	Footer	Footer category allows you to display multiple footer items based on category header
	Gift card	Gift card	Purchase, Payment	Gift card allows the user to pay for the order using a gift card
	Header	Header	Header	Header is shown on the top of the page and allows the user to browse products via navigation hierarchy. It

Do the same until you have 5 “Footer categories”.

Select the “Default container” and configure the ClassName to “ms-footer” and the **Width** to “Fill container”.



ClassName: ms-footer

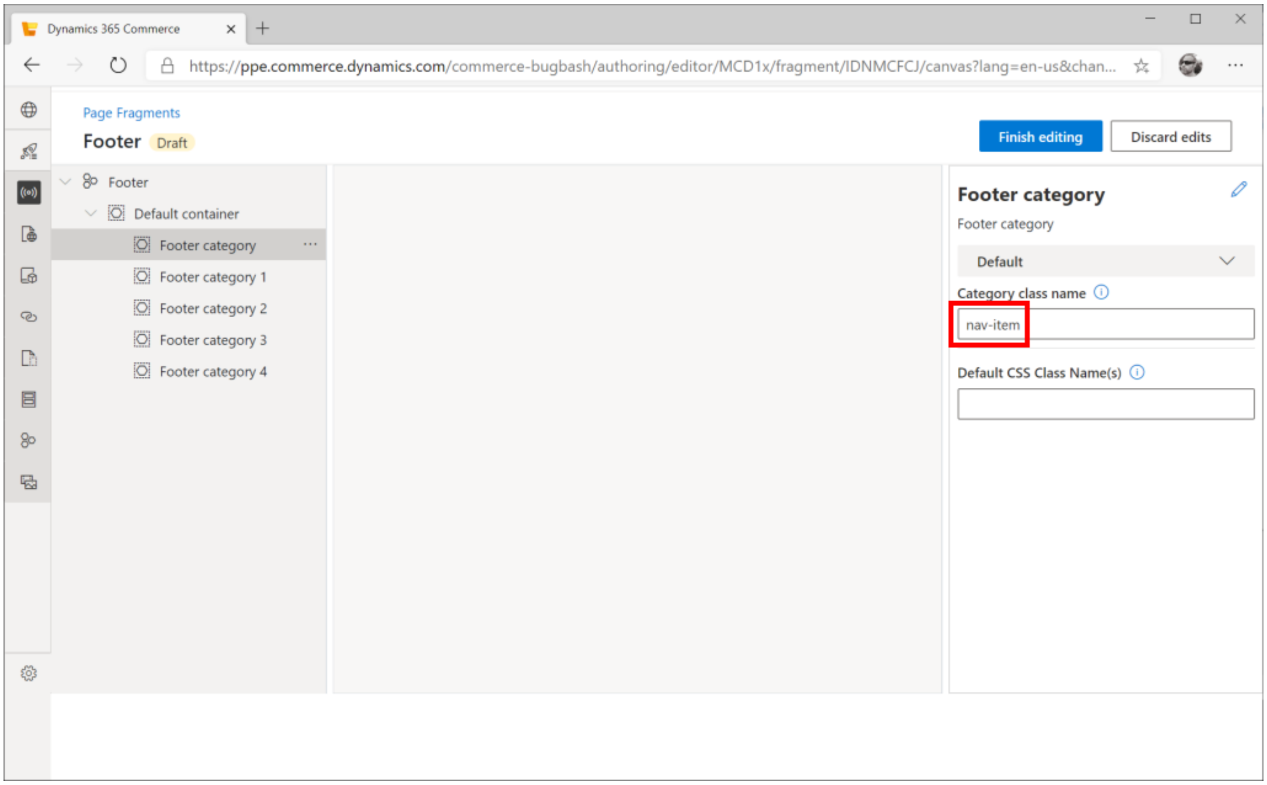
Width: Fill Container

Container Layout: Stacked

Heading: Heading

children Shown: Three

For each Footer category we need to configure the “Category class name”. For example, the first “Footer category”, set the configuration to “nav-item” as shown below.



Footer category

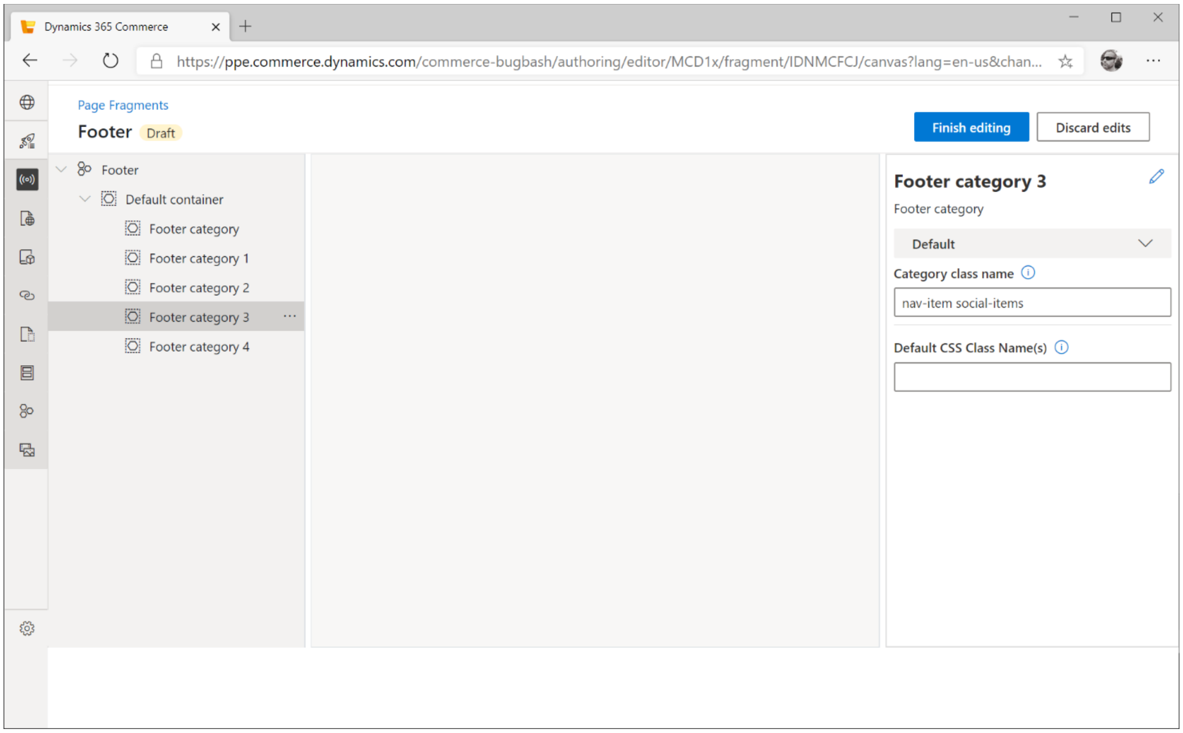
Footer category

Default

Category class name: nav-item

Default CSS Class Name(s):

For “Footer category 3” set “Category class name” to “nav-item social-items” and leave “Footer category 4” class name empty.



Footer category 3

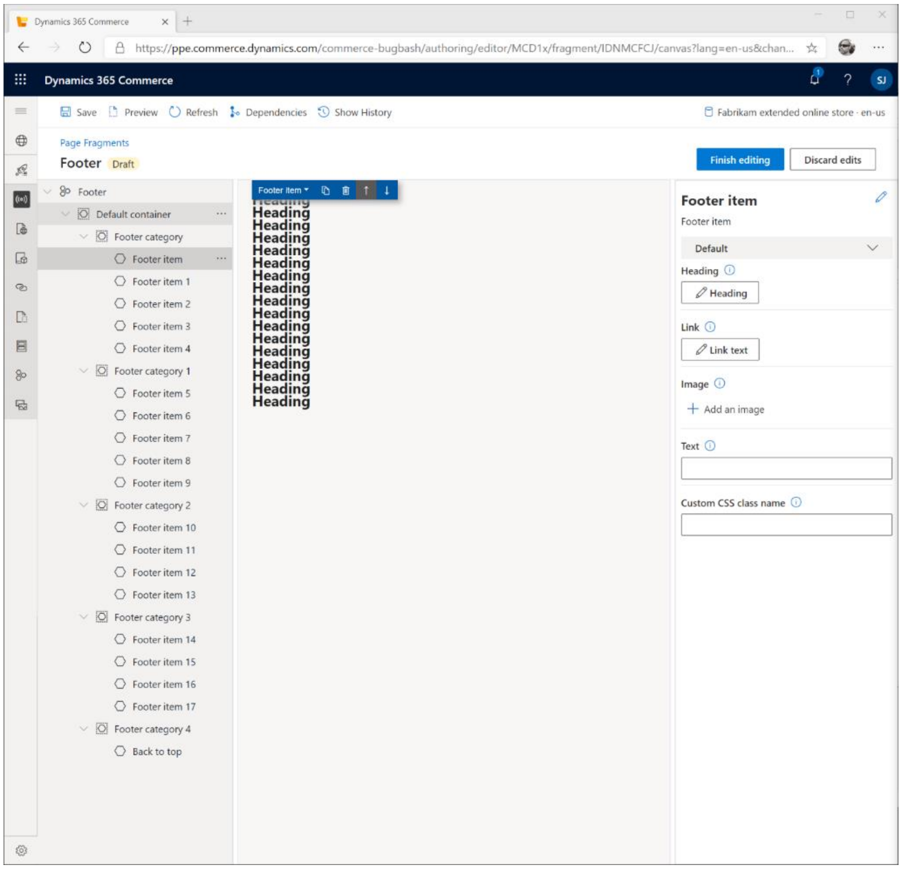
Footer category

Default

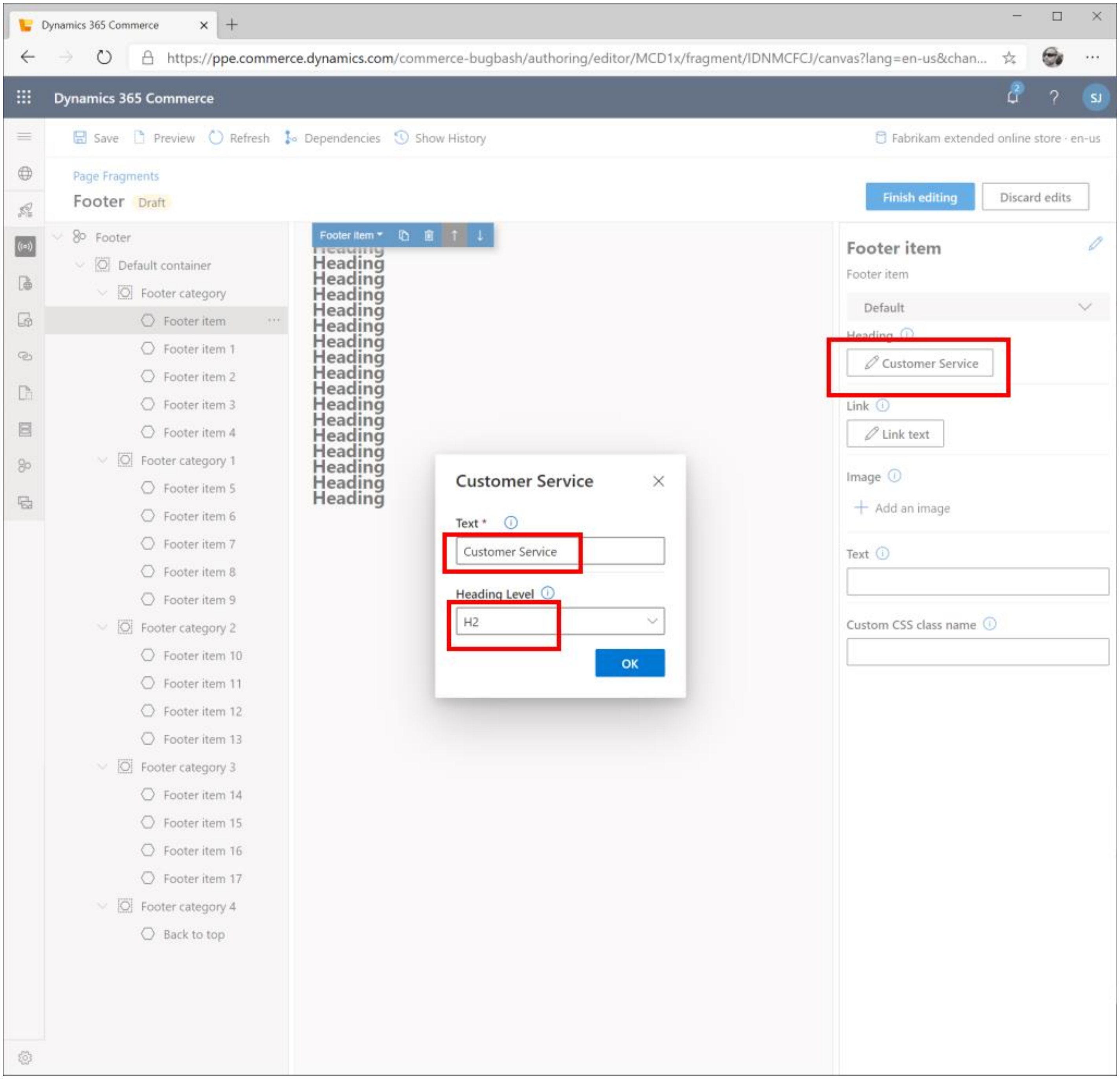
Category class name: nav-item social-items

Default CSS Class Name(s):

We will now add items to each category. Add five “Footer item” modules to “Footer category”, “Footer category 1” and four to “Footer category 2” and “Footer category 3”. Finally add a “Back to top” module to “Footer category 4”.



Select the top footer item in “Footer category” and set the “Heading” text to “Customer Service” with “H2”. This will serve as the column title.



Customer Service

Text *: Customer Service

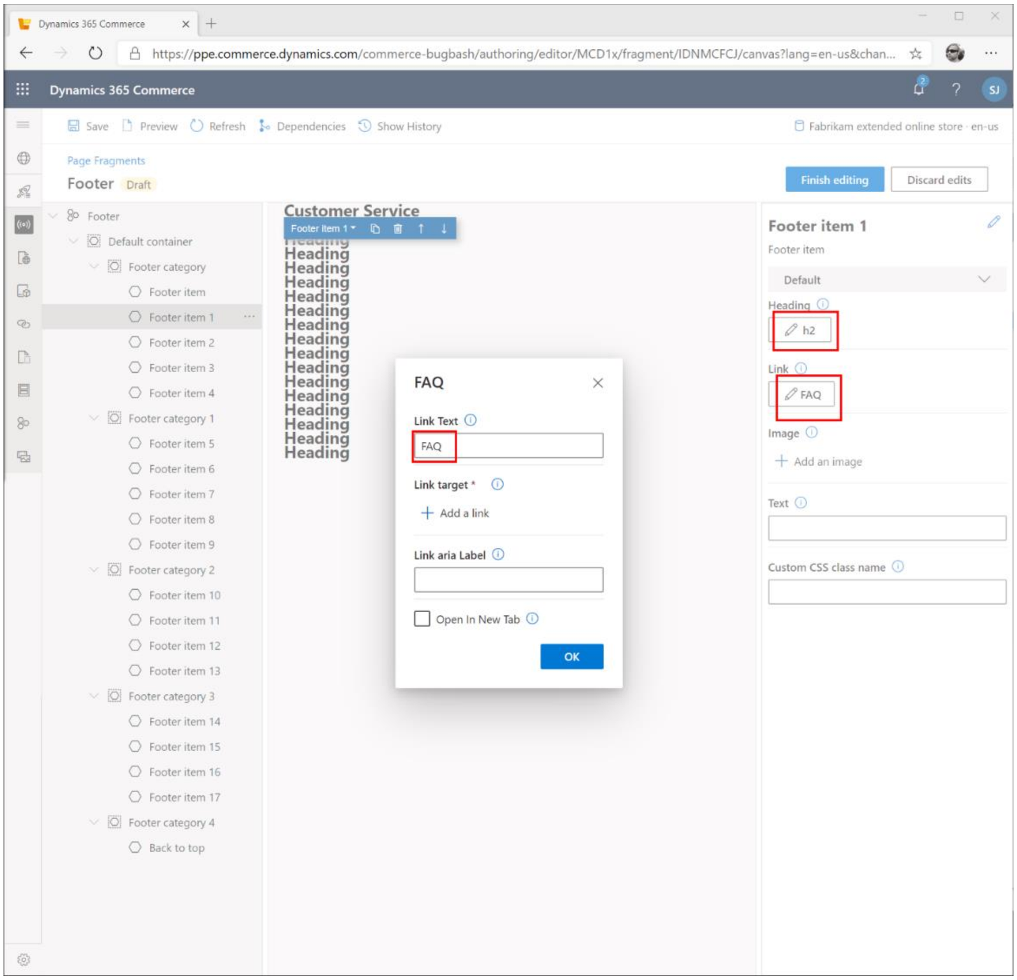
Heading Level: H2

Footer item

Heading: Customer Service

Link: Link text

Select "Footer item 1" and make the "Heading" config an empty string and set the "Link text" config to "FAQ". Since we don't have a FAQ page yet, we'll leave the link empty.



https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/fragment/IDNMCFCJ/canvas?lang=en-us&chan...

Footer item 1

Heading: h2

Link: FAQ

FAQ

Link Text: FAQ

Link target *

Link aria Label

□ Open In New Tab

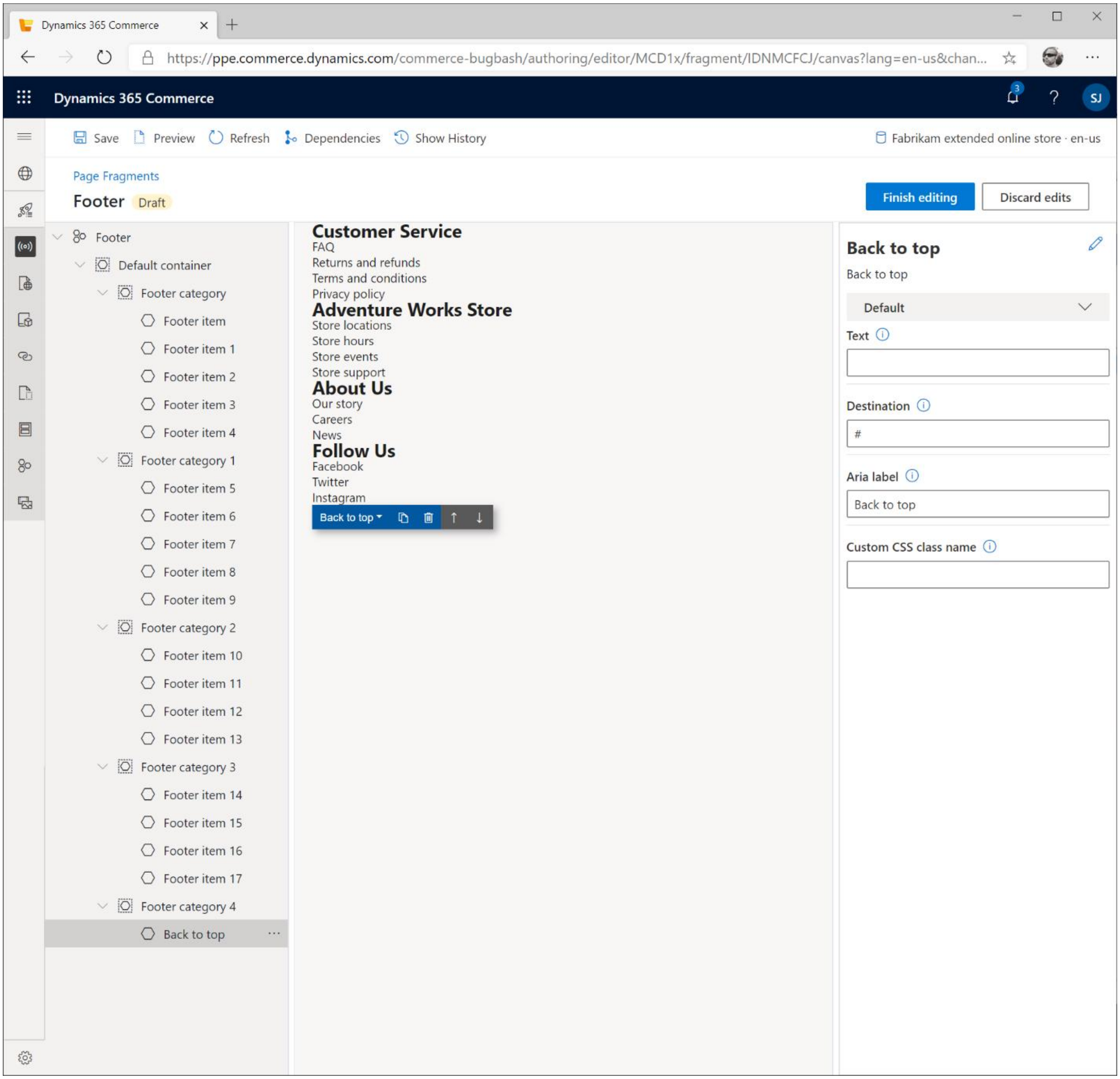
Set the remainder category strings as follows:

- Customer Service
    - FAQ
    - Returns and refunds
    - Terms and conditions
    - Privacy policy
- Adventure Works Store
    - Store locations
    - Store hours
    - Store events
    - Store support
- About Us
    - Our story
    - Careers
    - News
- Follow Us

- Facebook
- Twitter
- Instagram

For the “Back to top” module configure the “Destination” with “#” and “Aria label” to “Back to top”.

The final footer should look like the below image, select “Finish editing” then “Publish”.

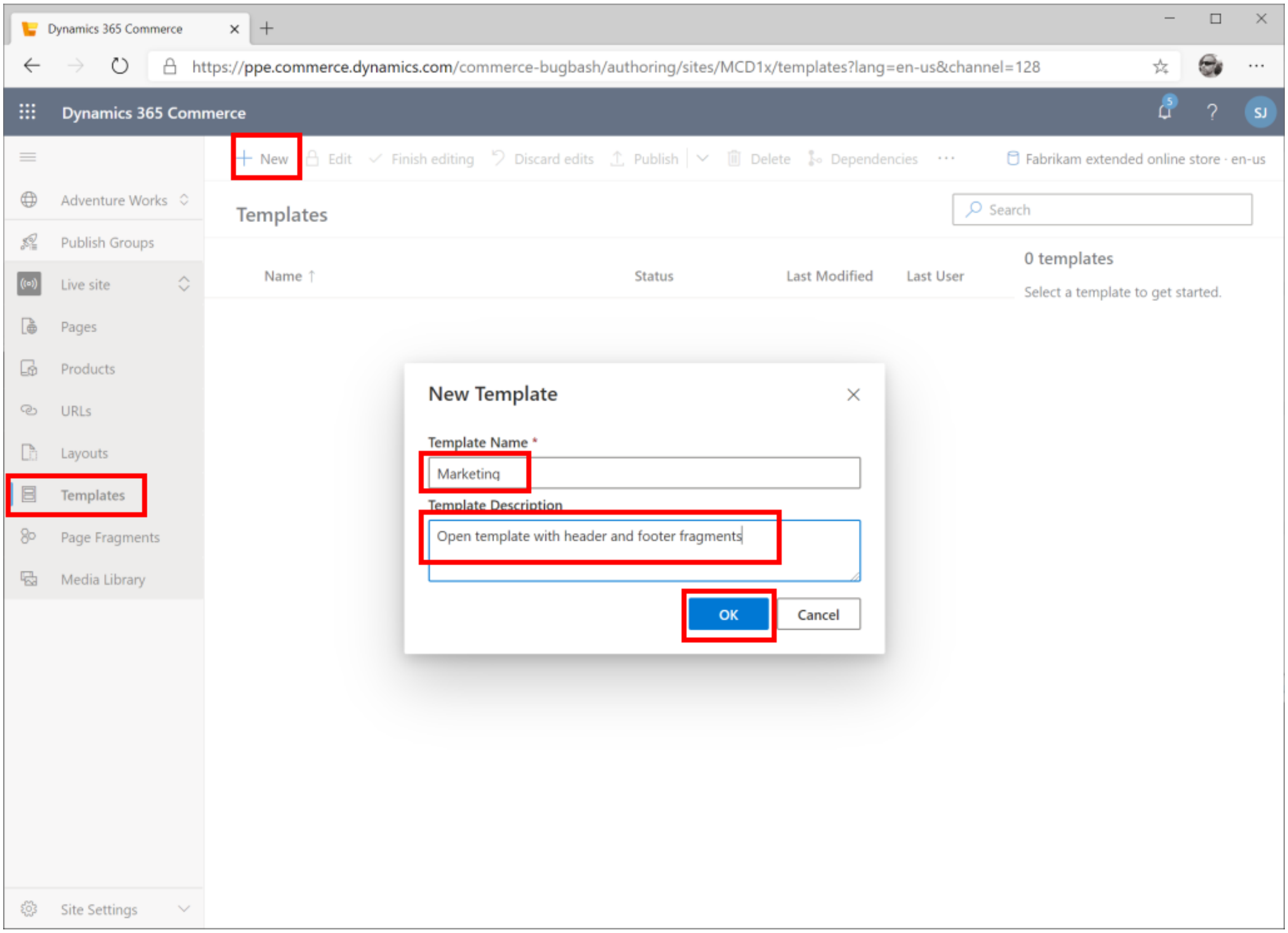


CREATE A HOME TEMPLATE AND PAGE

All e-Commerce pages must adhere to a template which provides page options. This section will show you how to create and configure a template. We will start with a generic template that has a locked header and footer but will allow any content in between. This will then be used to create our Homepage in the next section.

CREATE A HOME TEMPLATE

From the Site authoring tools, select the "**Templates**" tab followed by the "New" button. Provide a name such as "Marketing" and a description followed by the "OK" button.



New Template

Template Name *

Marketing

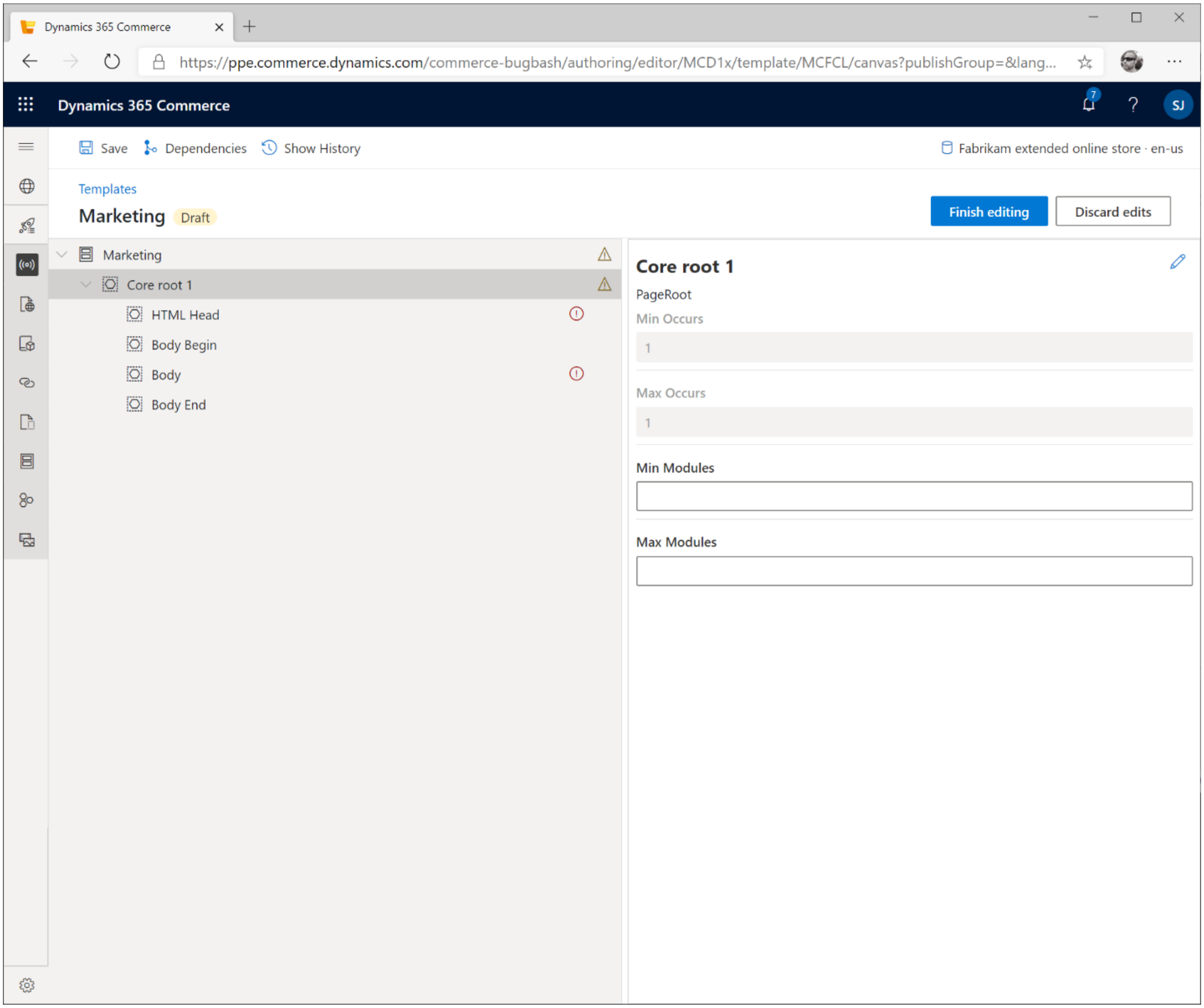
Template Description

Open template with header and footer fragments

The template page editor should now appear, notice how a tree view of the page shows up in the left column and the configurations in the right column similar to the fragment editor, but there is no rendered view in the middle. This is because a template isn't binding and data to it, but just provides the set of rules for a page.

Notice the layout module has some pre-defined slots (similar to slots we saw in the Header Fragment earlier)

The "HTML Head", "Body Begin" and "Body End" are special slots for a few of the modules including meta data tags and scripting. The "Body" slot is used for our main site contents.



https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/template/MCFCL/canvas?publishGroup=&lang...

Marketing Draft

- Marketing
  - Core root 1
    - HTML Head
    - Body Begin
    - Body
    - Body End

Core root 1

PageRoot

Min Occurs: 1

Max Occurs: 1

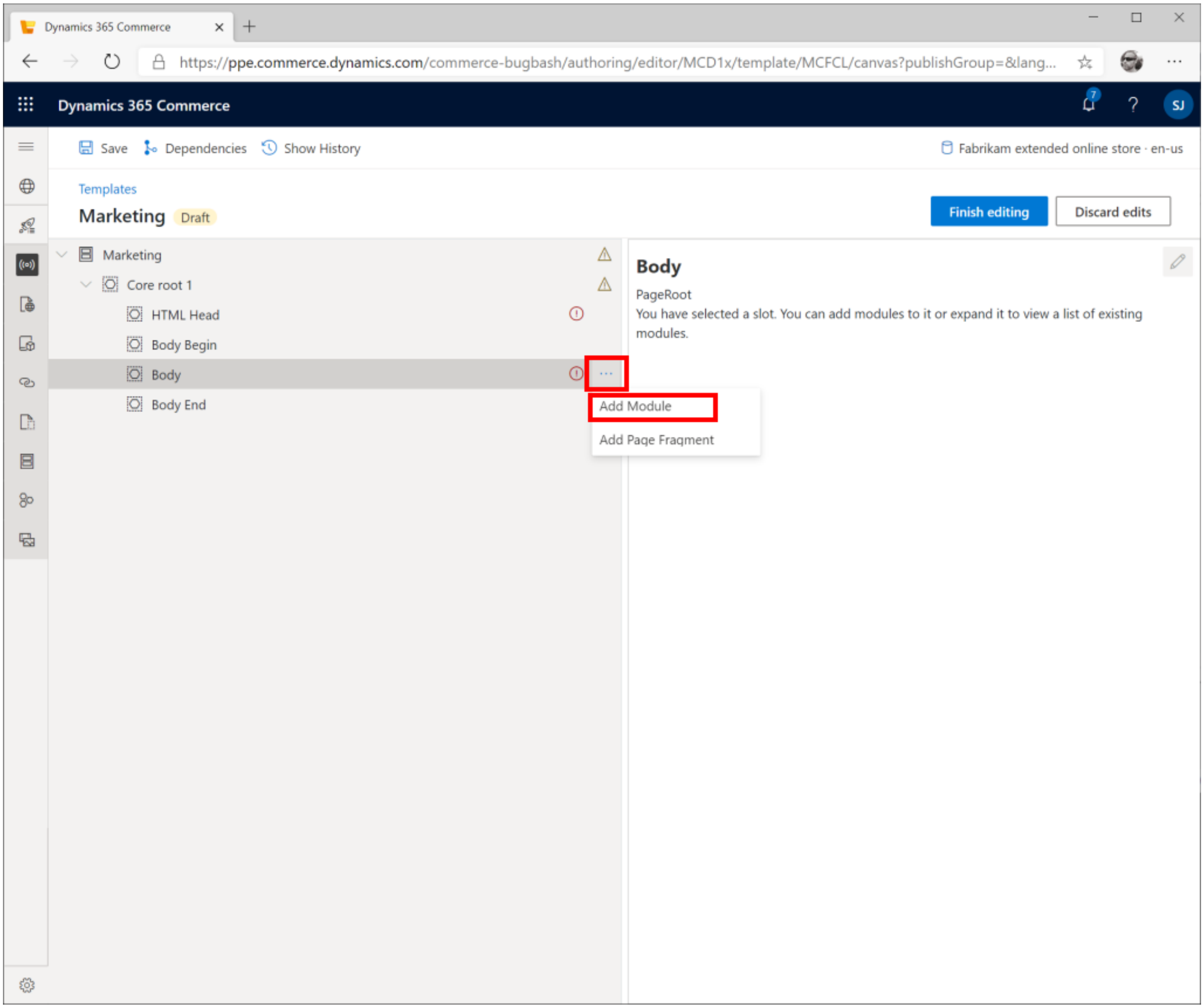
Min Modules:

Max Modules:

ADD A PAGE MODULE

All pages must have a root Page module, it is a special module that defines the page layout. The page module must be added to "Body" slot.

Select "Add Module" from the Body slot "…" menu as shown below.

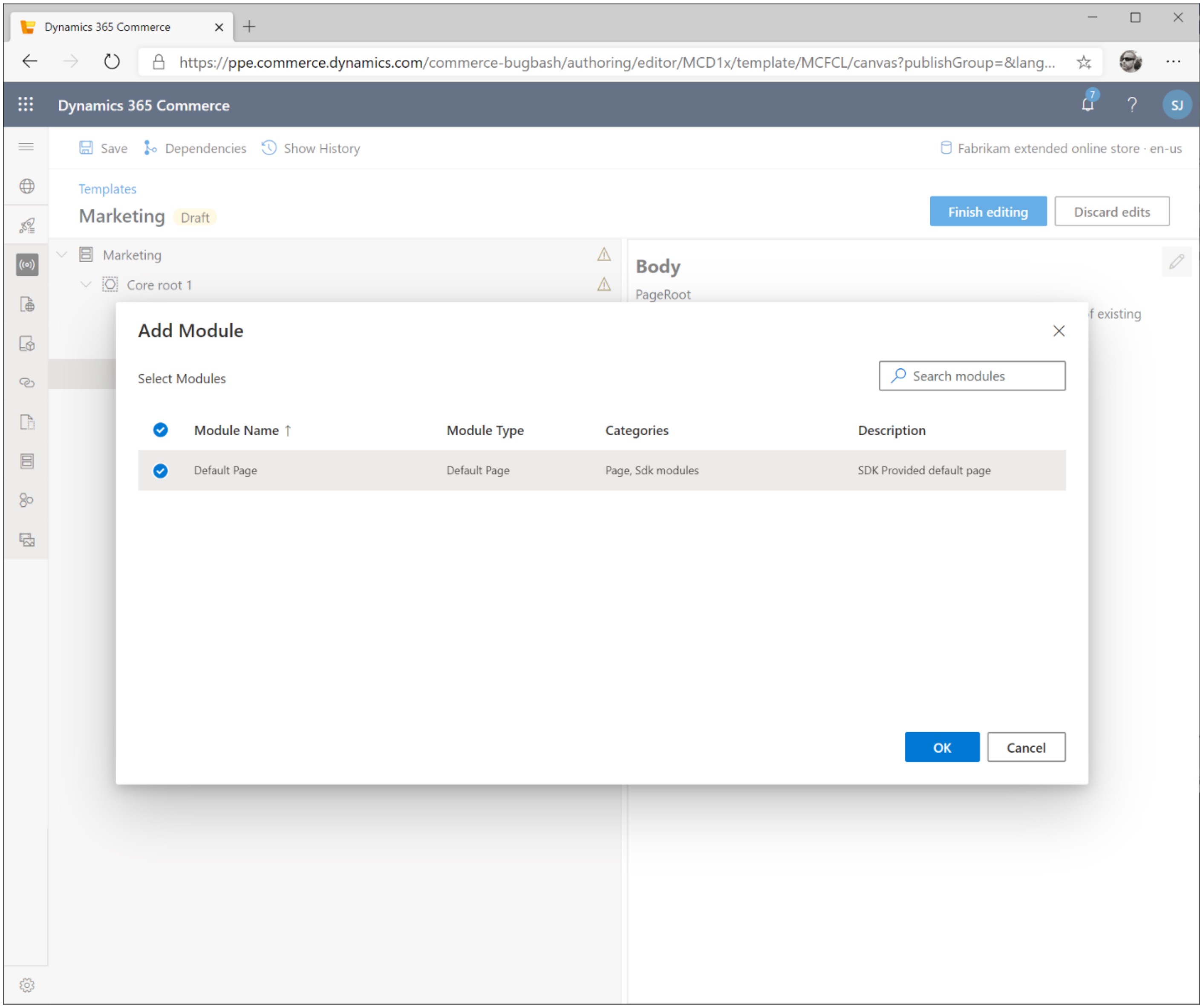


Body

PageRoot

You have selected a slot. You can add modules to it or expand it to view a list of existing modules.

The module picker will now only show "Page" modules. Select the "Default Page" module and click "OK".



Add Module

Select Modules

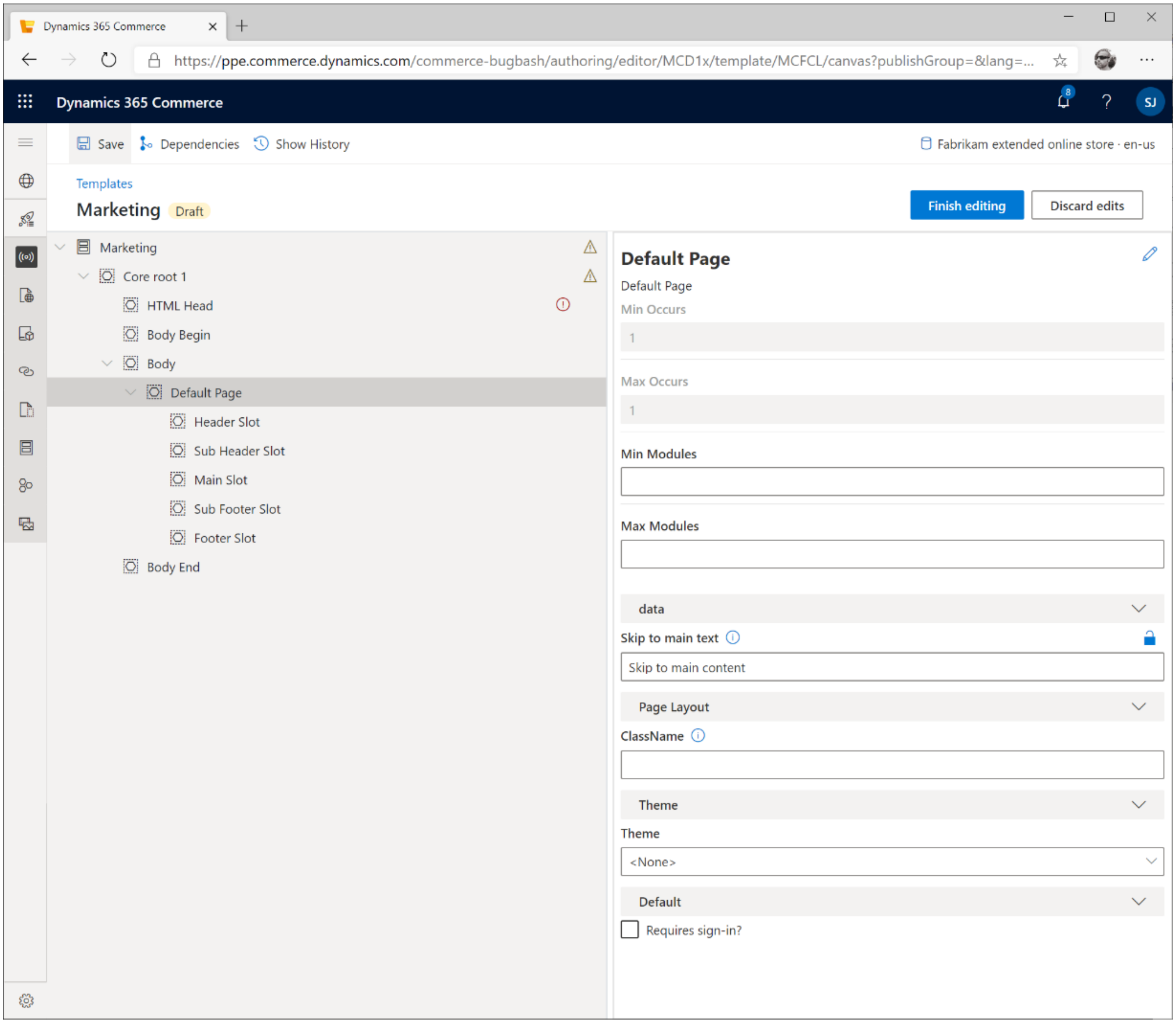
Module Name ↑	Module Type	Categories	Description
Default Page	Default Page	Page, Sdk modules	SDK Provided default page

The “Body” slot should now contain a “Default Page” module, which itself has several slots defined including:

- “Header Slot”
- “Sub Header Slot”
- “Main Slot”
- “Sub Footer Slot”
- “Footer Slot”

We’ll be placing our header fragment in the “Header Slot”, our main page modules will go inside the “Main Slot” and the “Footer Slot” will contain our Footer Fragment. Additionally pages could use the “Sub Header Slot” and “Sub Footer Slot” to render modules between the main content and the Header or Footer respectively.

We won’t change any of the configuration settings on the “Default Page” but notice if we wanted pages created from this template to use a different “Theme” one can be set here, otherwise the default will be the one set on the site configurations, set earlier in this guide.



Default Page

Default Page

Min Occurs: 1

Max Occurs: 1

Min Modules:

Max Modules:

data

Skip to main text: Skip to main content

Page Layout

ClassName:

Theme

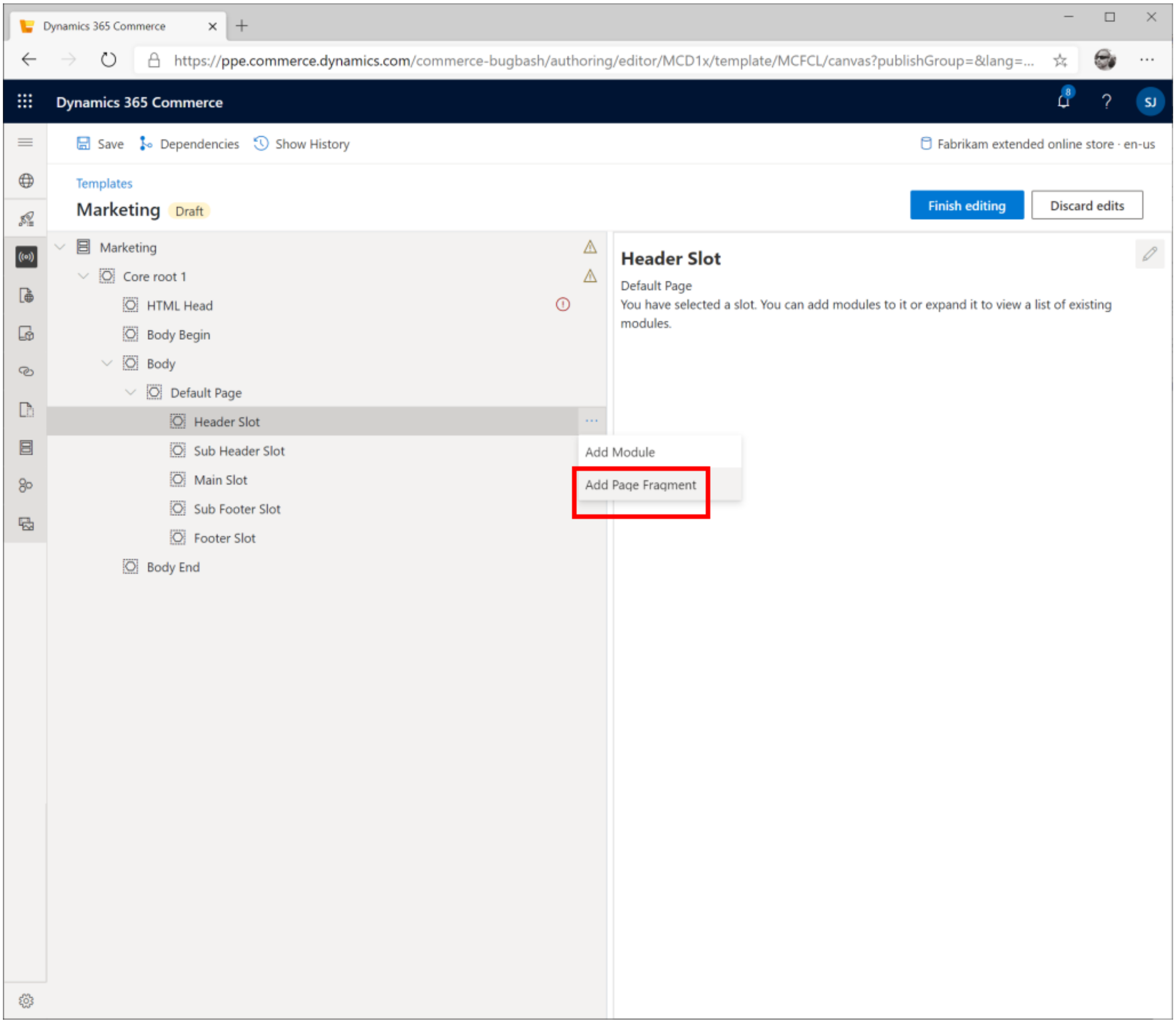
Theme: <None>

Default

□ Requires sign-in?

ADD HEADER FRAGMENT

Now we will add the previously created Header fragment to the template. Select the "…" menu on the "Header Slot" and select "Add Fragment" to bring up the Fragment Picker.

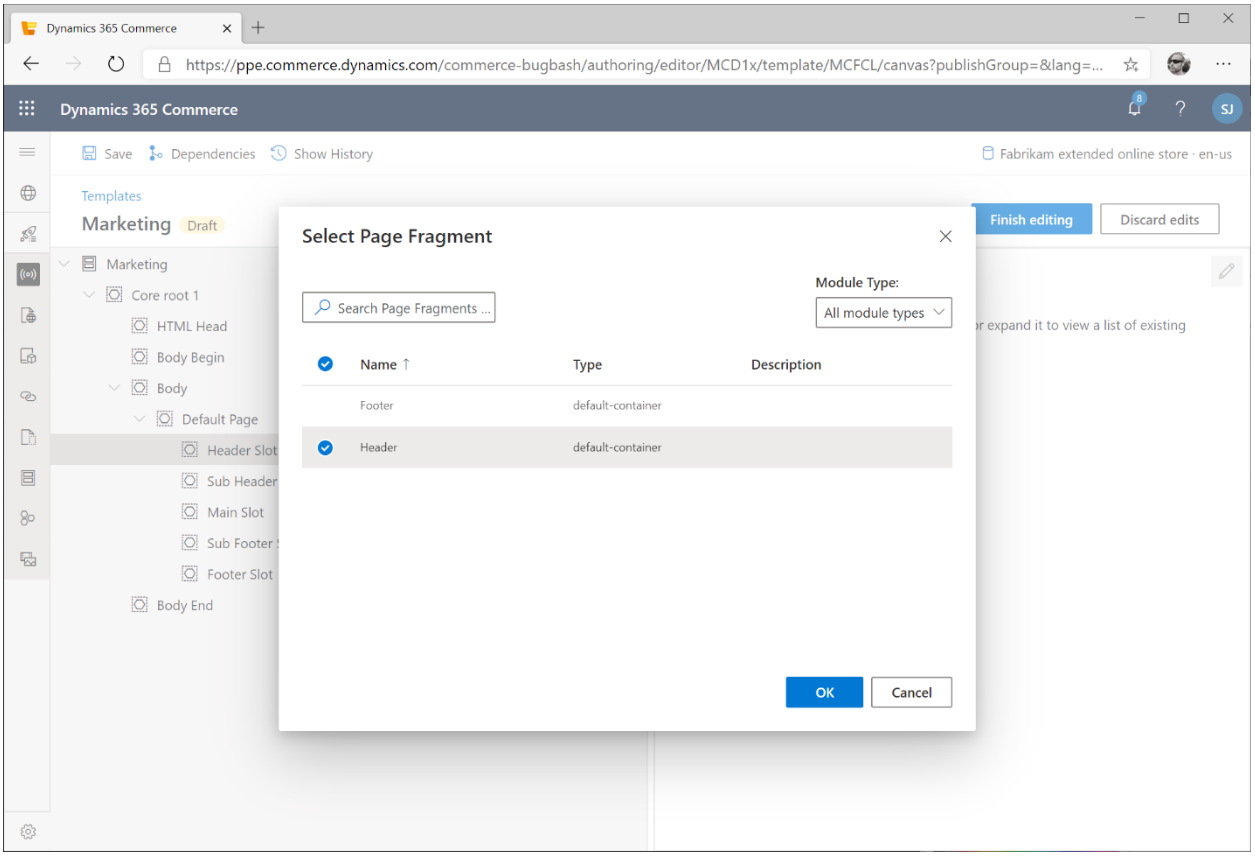


Header Slot

Default Page

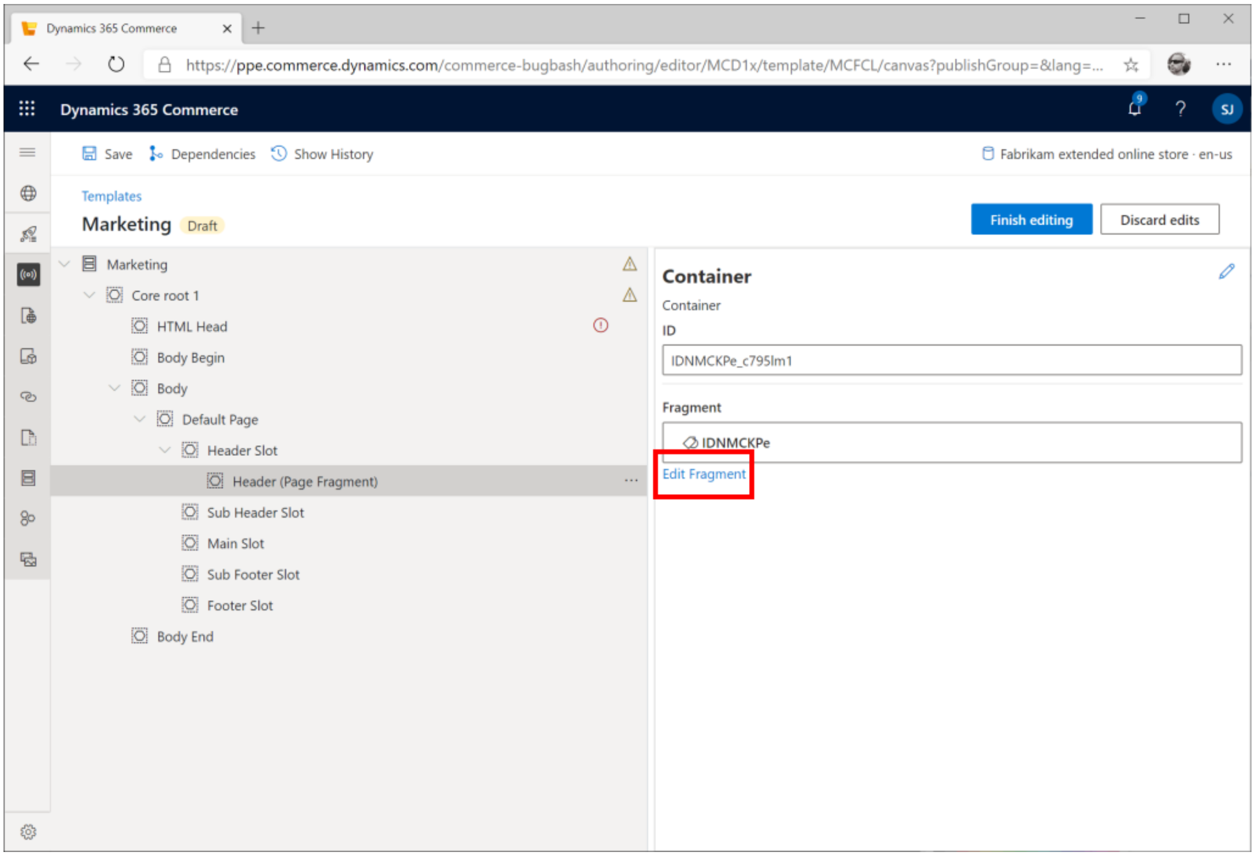
You have selected a slot. You can add modules to it or expand it to view a list of existing modules.

Select the Header Fragment and press the “Select” button.



	Name ↑	Type	Description
	Footer	default-container	
✓	Header	default-container	

Notice the Header fragment configuration properties cannot be modified in the template. Since it’s a fragment, it can only be edited in the fragment editor. There is a link to the editor if you select the “Edit Fragment” link, which will open up in another tab.



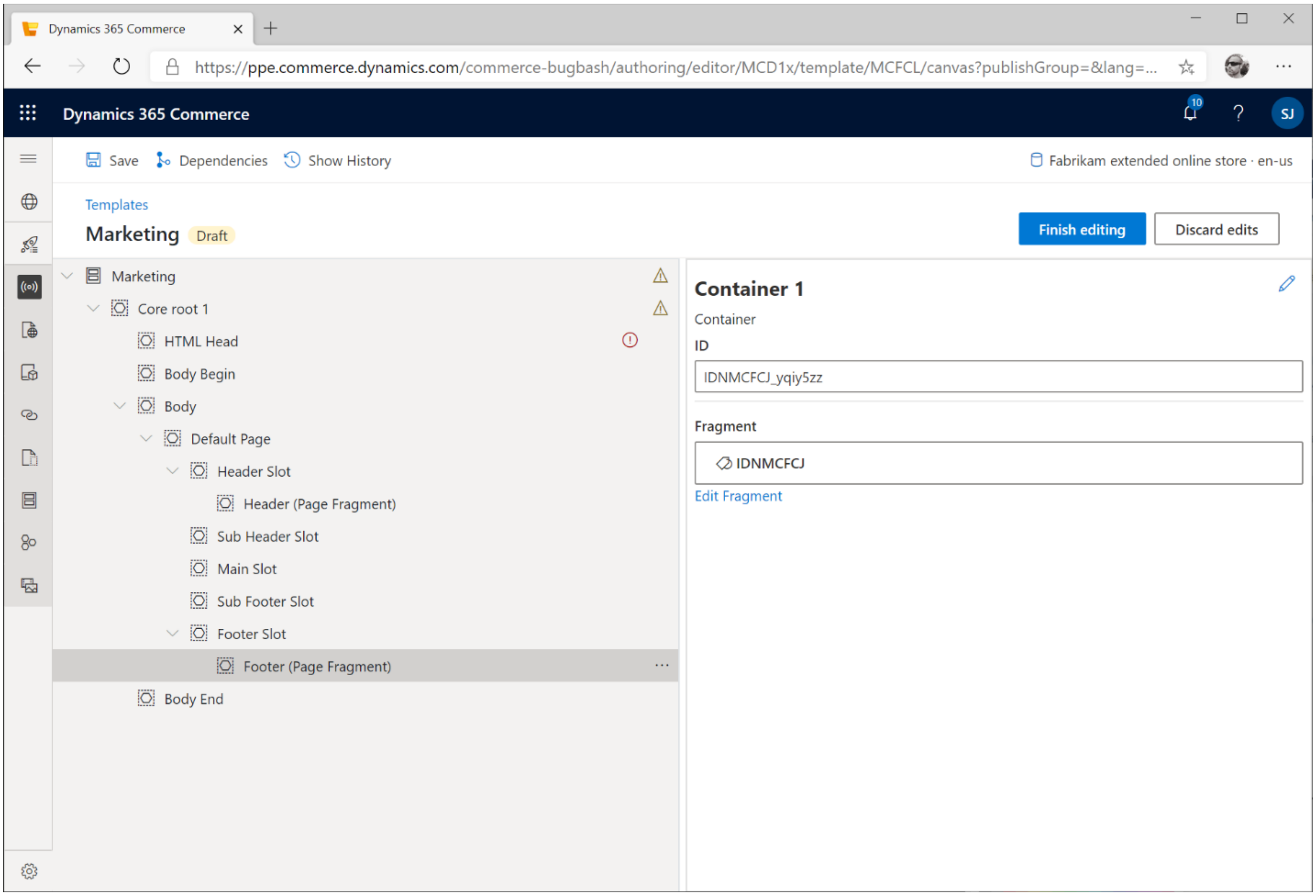
https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/template/MCFCL/canvas?publishGroup=&lang=...

ID: IDNMCKPe_c795lm1

Fragment: IDNMCKPe

ADD FOOTER FRAGMENT

Similar to how the header fragment was added above, add the Footer fragment to the “Footer Slot”.



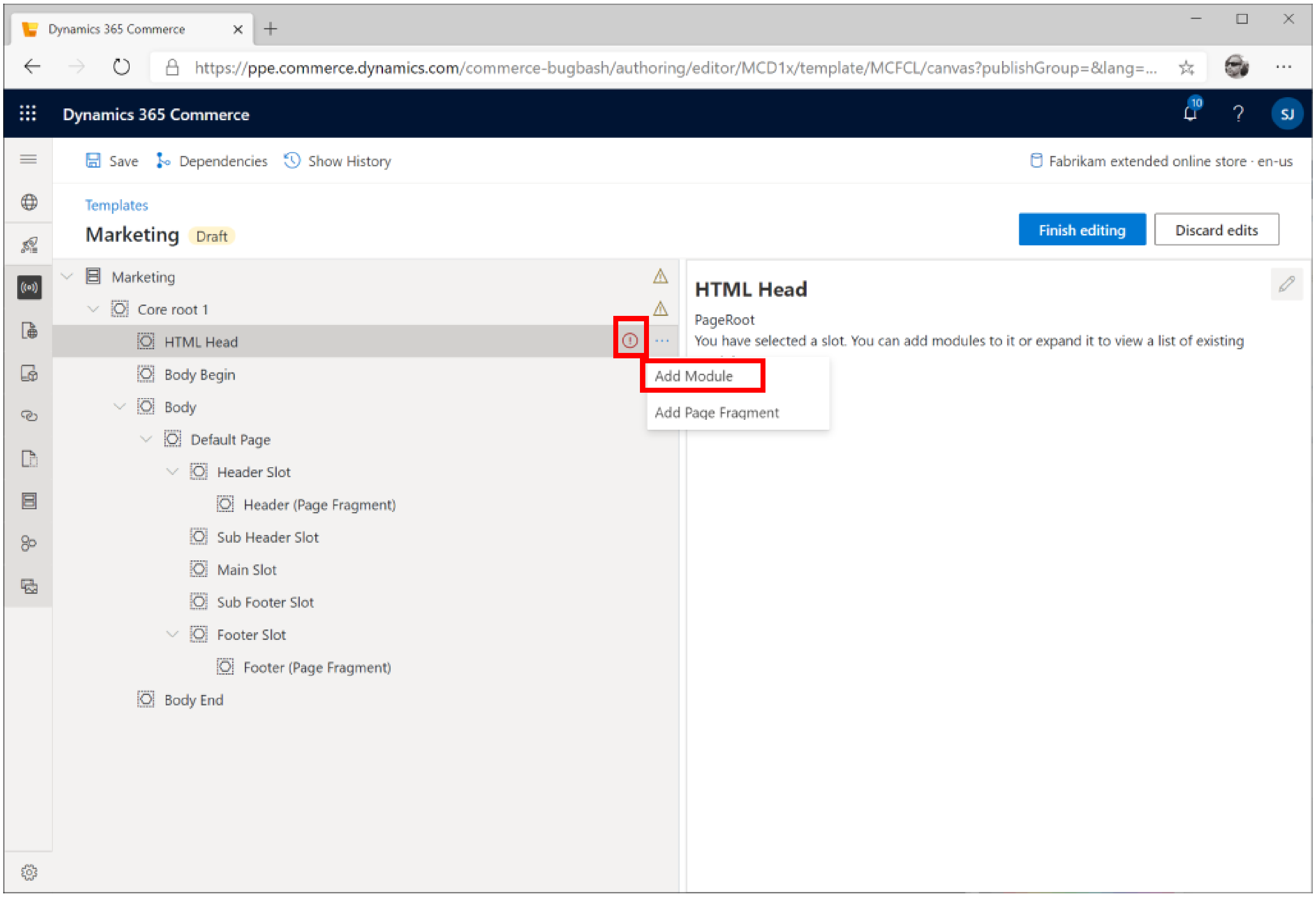
ID: IDNMCFCJ_yqiy5zz

Fragment: IDNMCFCJ

ADD HTML HEAD MODULES

The HTML Head slot can contain some special modules for meta data, scripts (ex: Google Analytics) and a special page summary module.

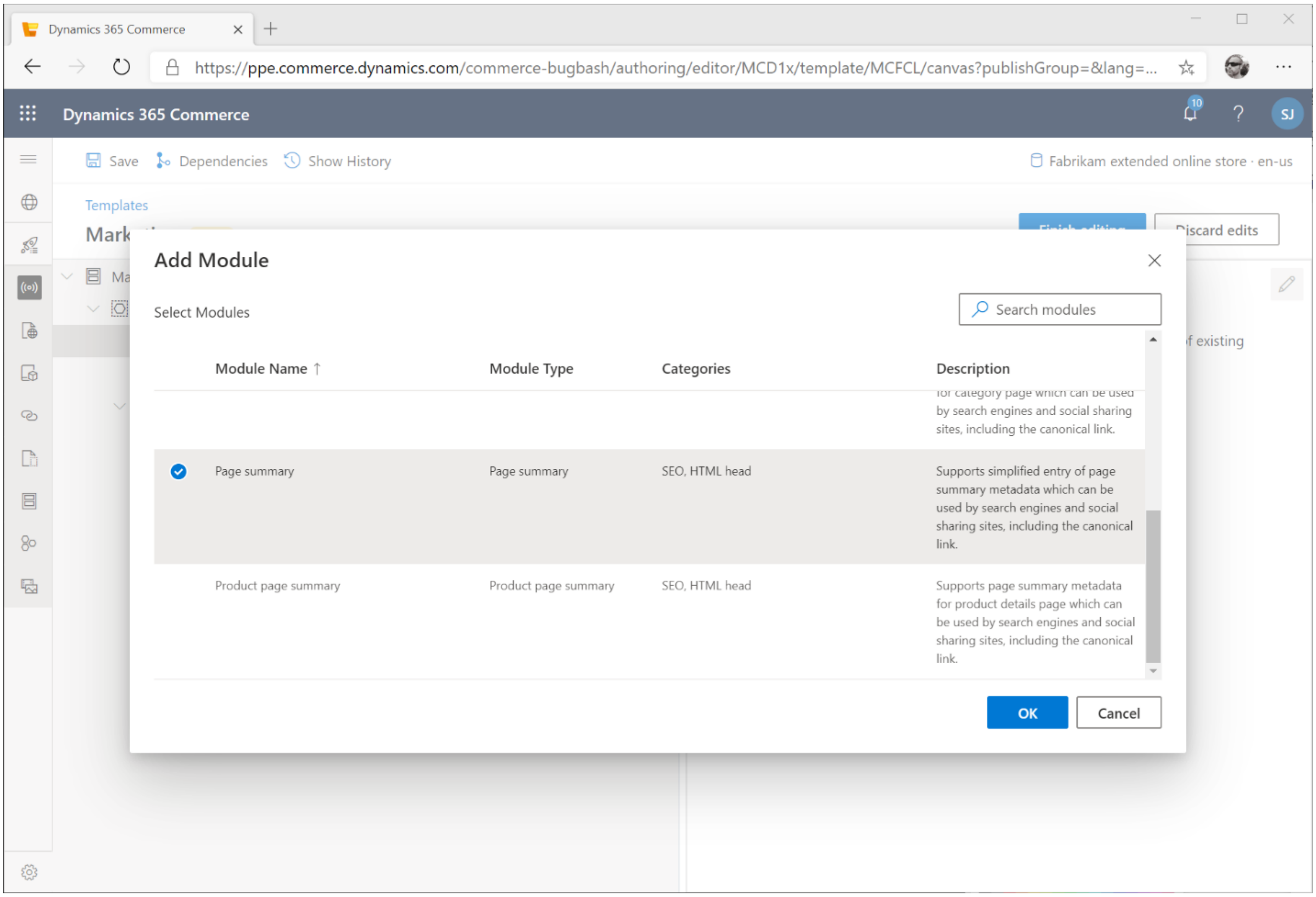
We will only be adding a default page summary module, others can be added later if required. Also, notice the warning icon on the HTML Head states that at least one module must be added, which will be satisfied with this step.



You have selected a slot. You can add modules to it or expand it to view a list of existing

Select the “**HTML Head**” slot followed by “…” and “**Add Module**” to bring up the module picker. Select the “P**age summary**” module. By adding this module, each page that derives from this layout will be able to set extra configurations on each page including a favicon and search keywords.

Save the template when complete.


Dynamics 365 Commerce
https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/template/MCFCL/canvas?publishGroup=&lang=...
Dynamics 365 Commerce
10
SJ
Save
Dependencies
Show History
Fabrikam extended online store · en-us
Templates
Mark
Finish editing
Discard edits
Ma
f existing

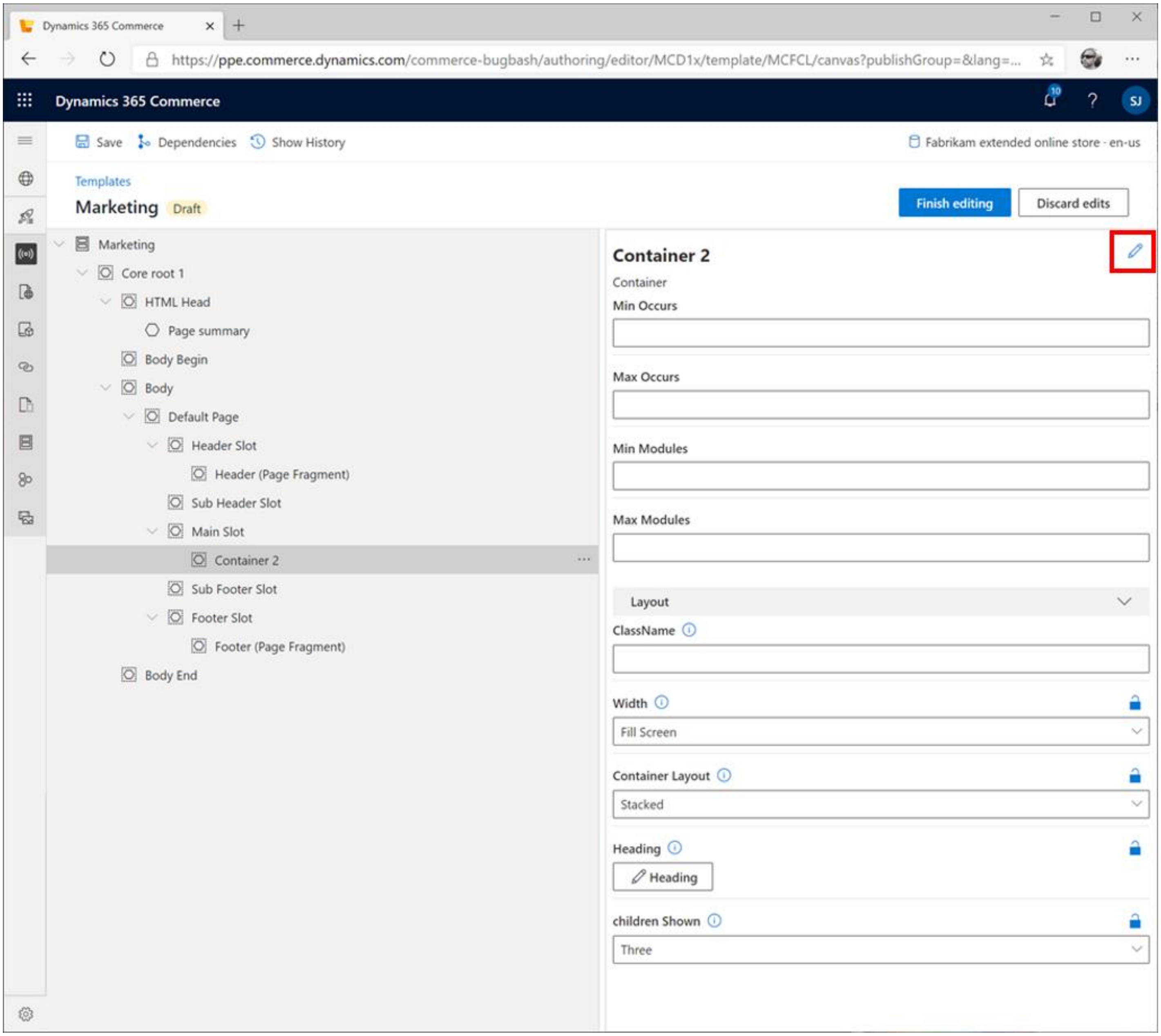
Add Module

Select Modules

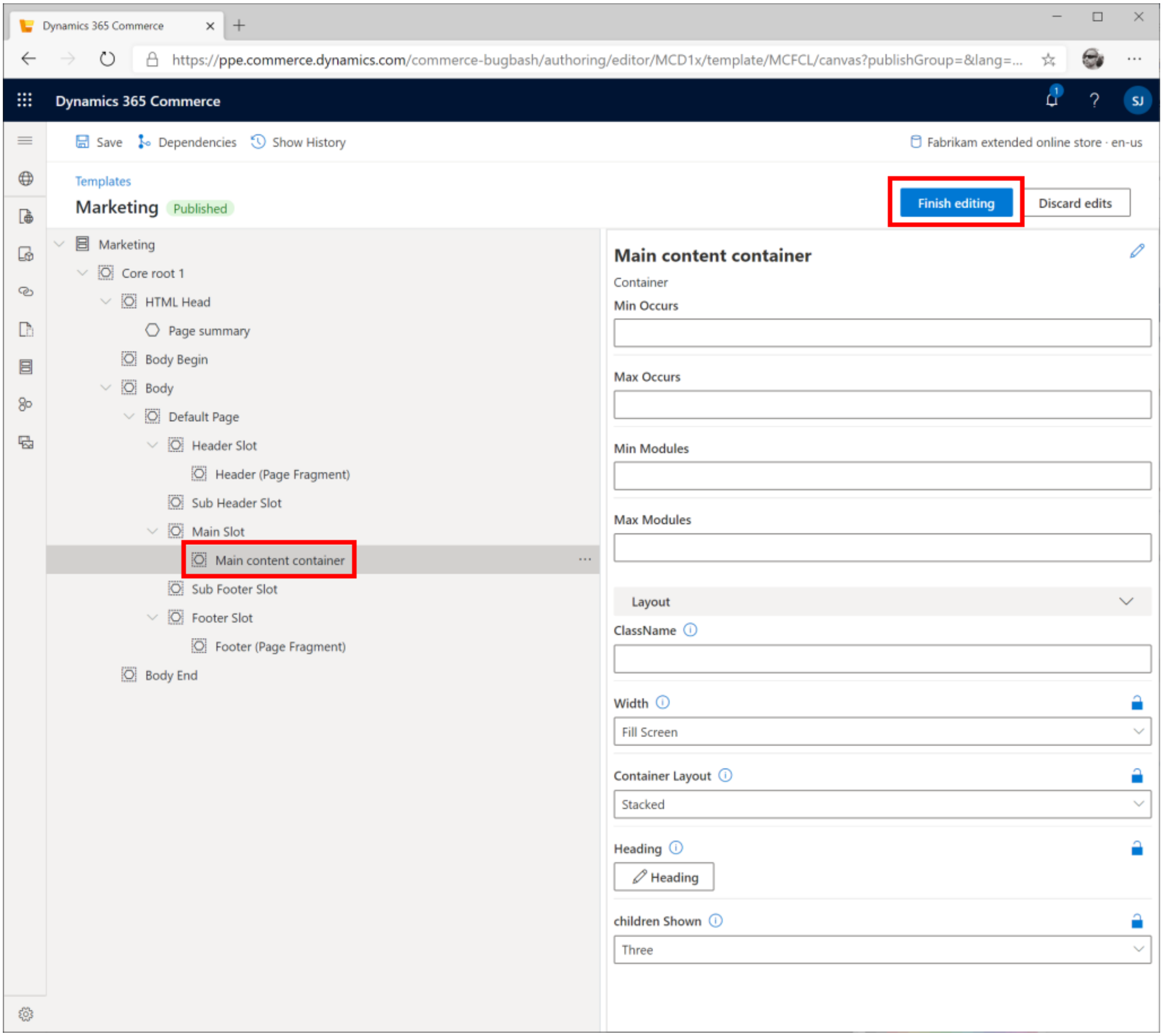
Search modules

	Module Name ↑	Module Type	Categories	Description
				for category page which can be used by search engines and social sharing sites, including the canonical link.
✓	Page summary	Page summary	SEO, HTML head	Supports simplified entry of page summary metadata which can be used by search engines and social sharing sites, including the canonical link.
	Product page summary	Product page summary	SEO, HTML head	Supports page summary metadata for product details page which can be used by search engines and social sharing sites, including the canonical link.

Last, we'll add a "container" slot to the "Main Slot", pages will use this slot to add their content modules. We can then also change the friendly name that shows up by selecting the "edit" button in the configuration properties panel on the right.



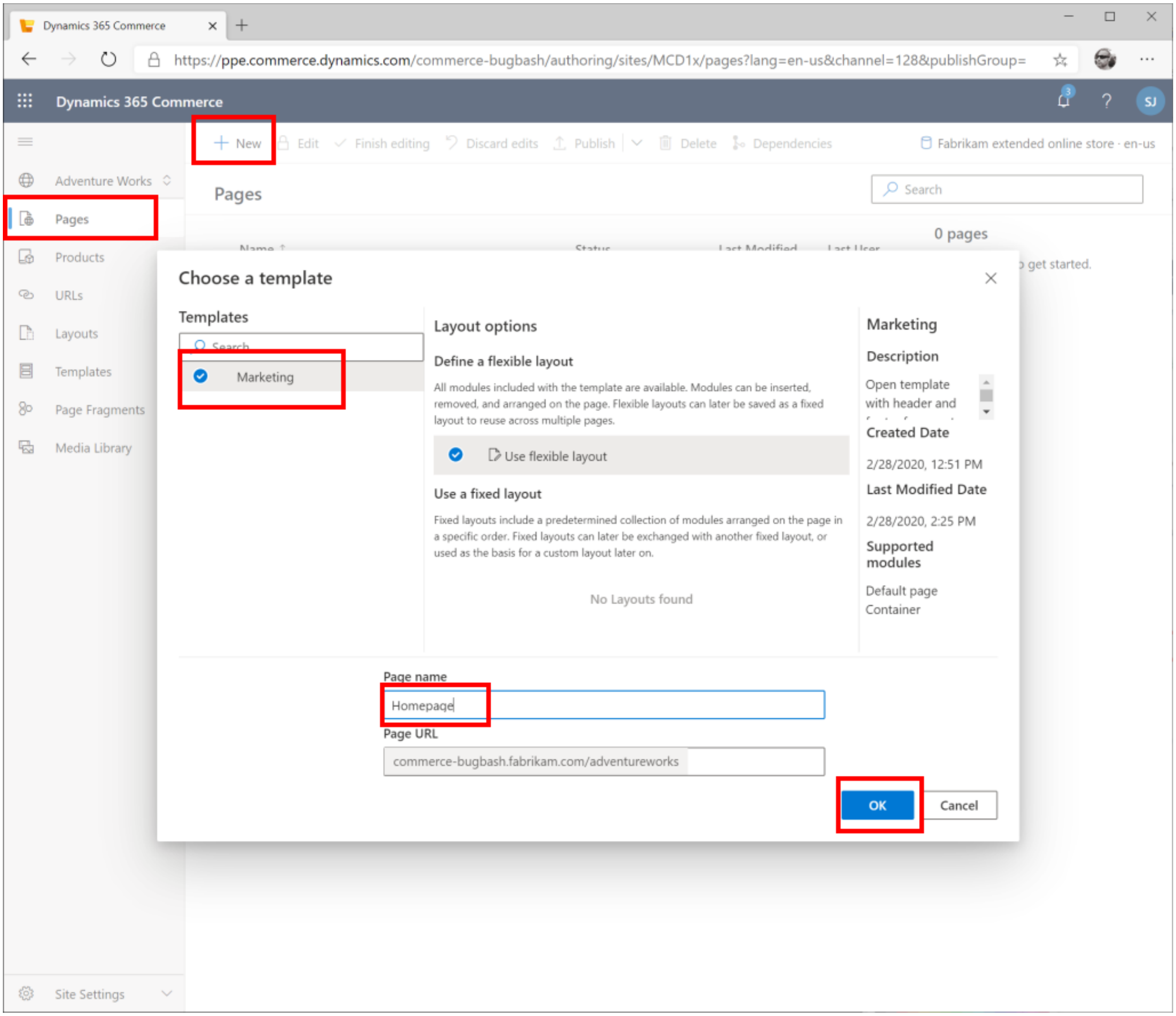
Below you can see the container name changed to “Main Content Container”. We are now complete with our Template page. Click the “Finish editing” button followed by “Publish”.



CREATE A HOME PAGE

We are now ready to create our Home Page for our site. Navigate back to our site by selecting the site name in the top breadcrumb.

Next select the “**Pages**” tab in the left pane, followed by “**New Page**” to bring up the New Page dialog box. Select the previously created template “Marketing” and provide a “Page name” such as “Homepage”.



Define a flexible layout

All modules included with the template are available. Modules can be inserted, removed, and arranged on the page. Flexible layouts can later be saved as a fixed layout to reuse across multiple pages.

Use a fixed layout

Fixed layouts include a predetermined collection of modules arranged on the page in a specific order. Fixed layouts can later be exchanged with another fixed layout, or used as the basis for a custom layout later on.

Description: Open template with header and

Created Date: 2/28/2020, 12:51 PM

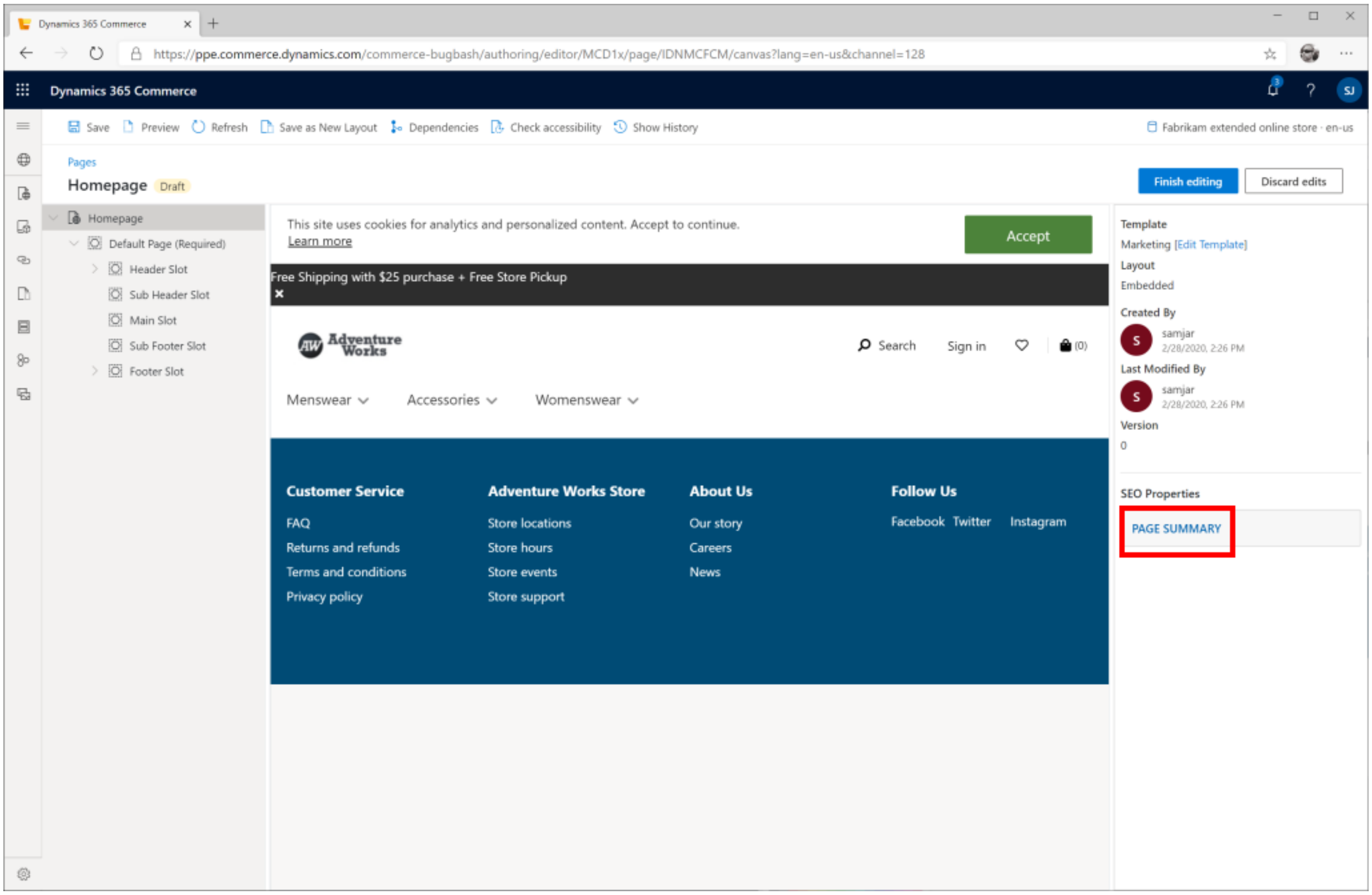
Last Modified Date: 2/28/2020, 2:25 PM

Supported modules: Default page Container

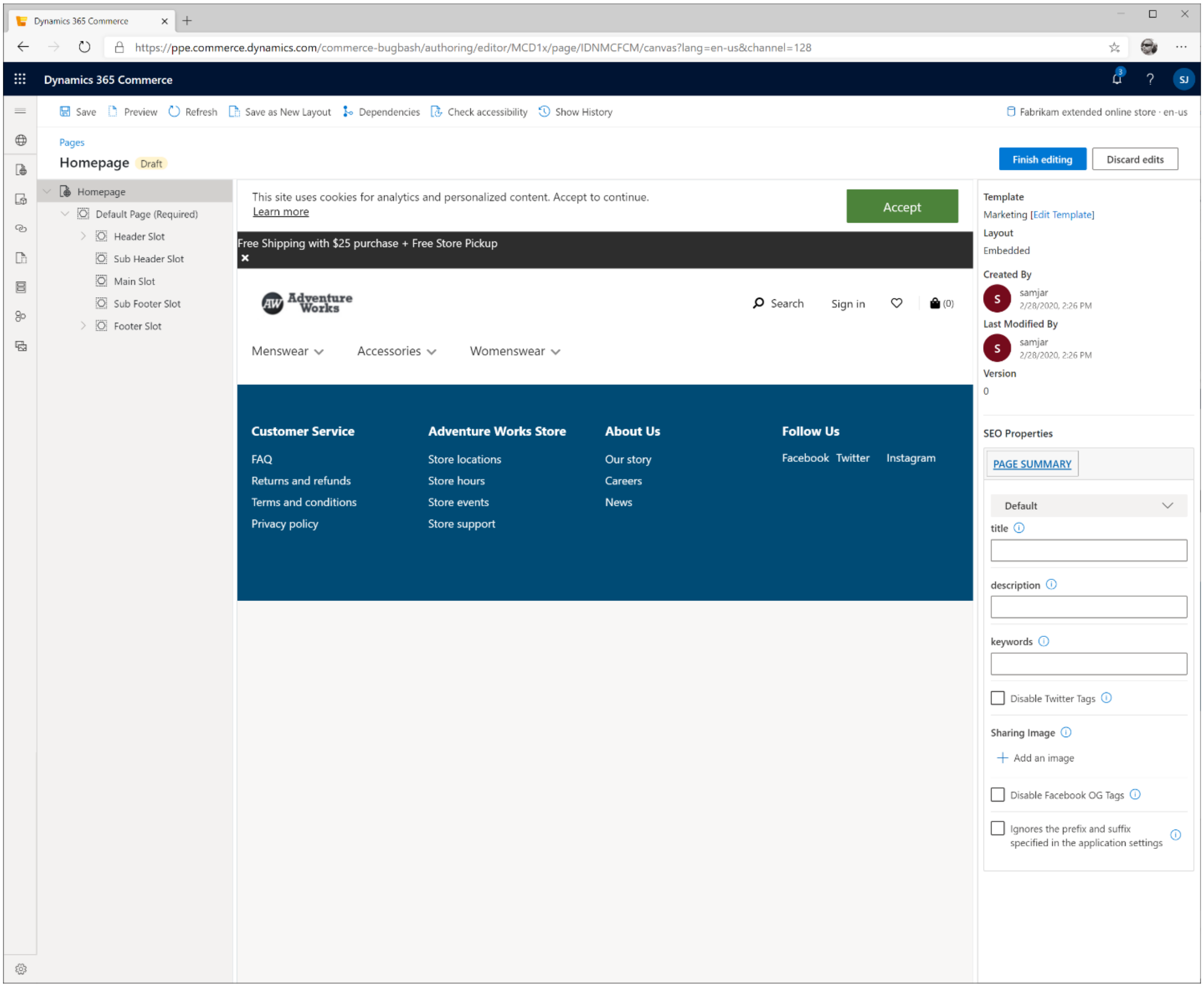
Page name: Homepage

Page URL: commerce-bugbash.fabrikam.com/adventureworks

You should now see the page rendered in the middle pane with our header and footer. Notice if you select the root node, there is a link to “DEFAULT PAGE SUMMARY”, select the link.



You should now see the same set of properties we saw on the template we created earlier. Here these configurations can be changes for this page. For example notice the favicon URL that was set in the layout page is set here as the default. It can be overridden if desired.



Template
Marketing [Edit Template]

Layout
Embedded

Created By
samjar
2/28/2020, 2:26 PM

Last Modified By
samjar
2/28/2020, 2:26 PM

Version
0

SEO Properties

PAGE SUMMARY

Default

title

description

keywords

□ Disable Twitter Tags

Sharing Image

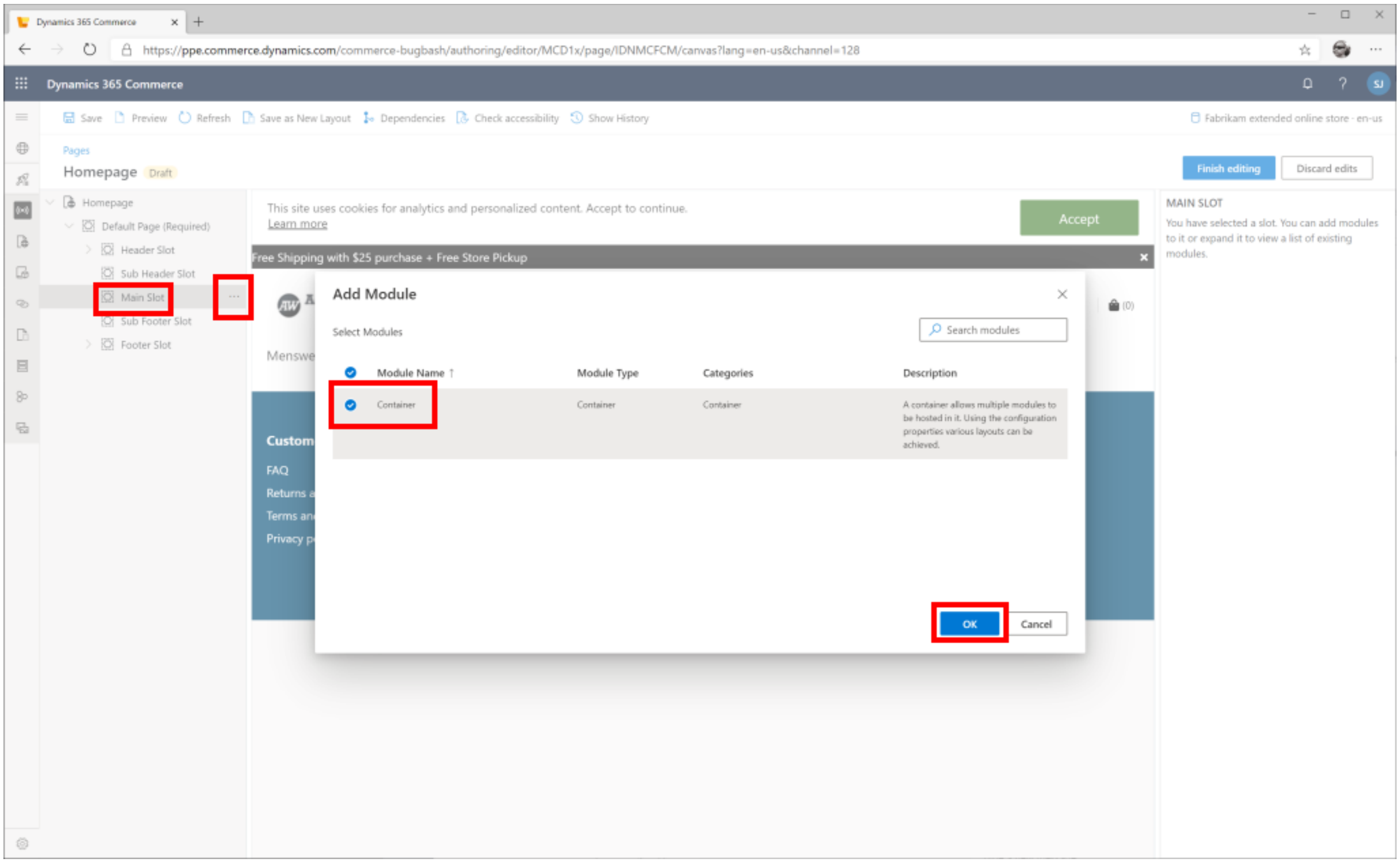
+ Add an image

□ Disable Facebook OG Tags

□ Ignores the prefix and suffix specified in the application settings

ADD CONTENT MODULES TO PAGE

Select the "**Main Slot**" followed by "**…**" and "**Add Module**" to bring up the module picker. You should now see the single module "**Container**" listed which the template defined as an allowed child of the "Main slot". Select the module and click the "OK" button.



This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Free Shipping with $25 purchase + Free Store Pickup

MAIN SLOT

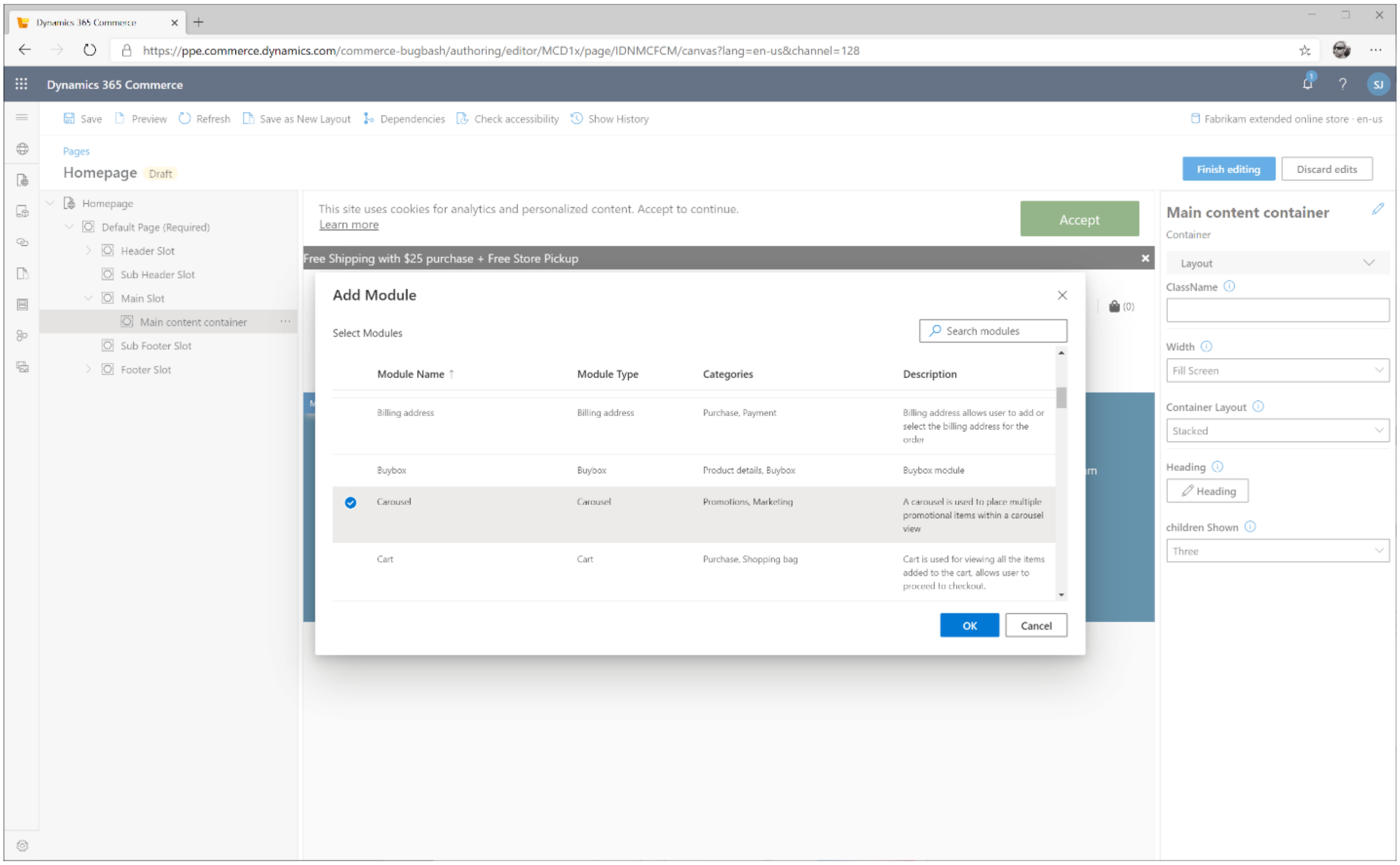
You have selected a slot. You can add modules to it or expand it to view a list of existing modules.

Add Module

Select Modules

	Module Name ↑	Module Type	Categories	Description
●	Container	Container	Container	A container allows multiple modules to be hosted in it. Using the configuration properties various layouts can be achieved.

Select the new “**Main content container**” module followed by “**…**” and “**Add Module**” to bring up the module picker. You should now see a full list of available modules. Pick the “**Carousel**” container module and select the “OK” button. We’ll leave all configurations at there defaults.



https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/page/IDNMCFCM/canvas?lang=en-us&channel=128

This site uses cookies for analytics and personalized content. Accept to continue.

Free Shipping with $25 purchase + Free Store Pickup

Add Module

Select Modules

	Module Name	Module Type	Categories	Description
	Billing address	Billing address	Purchase, Payment	Billing address allows user to add or select the billing address for the order
	Buybox	Buybox	Product details, Buybox	Buybox module
✓	Carousel	Carousel	Promotions, Marketing	A carousel is used to place multiple promotional items within a carousel view
	Cart	Cart	Purchase, Shopping bag	Cart is used for viewing all the items added to the cart, allows user to proceed to checkout.

Main content container

Container

ClassName:

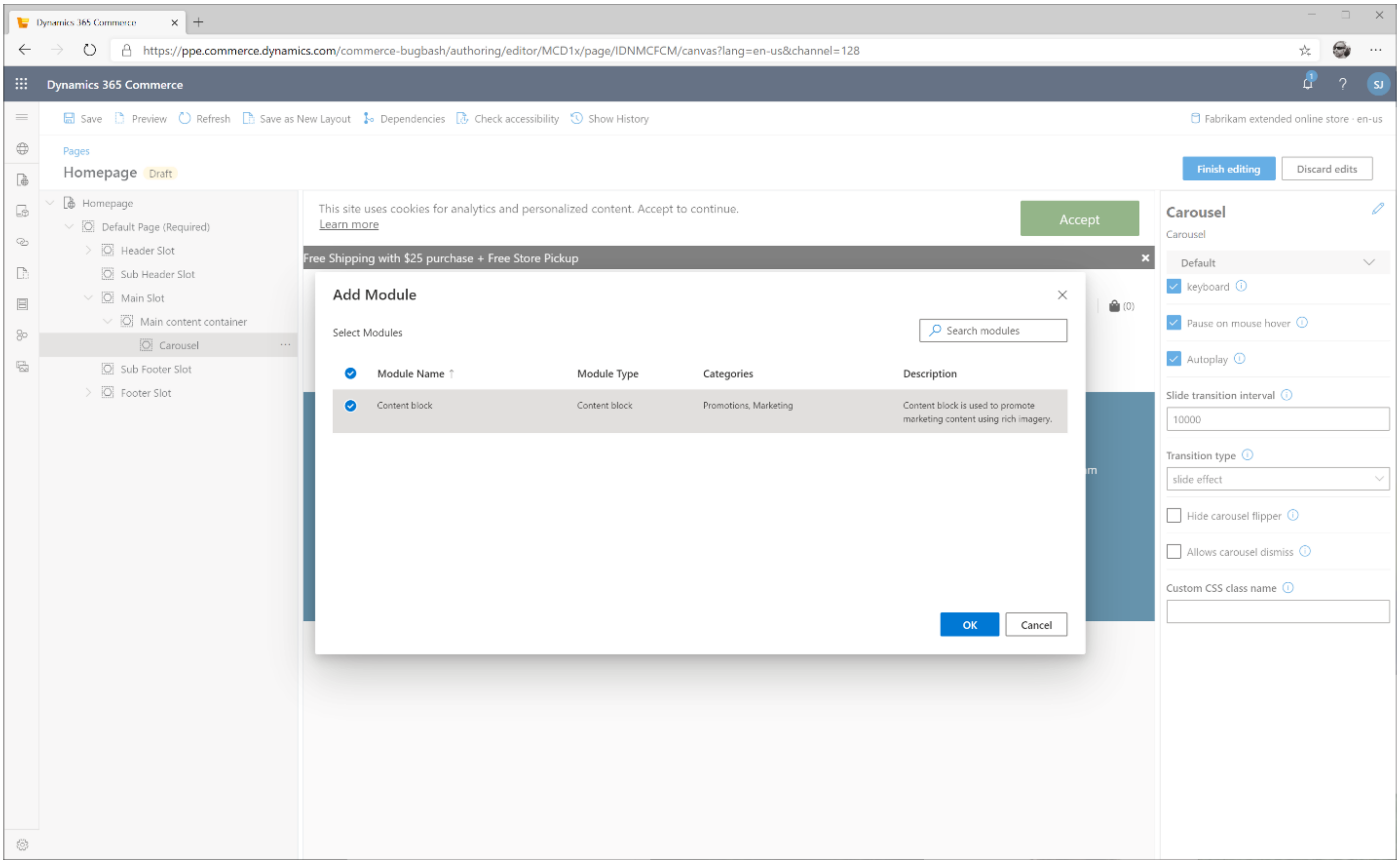
Width: Fill Screen

Container Layout: Stacked

children Shown: Three

Select the new Container module in the tree and add a new “Carousel” module.

We can now add individual modules that will rendered as individual carousel items. Select the “Carousel” module and add two “Content block” modules.



Add Module

Select Modules

Search modules

●	Module Name ↑	Module Type	Categories	Description
●	Content block	Content block	Promotions, Marketing	Content block is used to promote marketing content using rich imagery.

Carousel

Carousel

Default

☑ keyboard

☑ Pause on mouse hover

☑ Autoplay

Slide transition interval: 10000

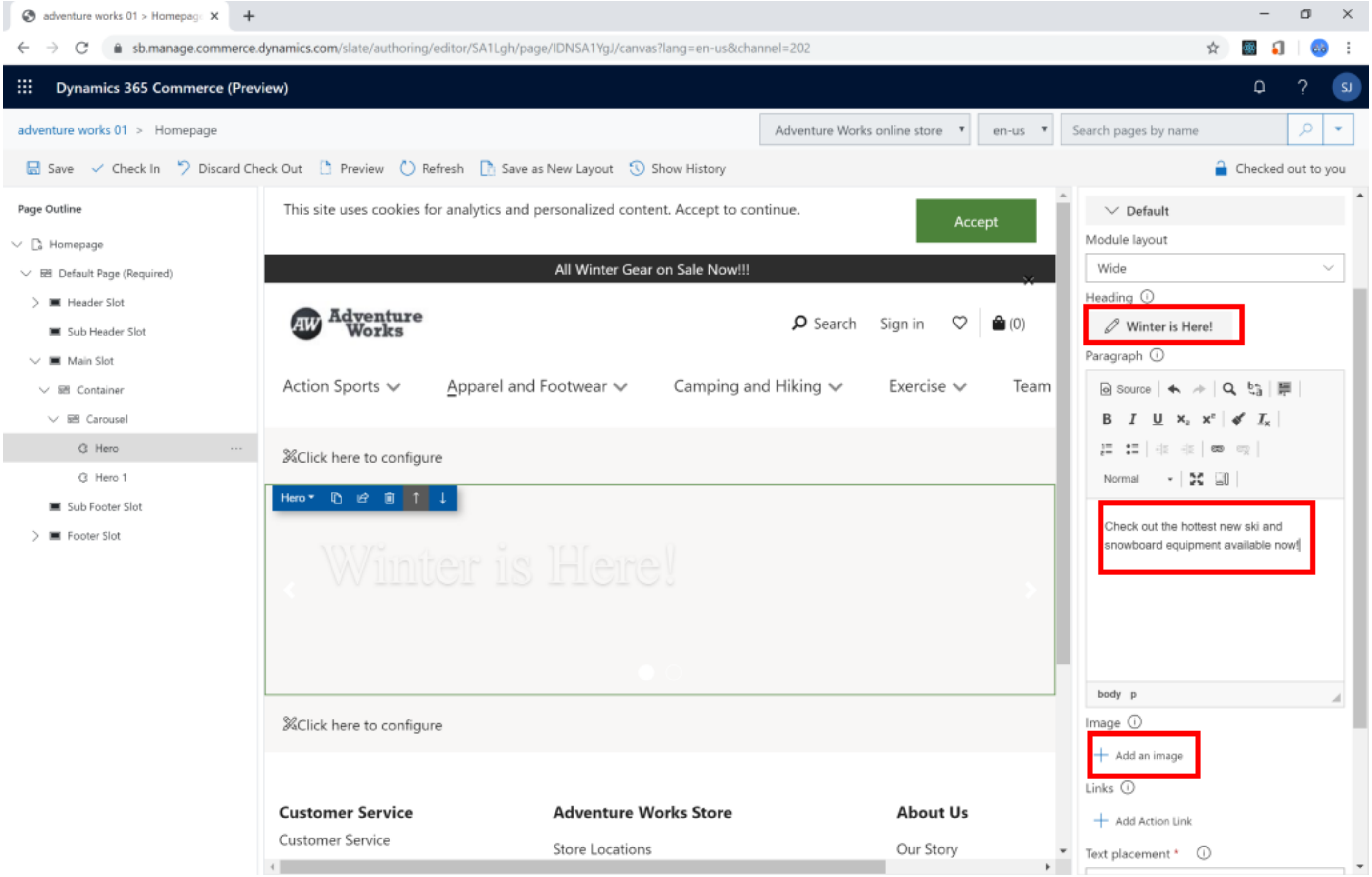
Transition type: slide effect

□ Hide carousel flipper

□ Allows carousel dismiss

Custom CSS class name

Configure the “Content block” modules. Select each and select “Module layout” = “Hero” and provide a “Heading” text Paragraph and select “Add an image” to bring up the image picker.



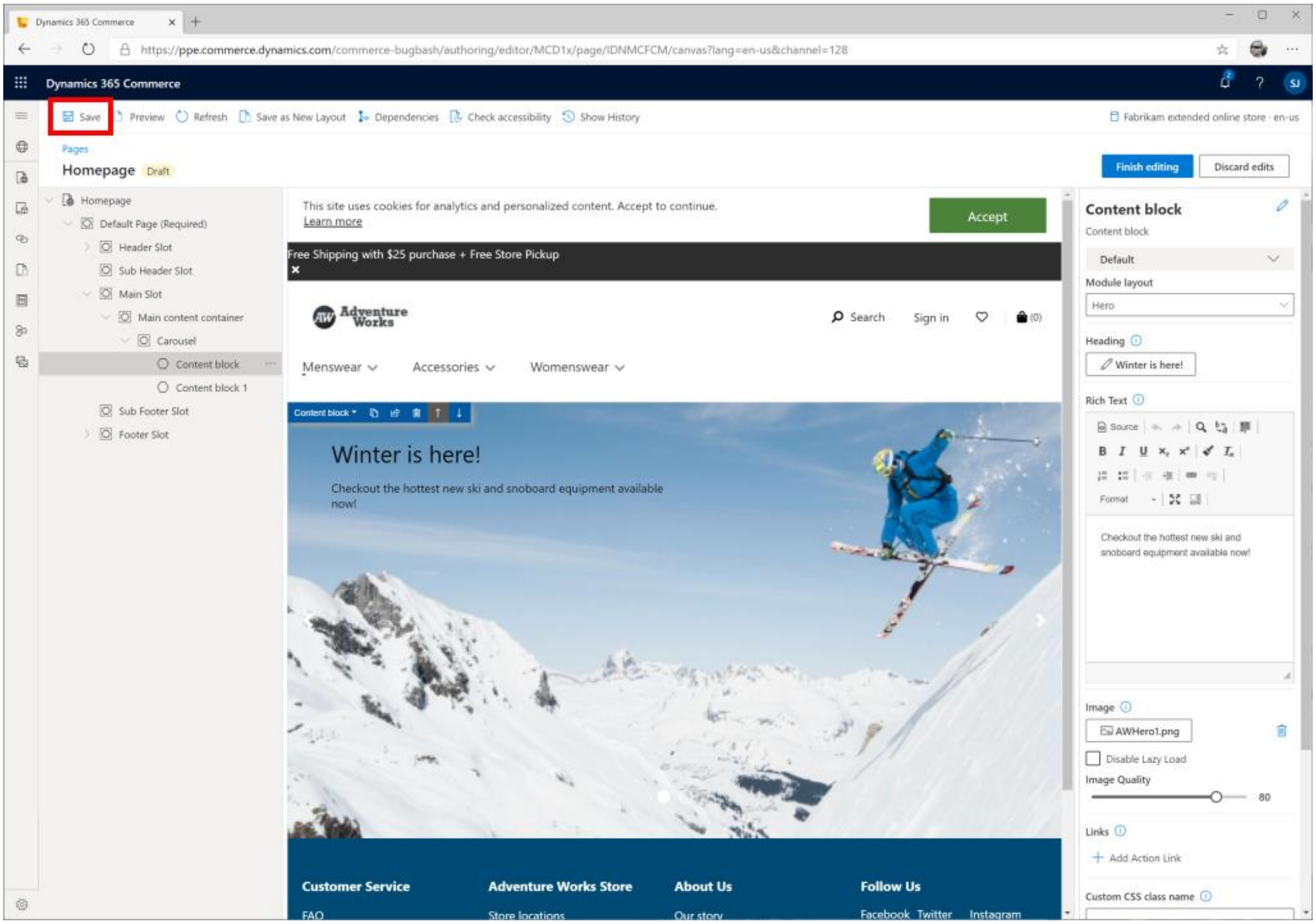
sb.manage.commerce.dynamics.com/slate/authoring/editor/SA1Lgh/page/IDNSA1YgJ/canvas?lang=en-us&channel=202

Module layout: Wide

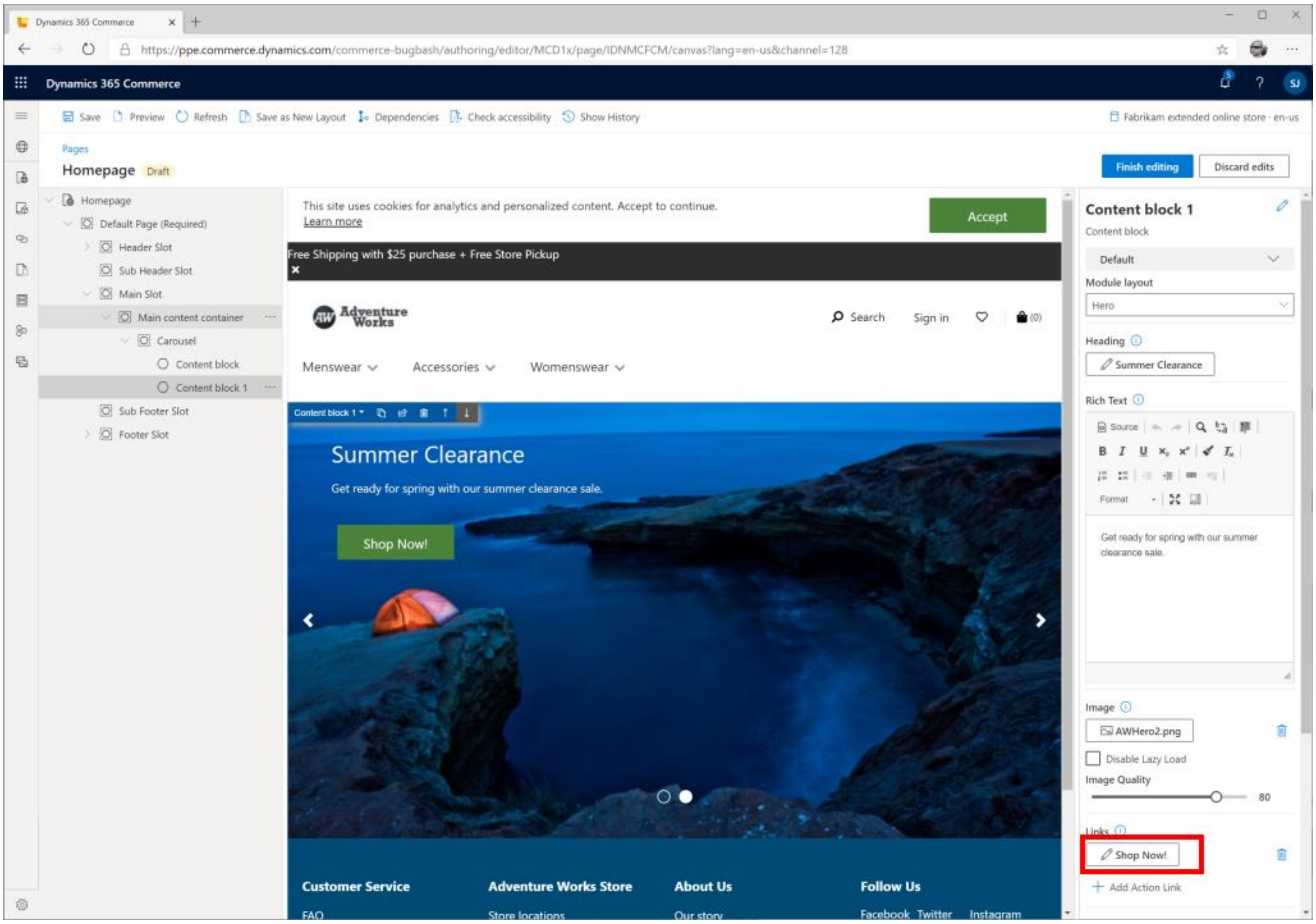
Heading: Winter is Here!

Paragraph: Check out the hottest new ski and snowboard equipment available now!

Upload an image, select it and click save to refresh the page.



Similarly configure the second Hero module. Notice below that an action link is also added, later this can be linked to a product category page.



https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/page/IDNMCFCM/canvas?lang=en-us&channel=128

Content block 1

Module layout: Hero

Heading: Summer Clearance

Rich Text: Get ready for spring with our summer clearance sale.

Image: AWHero2.png

□ Disable Lazy Load

Image Quality: 80

Links: Shop Now!

Next we'll add the remaining modules to make up our page. Add a new "**Container**" module to the "**Main Slot**" with the following children:

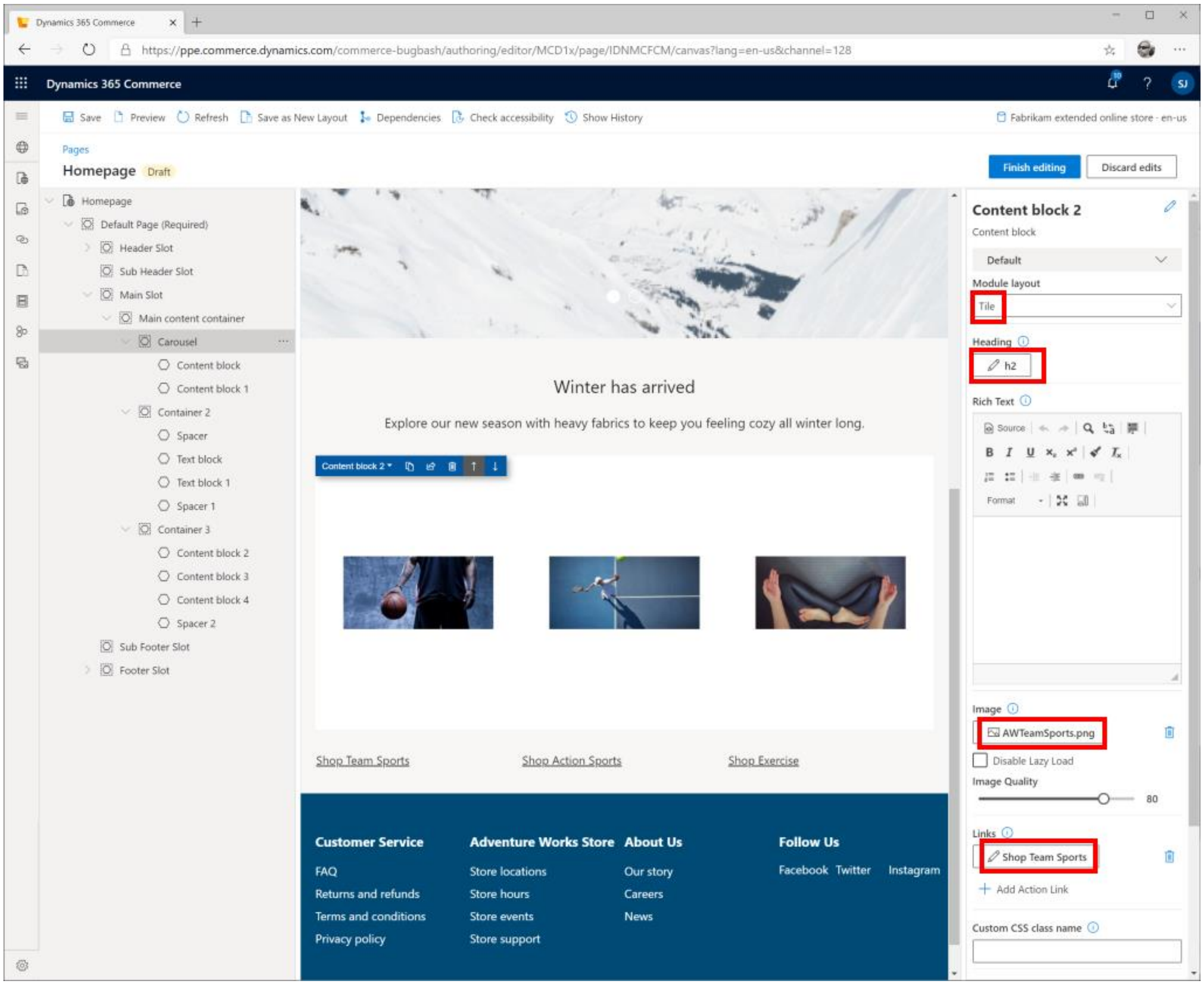
- "**Spacer**" with "Spacer height" set to 2x
- "**Text block**" with text "Winter has arrived", with "**Alignment**" = center and "**Font size**" = X-large
- "**Text block**" with text "Explore our new season with heavy fabrics to keep you feeling cozy all winter long.", with "**Alignment**" = center and "**Font size**" = Large
- "**Spacer**" with "Spacer height" set to 3x

Add another "**Container**" module to the "**Main Slot**" and set "**Width**" = "Fill Container" and "**Container Layout**" = "Flow".

Next add 3 "**Content block**" module children with the following configuration:

- "Module layout" = "Tile"
- Empty out the "Heading" text
- Upload an image for the module
- Set the "Tile Size" to 3
- Add an "Action Link" for each (just add the text, we can add a link later)

Last add a spacer module and set "Space height" = 3x.


https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/page/IDNMCFCM/canvas?lang=en-us&channel=128

Content block 2
Content block
Default
Module layout: Tile
Heading: h2
Image: AWTeamSports.png
Image Quality: 80
Links: Shop Team Sports

Winter has arrived

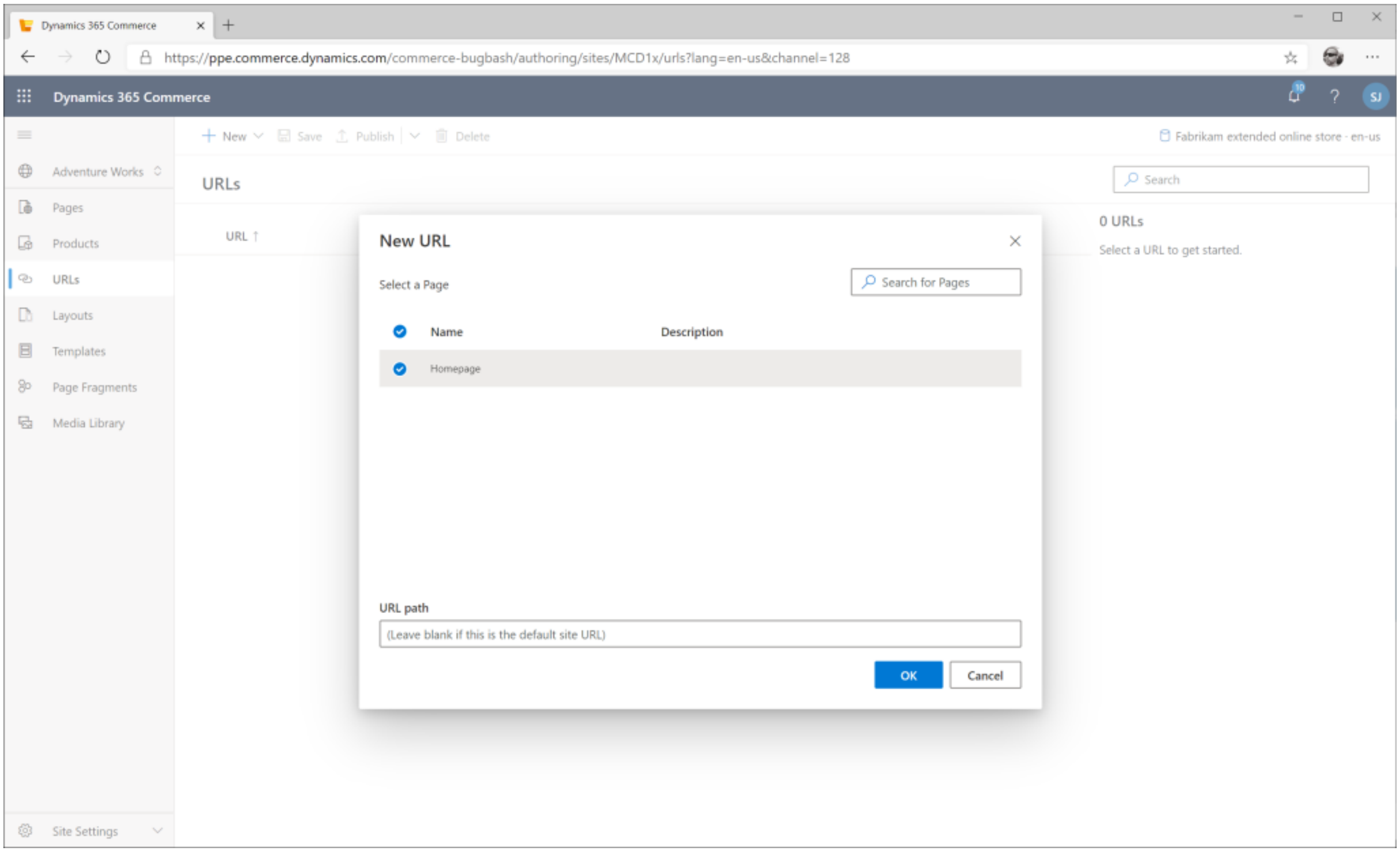
Explore our new season with heavy fabrics to keep you feeling cozy all winter long.

Once complete, select Save and “**Finish editing**”. Select “**Preview**” to preview the page. If you’re happy with changes, select “**Publish**”.

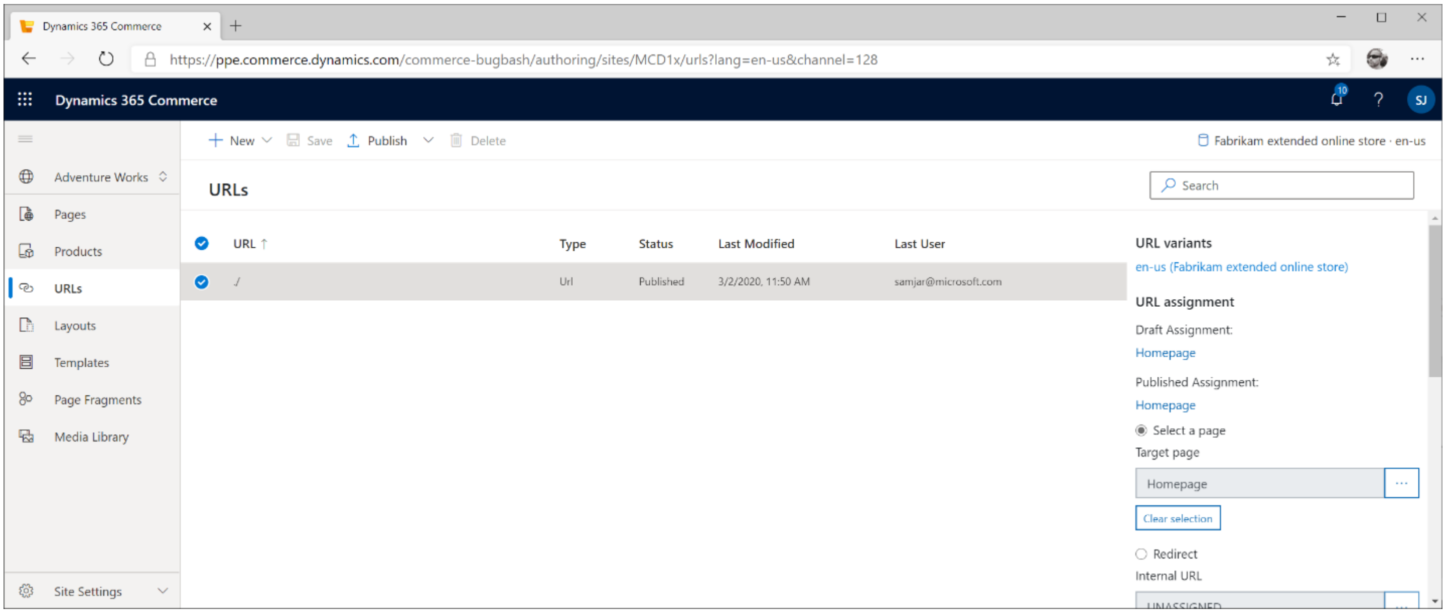
SETTING URL FOR HOME PAGE

We now want to set the page as the default page for our site. Select the “**URL**” tab then the “**+New**” button followed by “**New URL**” to create a new URL.

Select the Homepage we created and leave the URL empty and press “OK”.



Select the new URL and press “**Publish**” and you should now see it published in the URL list.



URLs

URL ↑	Type	Status	Last Modified	Last User
/	Url	Published	3/2/2020, 11:50 AM	samjar@microsoft.com

URL variants

en-us (Fabrikam extended online store)

URL assignment

Draft Assignment:
Homepage

Published Assignment:
Homepage

● Select a page

Target page: Homepage

○ Redirect

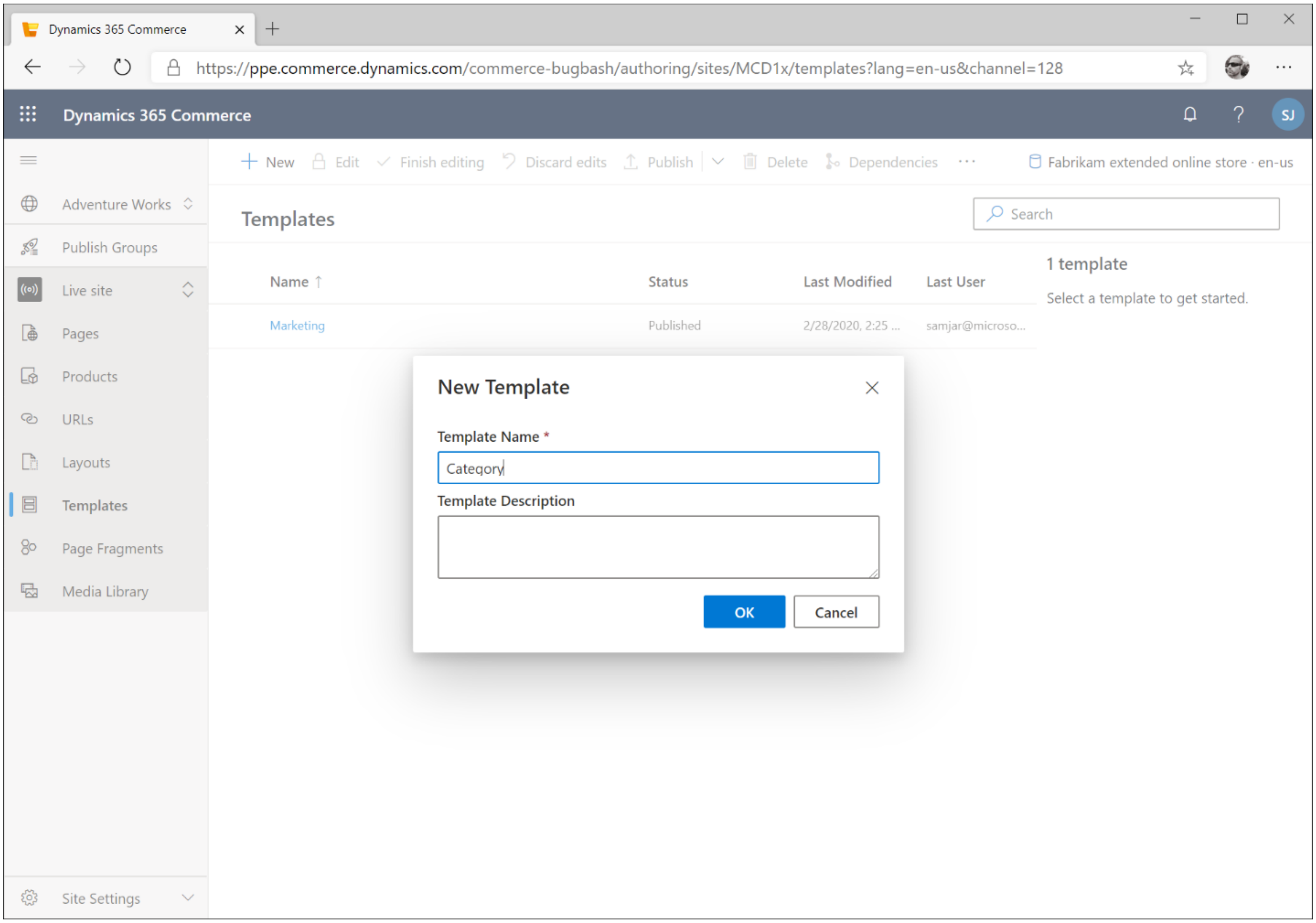
Internal URL: UNASSIGNED

CREATE A CATEGORY LAYOUT AND PAGE

This section will cover creating a category layout and page that will be used when a category is selected from the product hierarchy. It will also be used to display search results.

CREATE A CATEGORY LAYOUT

First, we will create a Category Layout Page, select the “Templates” tab and “New Template”. Provide a Template name such as “Category”.



Name ↑	Status	Last Modified	Last User
Marketing	Published	2/28/2020, 2:25 ...	samjar@microso...

New Template

Template Name *: Category

Template Description:

Add the following modules to the template and leave each configuration at its default values.

Under **"HTML Head**" slot add "**List page summary**" and configure favicon configuration as done previously when creating Homepage template with the same URL.

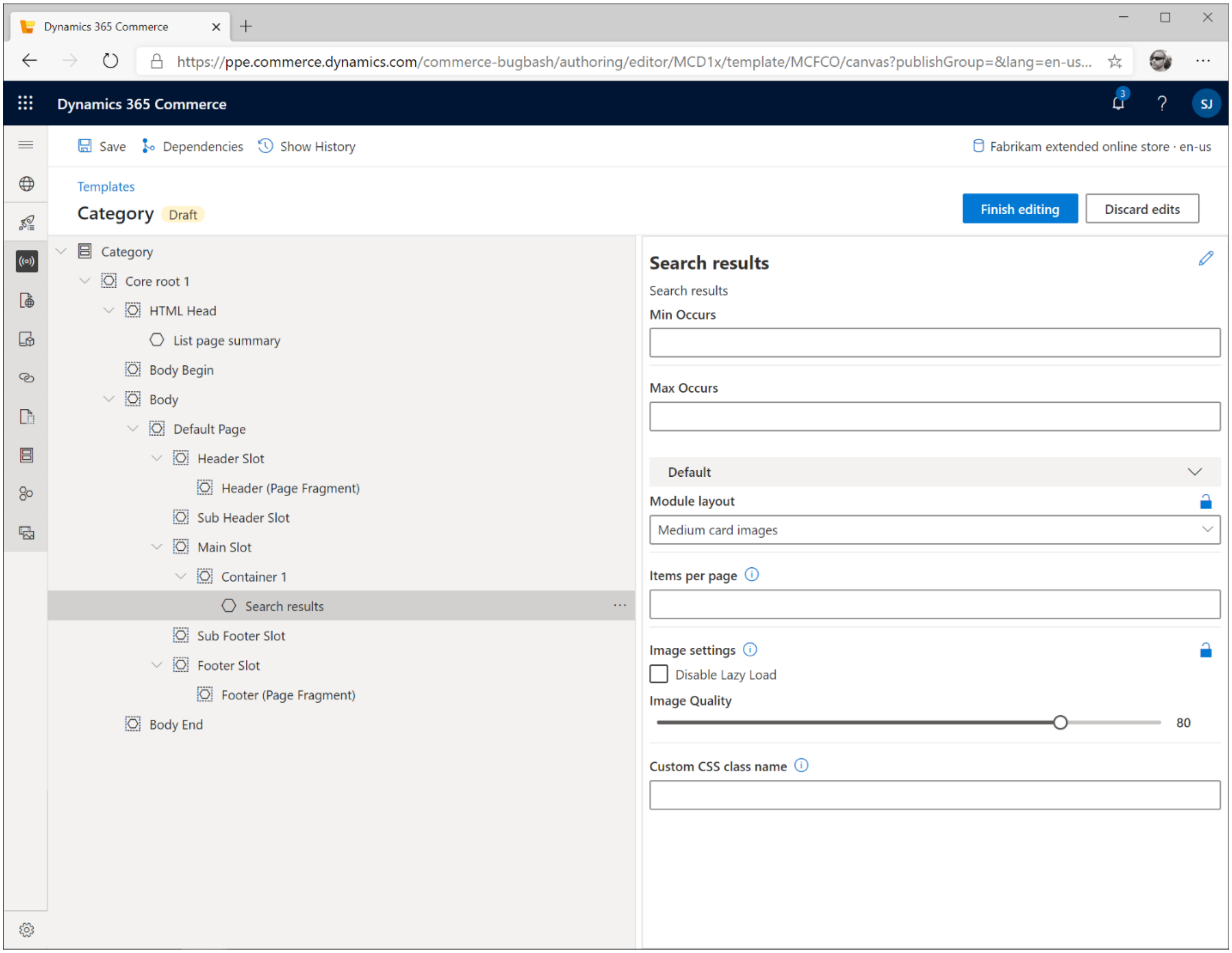
Under the "**Body**" slot, add a "**Default page**" module and you should now see several new slots. Configure them as follows.

"**Header Slot**" add the "**Header**" fragment created earlier

"**Footer Slot**" add the "**Footer**" fragment created earlier

Under "**Main Slot**" add a "**Container**" add the "**Search results**" module

The tree should look similar to the below image. Click "Finished editing" and Publish



Dynamics 365 Commerce

https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/template/MCFCO/canvas?publishGroup=&lang=en-us...

Dynamics 365 Commerce

Save Dependencies Show History

Fabrikam extended online store · en-us

Templates

Category Draft

Finish editing Discard edits

- Category
  - Core root 1
    - HTML Head
      - List page summary
    - Body Begin
    - Body
      - Default Page
        - Header Slot
          - Header (Page Fragment)
        - Sub Header Slot
        - Main Slot
          - Container 1
            - Search results
        - Sub Footer Slot
        - Footer Slot
          - Footer (Page Fragment)
    - Body End

Search results

Search results

Min Occurs

Max Occurs

Default

Module layout

Medium card images

Items per page

Image settings

□ Disable Lazy Load

Image Quality

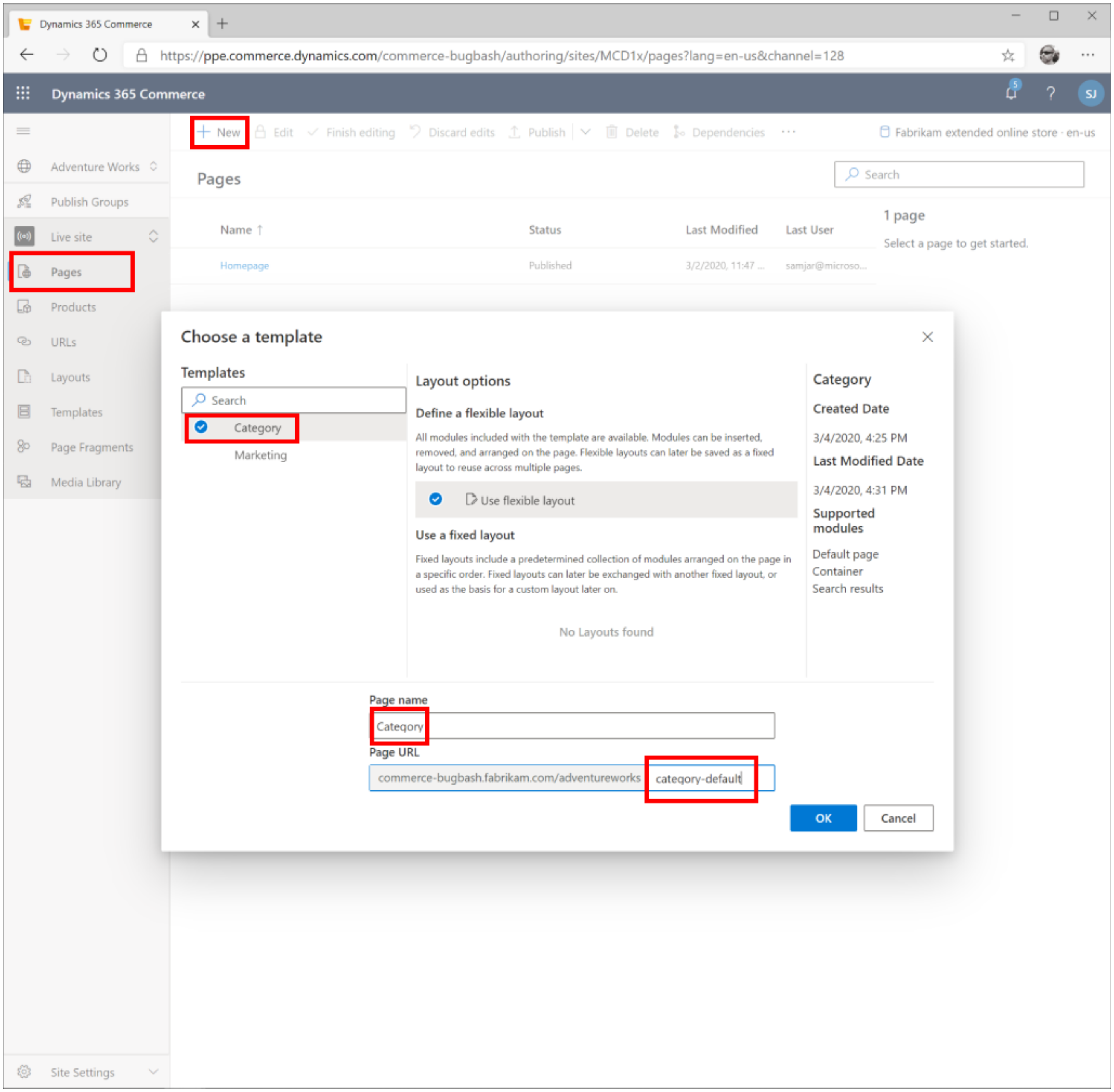
80

Custom CSS class name

CREATE A CATEGORY PAGE

We will now create a page based on the template we just created. Navigate back to the site and select the “Pages” tab on the left column.

Navigate back to the “**Pages**” tab and press the “**New**” button. Select the “**Category**” template and provide “**Category**” for the name and “**category-default**” for the URL. Press the “**OK**” button to create the page



Choose a template

Templates

Search

Category

Marketing

Layout options

Define a flexible layout

All modules included with the template are available. Modules can be inserted, removed, and arranged on the page. Flexible layouts can later be saved as a fixed layout to reuse across multiple pages.

● Use flexible layout

Use a fixed layout

Fixed layouts include a predetermined collection of modules arranged on the page in a specific order. Fixed layouts can later be exchanged with another fixed layout, or used as the basis for a custom layout later on.

No Layouts found

Category

Created Date

3/4/2020, 4:25 PM

Last Modified Date

3/4/2020, 4:31 PM

Supported modules

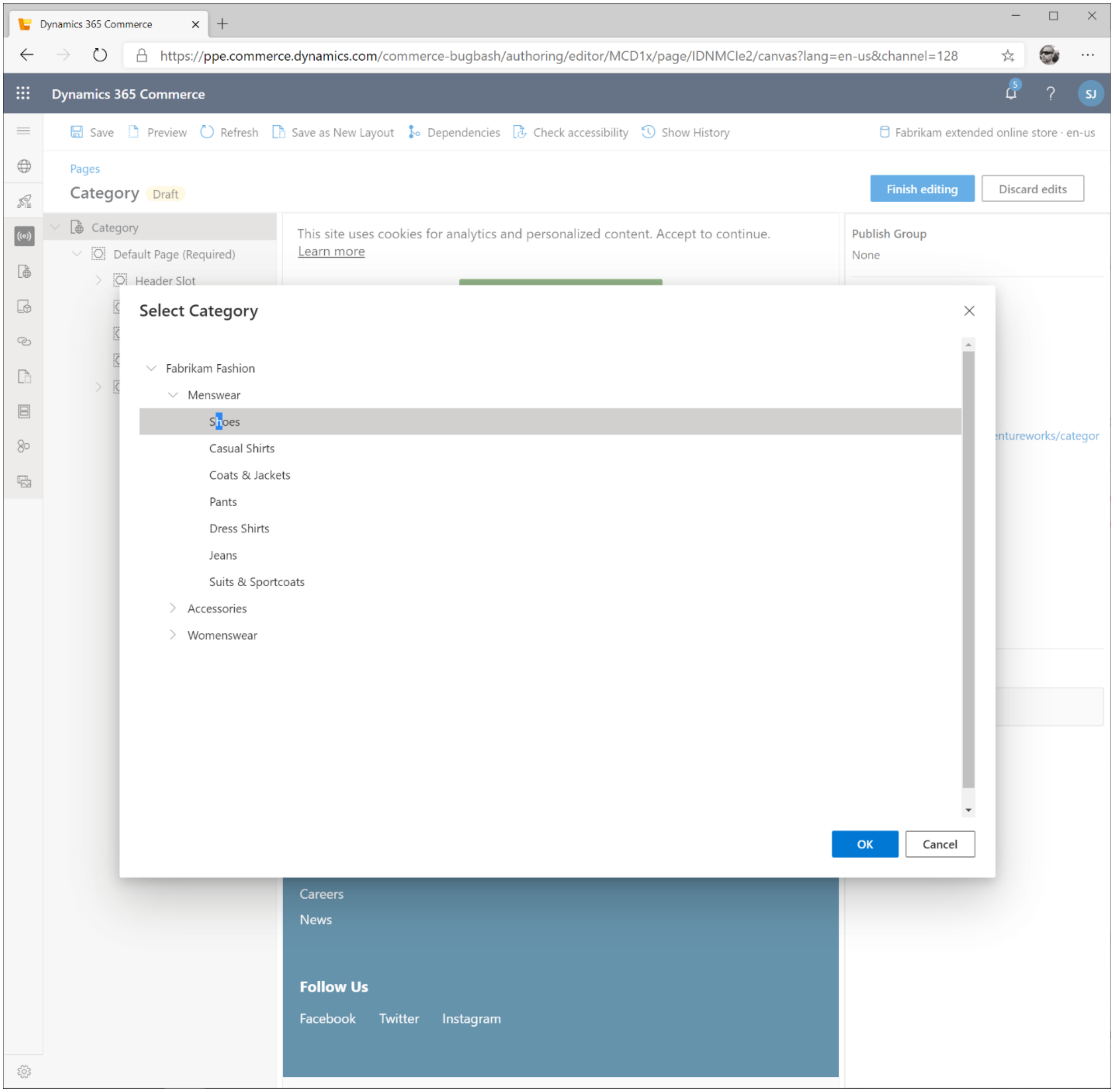
Default page
Container
Search results

Page name: Category

Page URL: commerce-bugbash.fabrikam.com/adventureworks category-default

OK Cancel

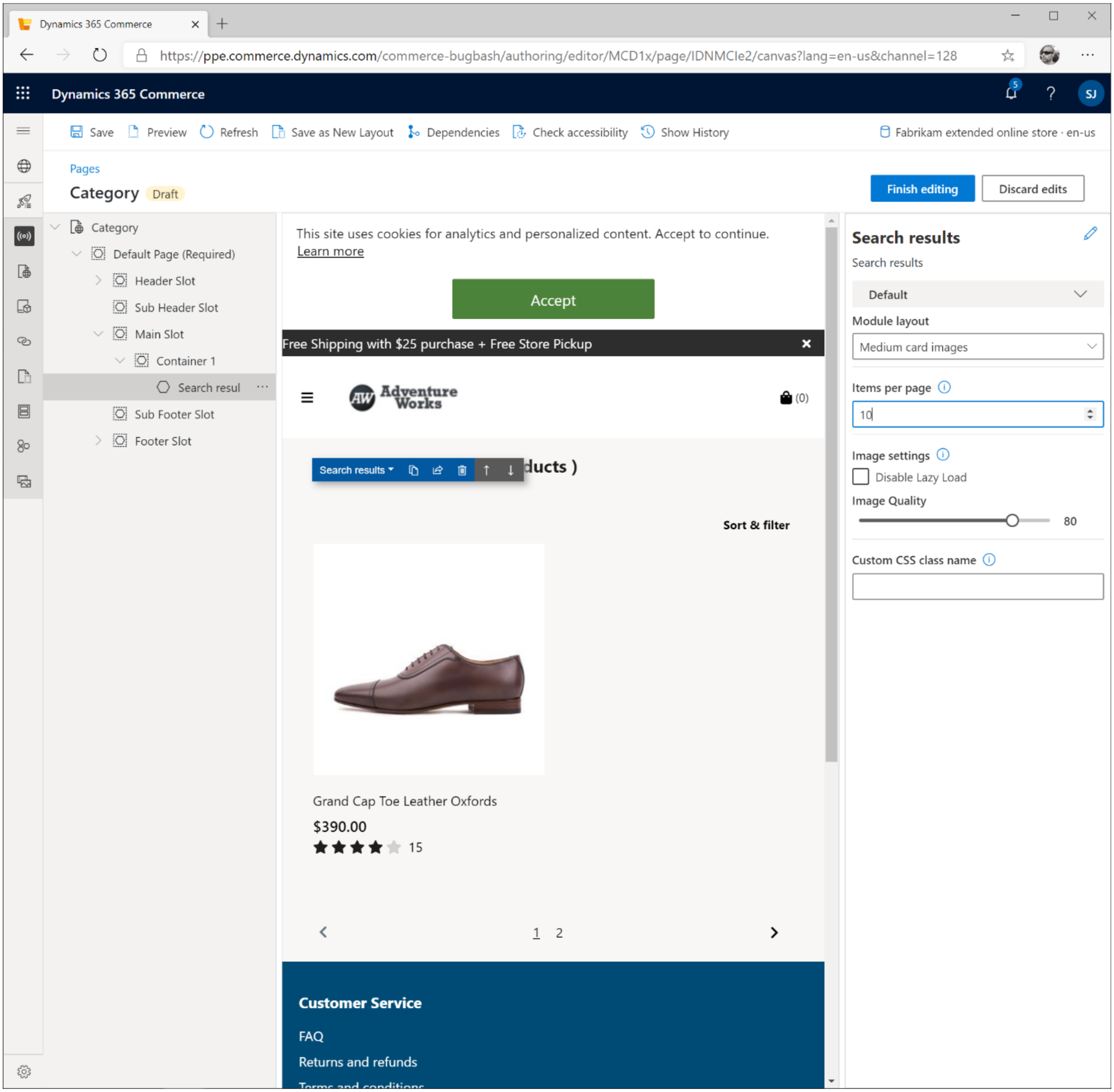
Since this is a category page, when it opens it will ask you to pick one of your category pages to use so that it has data to render. Select any category and press “**OK**”.



We will now add modules to the page. Even though we added modules to the layout, templates only dictate them as options, we still need to add them to the page.

Add the following modules to their respective slots:

- "**Main Slot**"
    - Add "**Container**"
        - Set "Width" to "Fill Container"
- "**Container**"
    - Add "**Search result**"
        - Set "items per page" = 10



Search results

Search results

Default

Module layout: Medium card images

Items per page: 10

Image settings

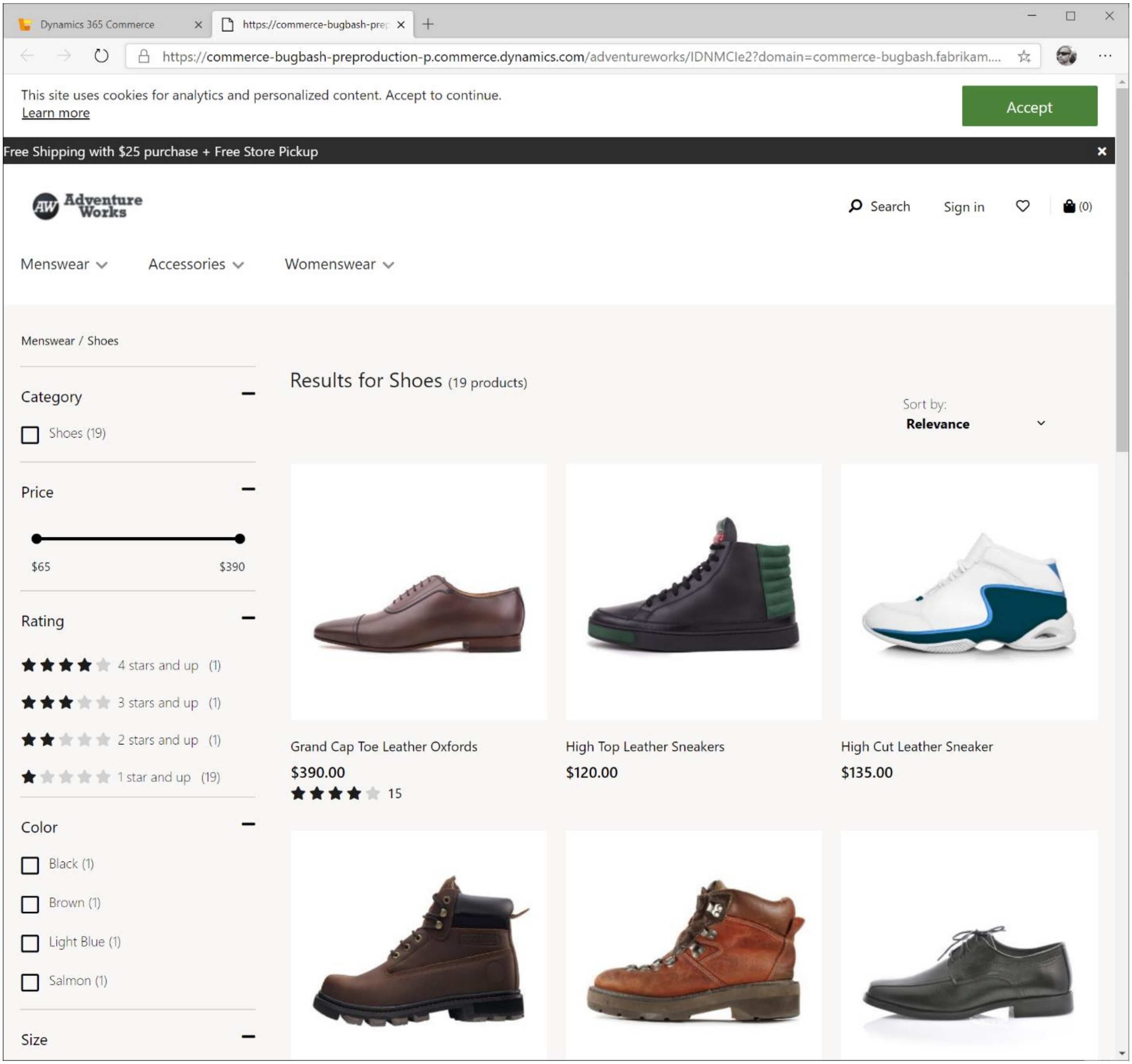
☐ Disable Lazy Load

Image Quality: 80

Custom CSS class name:

Now you can “Finish editing”, “Preview” and “Publish” the page.

When you preview the page, you should see a page that looks like the below page. Notice as you select different categories, they will load the appropriate set of products.



https://commerce-bugbash-preproduction-p.commerce.dynamics.com/adventureworks/IDNMCIe2?domain=commerce-bugbash.fabrikam....

This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Free Shipping with $25 purchase + Free Store Pickup

Menswear / Shoes

Results for Shoes (19 products)

Sort by: **Relevance**

Category

□ Shoes (19)

Price

$65 $390

Rating

★★★★☆ 4 stars and up (1)

★★★☆☆ 3 stars and up (1)

★★☆☆☆ 2 stars and up (1)

★☆☆☆☆ 1 star and up (19)

Color

□ Black (1)

□ Brown (1)

□ Light Blue (1)

□ Salmon (1)

Size

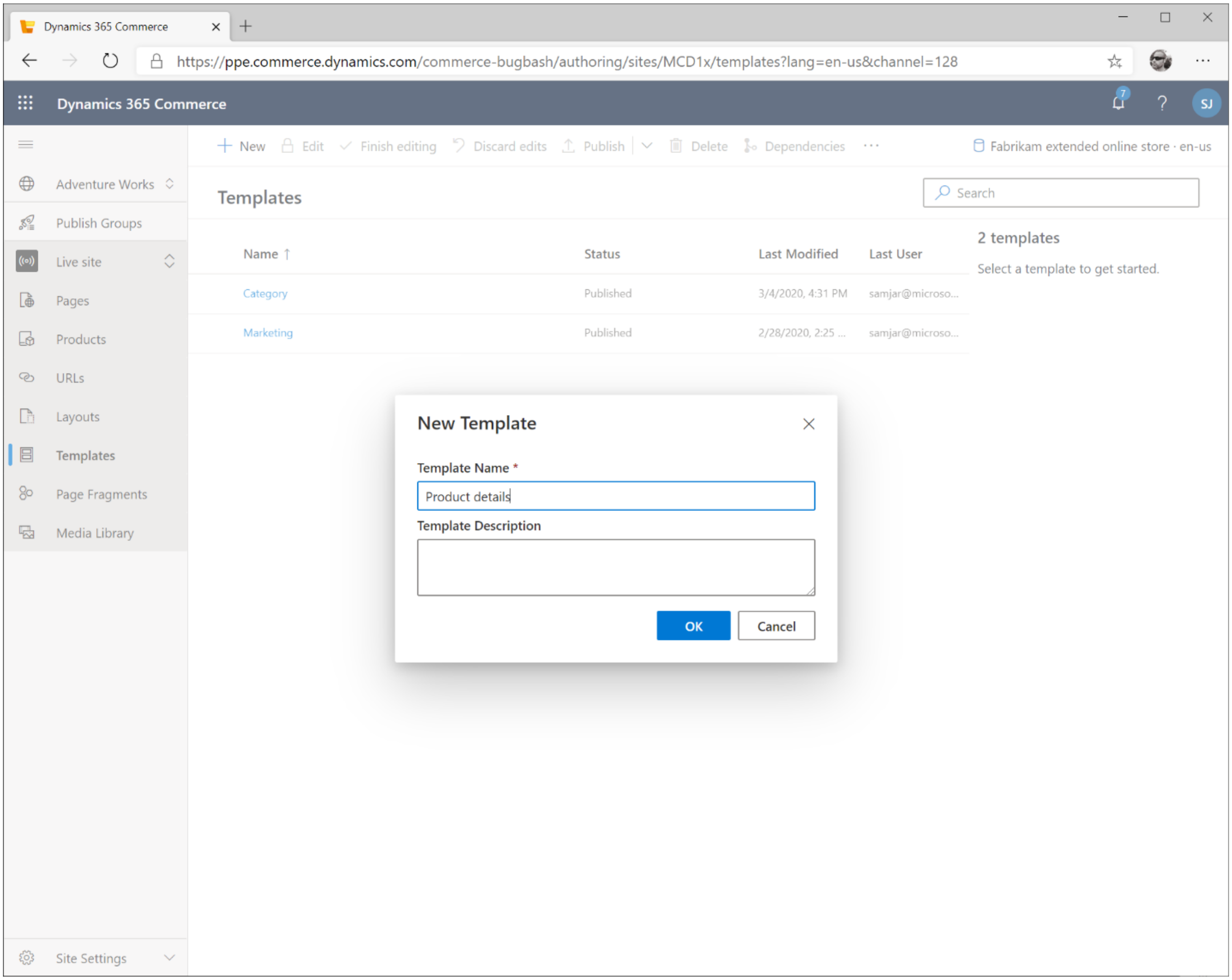
Grand Cap Toe Leather Oxfords **$390.00** ★★★★☆ 15	High Top Leather Sneakers **$120.00**	High Cut Leather Sneaker **$135.00**

CREATE A PRODUCT DETAIL TEMPLATE AND PAGE

In this section we will create the needed template followed by a page for the product details page.

CREATE A PRODUCT DETAIL PAGE TEMPLATE

Navigate to the “**Templates**” tab and select “**New**” and provide “**Product details**” for the template name.



Name ↑	Status	Last Modified	Last User
Category	Published	3/4/2020, 4:31 PM	samjar@microso...
Marketing	Published	2/28/2020, 2:25 ...	samjar@microso...

New Template

Template Name *

Product details

Template Description

Under the “**HTML Head**” slot add a “**Product page summary**” module. Leave the configuration properties at their defaults.

Under the “**Body**” add a “**Default Page**” module.

Under the “**Default Page**”, **“Header Slot**” add the “**Header fragment**”.

Under the “**Default Page**”, **“Footer Slot**” add the “**Footer fragment**”.

We will now focus on the main content structure for the PDP page.

Inside the “**Main slot**” add a “**Container**” module and se the “**Width**” to “**Fill Container**”.

Add a “**Buybox**” module to the “**Container**”.

In the “**Media Gallery**” slot add a “**Media Gallery**” module.

In the “**Store Selector**” slot add a “**Store selector**” module.

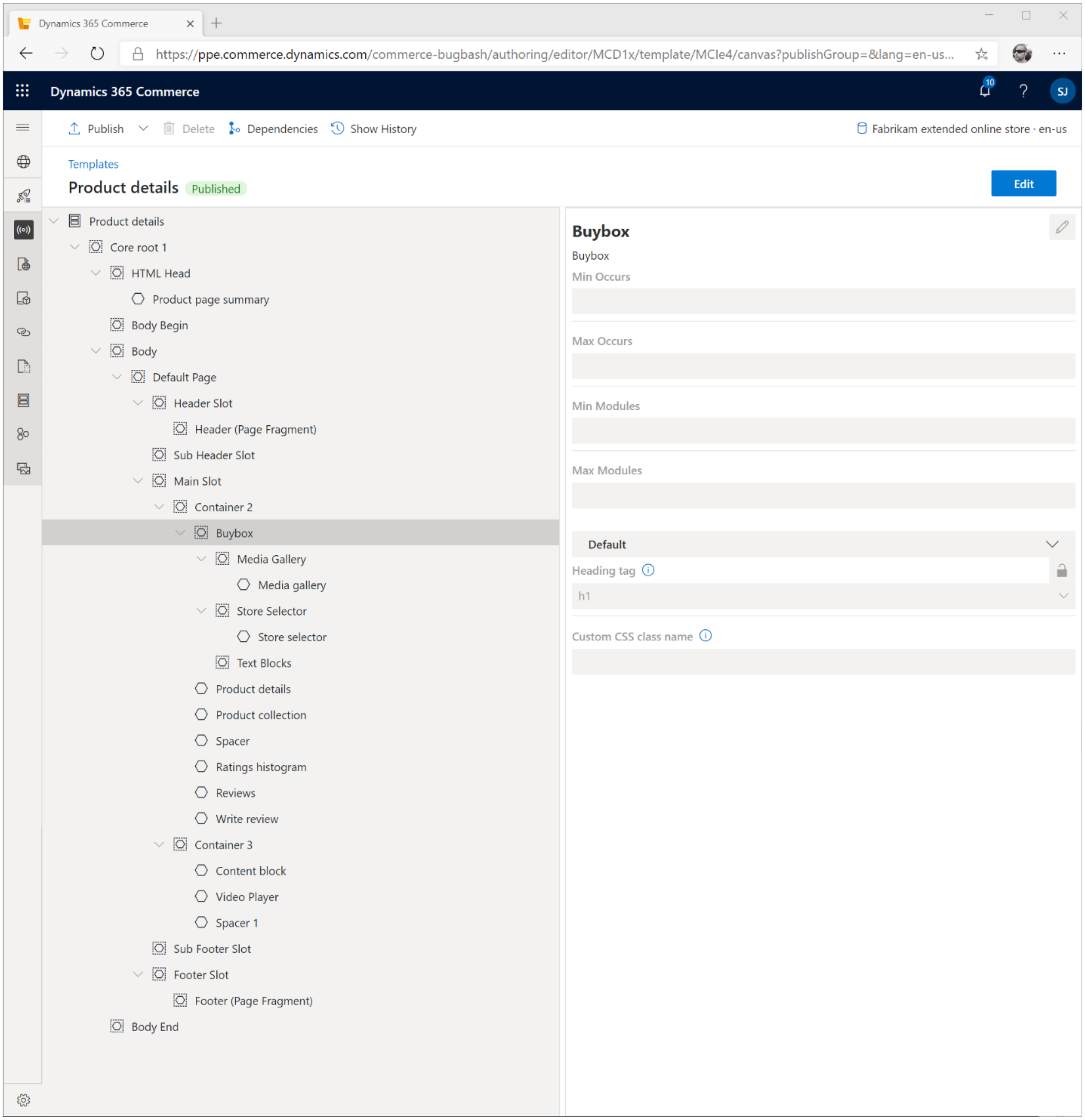
Inside of the same “Container” we added the Buybox above also add the following modules:

- Product Details
- Product Collection
- Spacer (Set Spacer height to 2x)
- Ratings histogram
- Reviews
- Write review

Under the second “Container” under the “Main Slot” add the following modules

- Content block
- Video Player
- Spacer 1

Select “Finish editing” and “Publish”


Dynamics 365 Commerce
https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/template/MCle4/canvas?publishGroup=&lang=en-us...
Dynamics 365 Commerce
10
?
SJ
Publish
Delete
Dependencies
Show History
Fabrikam extended online store · en-us
Templates
Edit

Product details Published

- Product details
  - Core root 1
    - HTML Head
      - Product page summary
    - Body Begin
    - Body
      - Default Page
        - Header Slot
          - Header (Page Fragment)
        - Sub Header Slot
        - Main Slot
          - Container 2
            - Buybox
              - Media Gallery
                - Media gallery
              - Store Selector
                - Store selector
              - Text Blocks
            - Product details
            - Product collection
            - Spacer
            - Ratings histogram
            - Reviews
            - Write review
          - Container 3
            - Content block
            - Video Player
            - Spacer 1
        - Sub Footer Slot
        - Footer Slot
          - Footer (Page Fragment)
    - Body End

Buybox

Buybox

Min Occurs

Max Occurs

Min Modules

Max Modules

Default

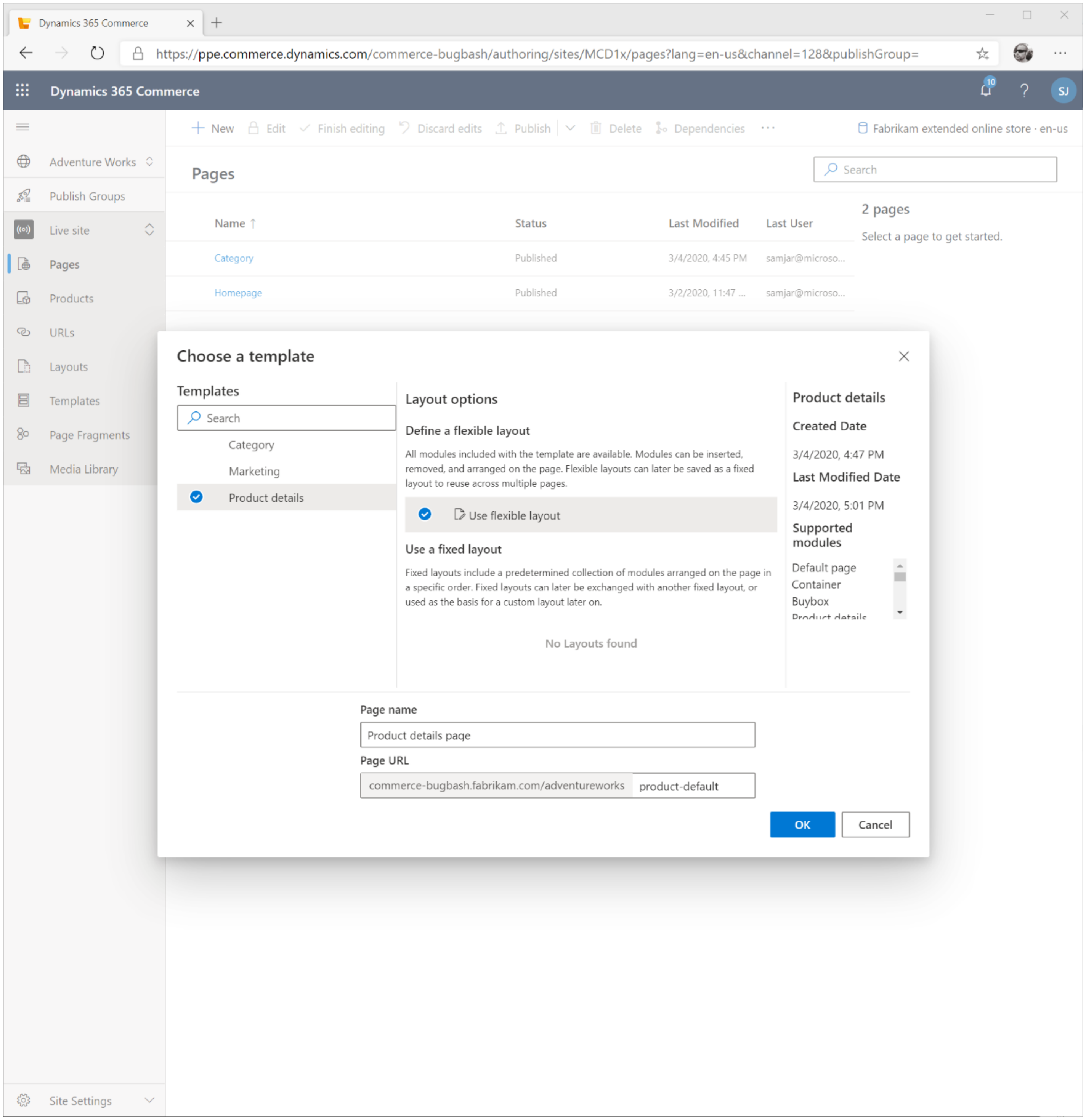
Heading tag

h1

Custom CSS class name

CREATE A PRODUCT DETAIL PAGE

Now that we have a PDP template, we can create a page. Navigate back to the "**Pages**" tab and click "**New Page**". Give it the name "**Product details page**" and select the "**Product details**" template we just created and add "product-default" for the URL.



Name ↑	Status	Last Modified	Last User
Category	Published	3/4/2020, 4:45 PM	samjar@microso...
Homepage	Published	3/2/2020, 11:47 ...	samjar@microso...

Choose a template

Templates

- Category
- Marketing
- ● Product details

Layout options

Define a flexible layout

All modules included with the template are available. Modules can be inserted, removed, and arranged on the page. Flexible layouts can later be saved as a fixed layout to reuse across multiple pages.

● Use flexible layout

Use a fixed layout

Fixed layouts include a predetermined collection of modules arranged on the page in a specific order. Fixed layouts can later be exchanged with another fixed layout, or used as the basis for a custom layout later on.

No Layouts found

Product details

Created Date

3/4/2020, 4:47 PM

Last Modified Date

3/4/2020, 5:01 PM

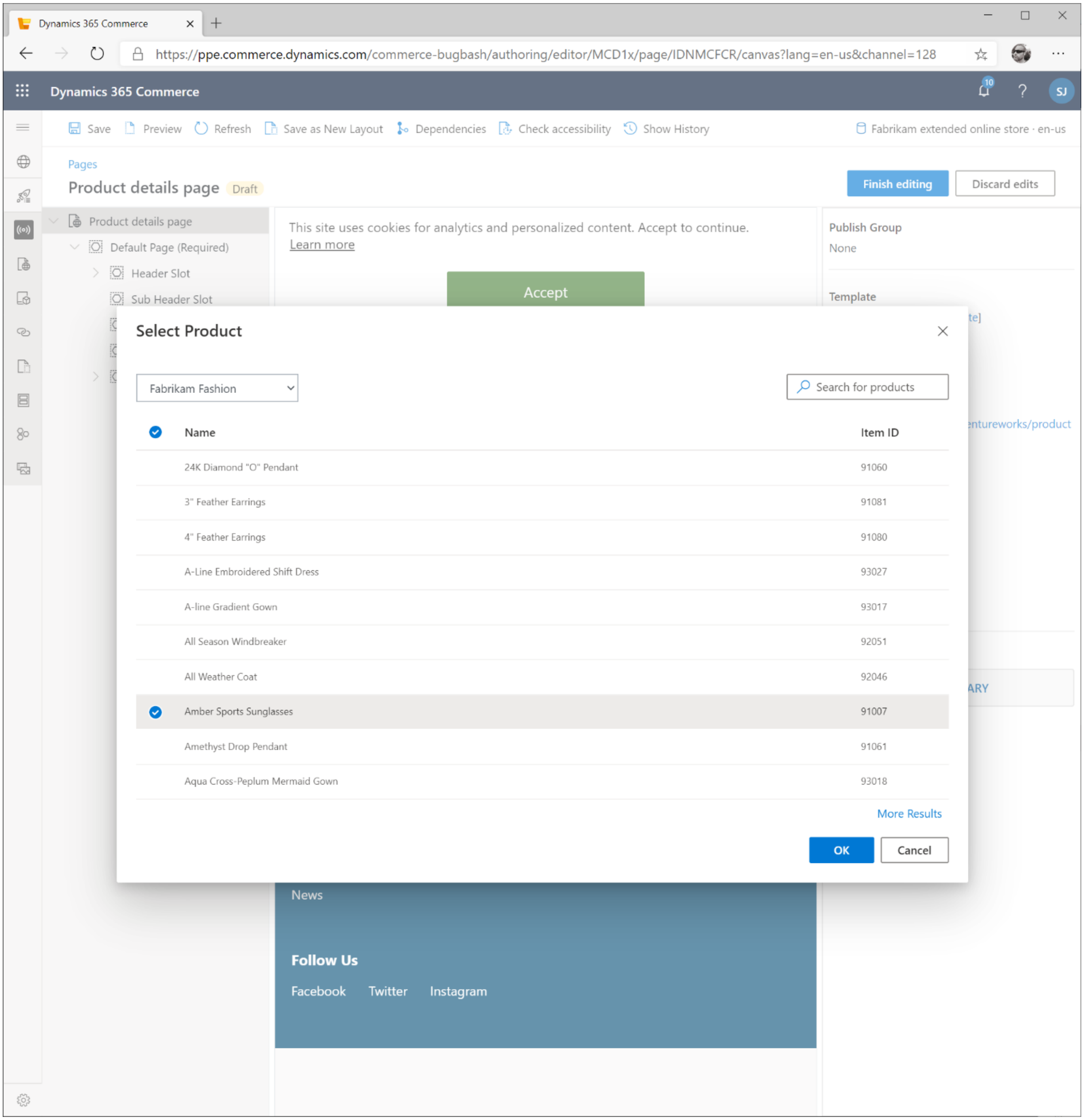
Supported modules

Default page
Container
Buybox
Product details

Page name: Product details page

Page URL: commerce-bugbash.fabrikam.com/adventureworks product-default

As with the category page, we needed to select a category to render content in the page editor, we also get prompt to select a product, this is just used for rending a test view of our page, select any product.



This site uses cookies for analytics and personalized content. Accept to continue. Learn more

Select Product

Fabrikam Fashion

Search for products

	Name	Item ID
	24K Diamond "O" Pendant	91060
	3" Feather Earrings	91081
	4" Feather Earrings	91080
	A-Line Embroidered Shift Dress	93027
	A-line Gradient Gown	93017
	All Season Windbreaker	92051
	All Weather Coat	92046
✓	Amber Sports Sunglasses	91007
	Amethyst Drop Pendant	91061
	Aqua Cross-Peplum Mermaid Gown	93018

More Results

OK Cancel

We can now add the following modules to our page

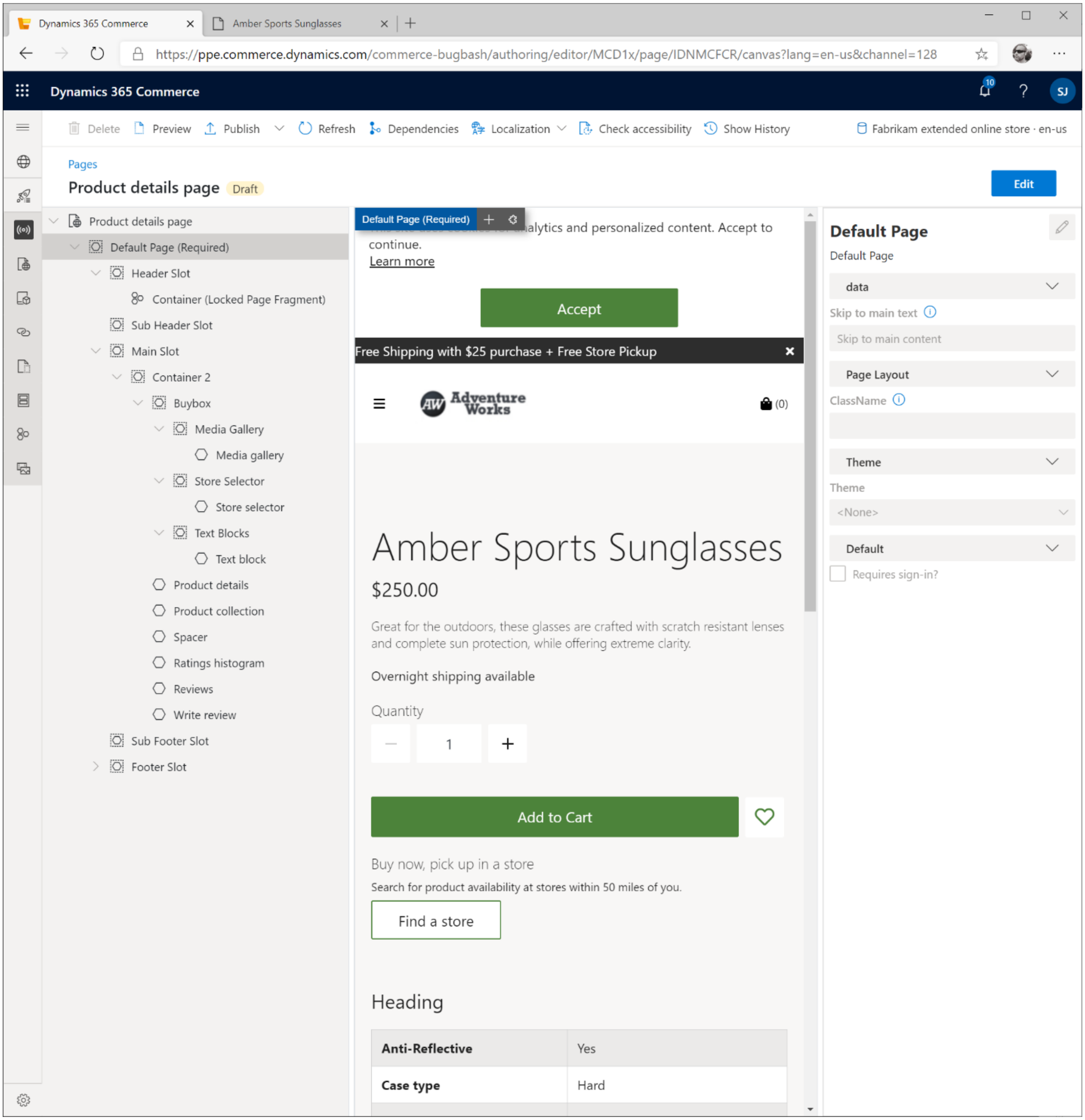
Under “Main Slot” add a “Container” module,

Then under this add a

- “Buybox” module and under it add
  - Media Gallery Slot
    - Media Gallery module
  - Store Selector Slot
    - Store selector slot
  - Text Block
    - Text block” module and set text to “Overnight shipping available”
- Product details
- Product collection
- Spacer
- Ratings historgram
- Reviews
- Write Review

As you add the modules, you should see updates happening live in the middle panel against the product you chose when you opened this page.

Click “Finish editing”, Preview, and Publish when complete. Try navigating to different categories and selecting different products and the PDP page should reflect the product you specified.



...alytics and personalized content. Accept to continue.

Learn more

Free Shipping with $25 purchase + Free Store Pickup

Amber Sports Sunglasses

$250.00

Great for the outdoors, these glasses are crafted with scratch resistant lenses and complete sun protection, while offering extreme clarity.

Overnight shipping available

Quantity

Buy now, pick up in a store

Search for product availability at stores within 50 miles of you.

Heading

Anti-Reflective	Yes
Case type	Hard

Skip to main text: Skip to main content

ClassName:

Theme: <None>

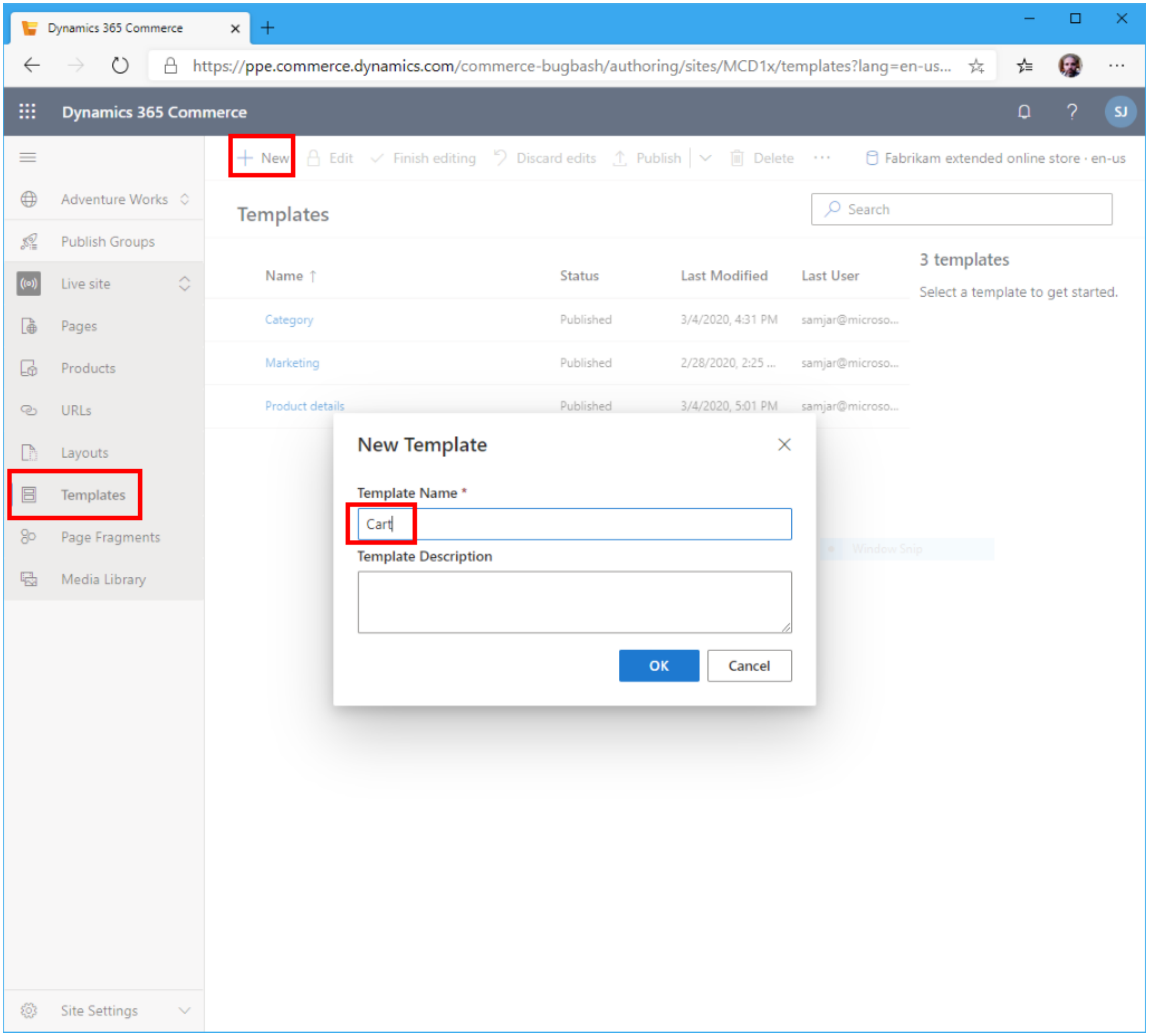
□ Requires sign-in?

CREATE A CART TEMPLATE AND PAGE

As we did with previous pages, we will start by creating a template and then the page.

CREATE A CART TEMPLATE

Navigate to the “Templates” tab and select “New Template” and provide a name such as “Cart”.



Templates

Name ↑	Status	Last Modified	Last User
Category	Published	3/4/2020, 4:31 PM	samjar@microso...
Marketing	Published	2/28/2020, 2:25 ...	samjar@microso...
Product details	Published	3/4/2020, 5:01 PM	samjar@microso...

New Template

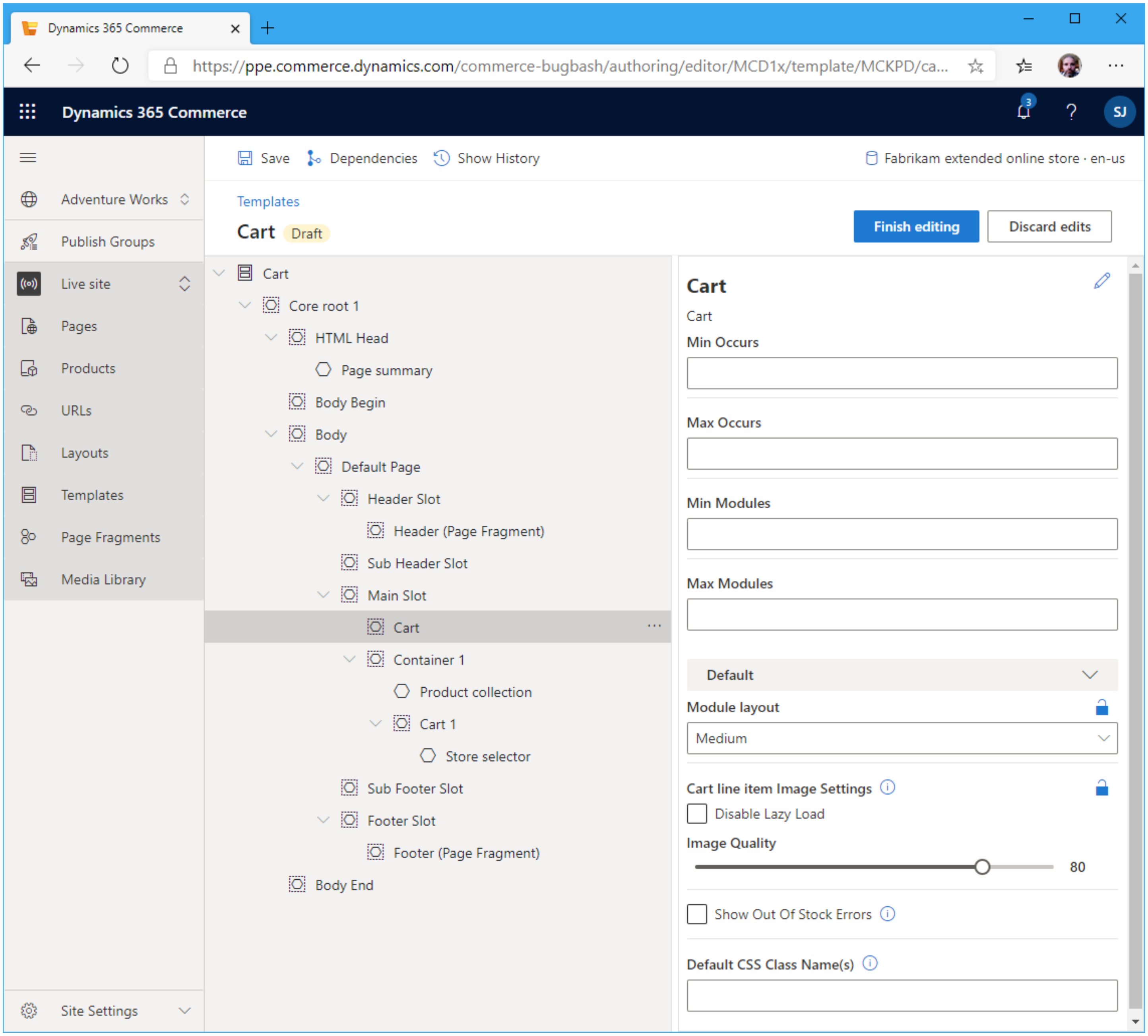
Template Name *

Cart

Template Description

Next add the following to the specified slots:

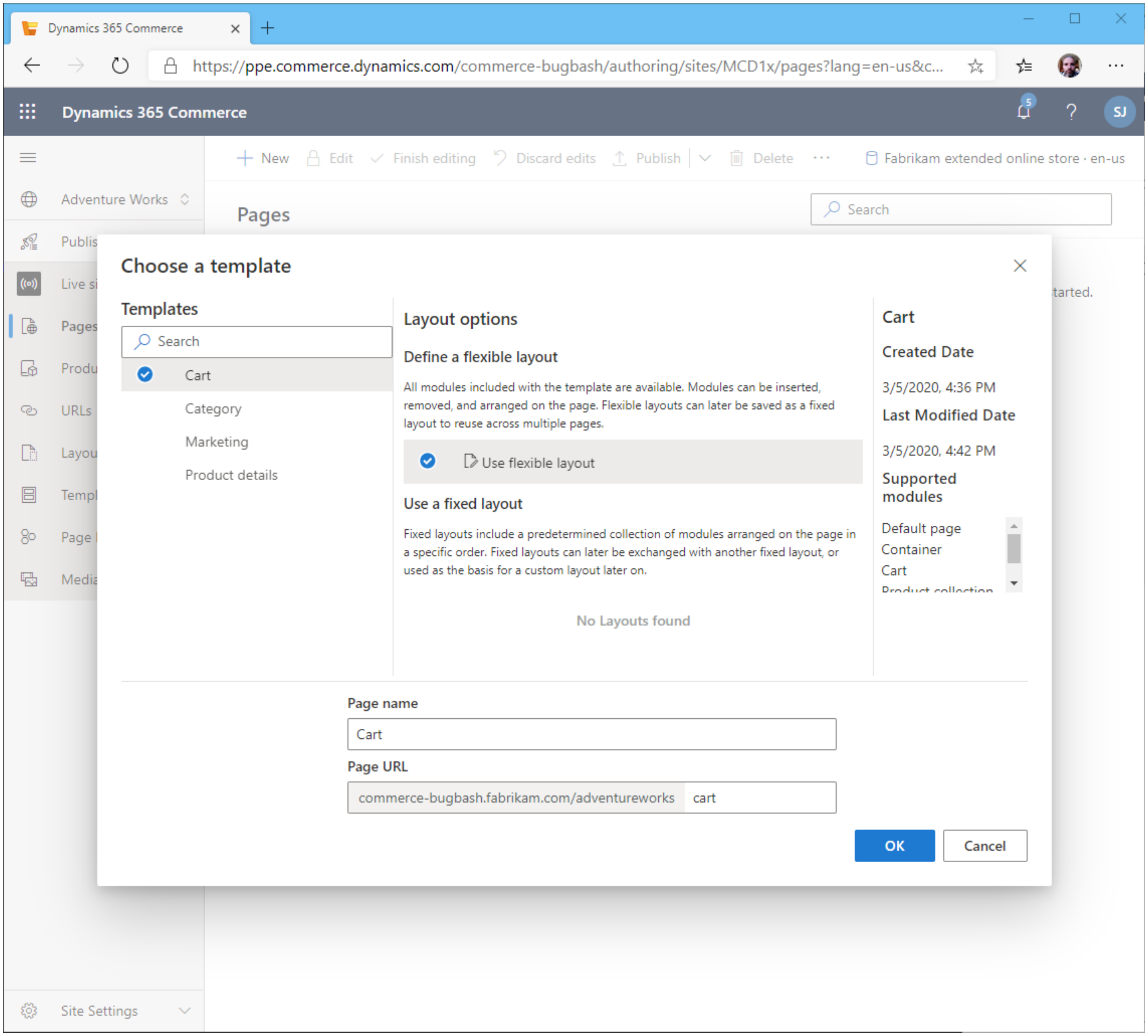
- HTML Head
    - Add “Page summary” module
- Body Slot -> Add “Default page” then:
    - “Header Slot” – add Header fragment
    - “Footer Slot” – add Footer fragment
    - “Main Slot” Add the following
        - “Cart” module
        - “Container” module and add:
            - “Product collection”
            - Cart module and add a child
                - “Store selector”



"Finish editing" and Publish the template

CREATE A CART PAGE

Navigate to the "Pages" tab and select "New". Provide the name "Cart" and select the "Cart" template and specify "cart" as the URL. Click "OK" button to create the page. Add all of the modules we added in the template.



Choose a template

Templates

Layout options

Define a flexible layout

All modules included with the template are available. Modules can be inserted, removed, and arranged on the page. Flexible layouts can later be saved as a fixed layout to reuse across multiple pages.

☑ Use flexible layout

Use a fixed layout

Fixed layouts include a predetermined collection of modules arranged on the page in a specific order. Fixed layouts can later be exchanged with another fixed layout, or used as the basis for a custom layout later on.

No Layouts found

Cart

Created Date: 3/5/2020, 4:36 PM

Last Modified Date: 3/5/2020, 4:42 PM

Supported modules:

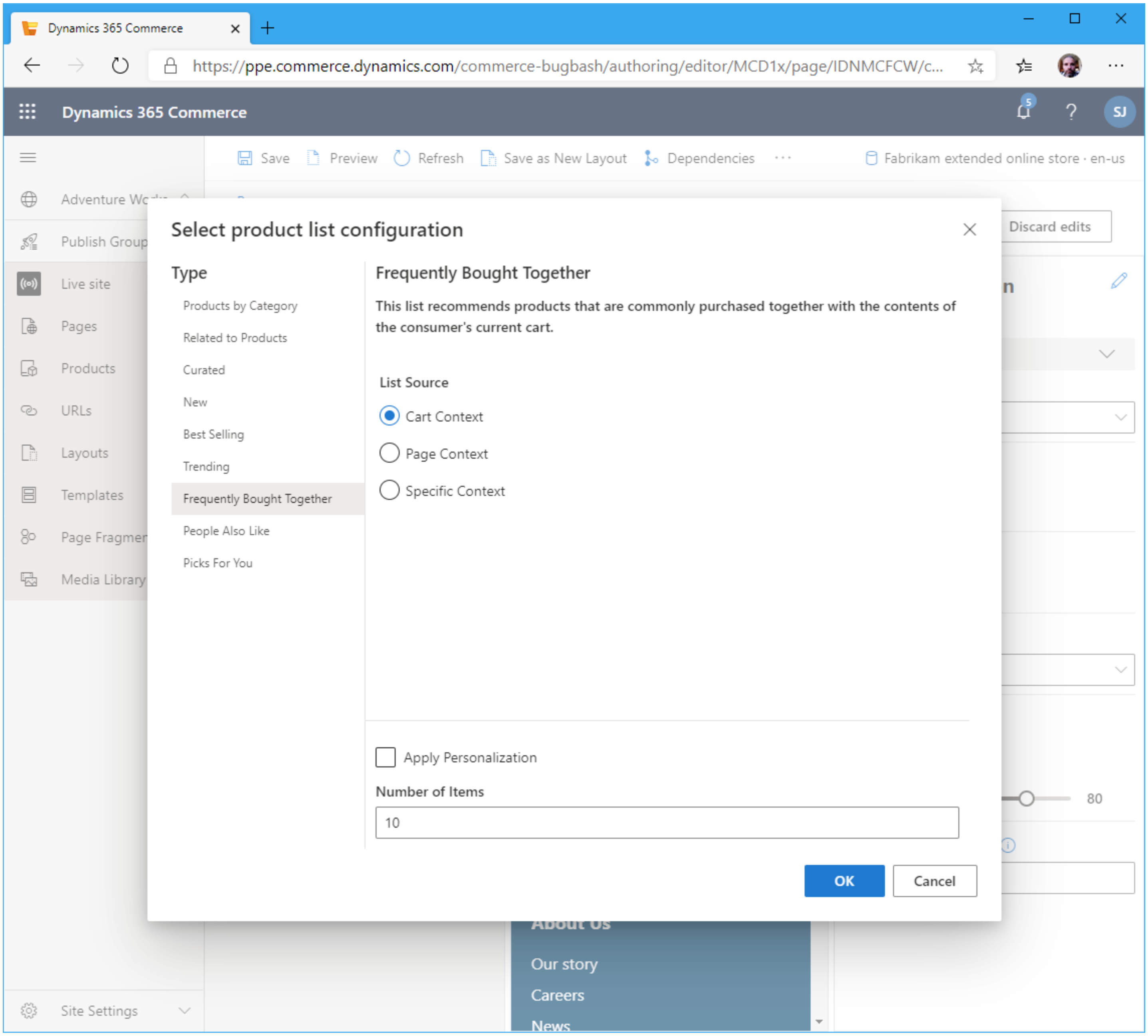
- Default page
- Container
- Cart
- Product collection

Page name: Cart

Page URL: commerce-bugbash.fabrikam.com/adventureworks cart

Under the Main slot, add:

- "Container" module and set Width to "Fill container" then
    - Add "Cart" module then
        - Add "Store selector" module
    - Add Product collection module
        - Configure "Product" "Add a product list" and pick "frequently bought together"



Select product list configuration

Type

- Products by Category
- Related to Products
- Curated
- New
- Best Selling
- Trending
- Frequently Bought Together
- People Also Like
- Picks For You

Frequently Bought Together

This list recommends products that are commonly purchased together with the contents of the consumer's current cart.

List Source

● Cart Context

○ Page Context

○ Specific Context

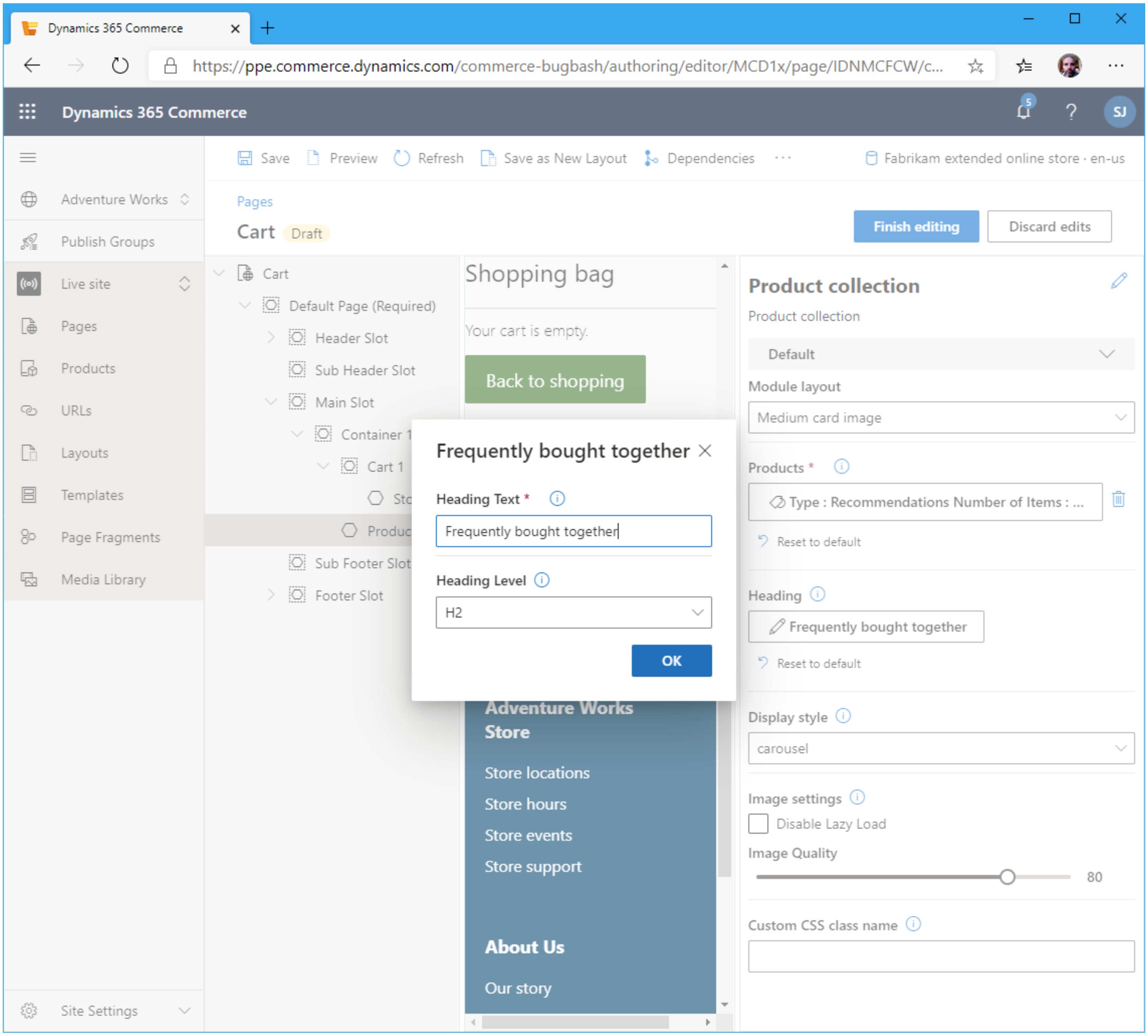
□ Apply Personalization

Number of Items: 10

OK Cancel

We'll now configure the frequently bought together recommendations.

Select the "Product collection" module and configure the "Heading" text to "Frequently bought together"



Frequently bought together

Heading Text *: Frequently bought together

Heading Level: H2

OK

Product collection

Product collection: Default

Module layout: Medium card image

Products *: Type : Recommendations Number of Items : ...

Reset to default

Heading: Frequently bought together

Reset to default

Display style: carousel

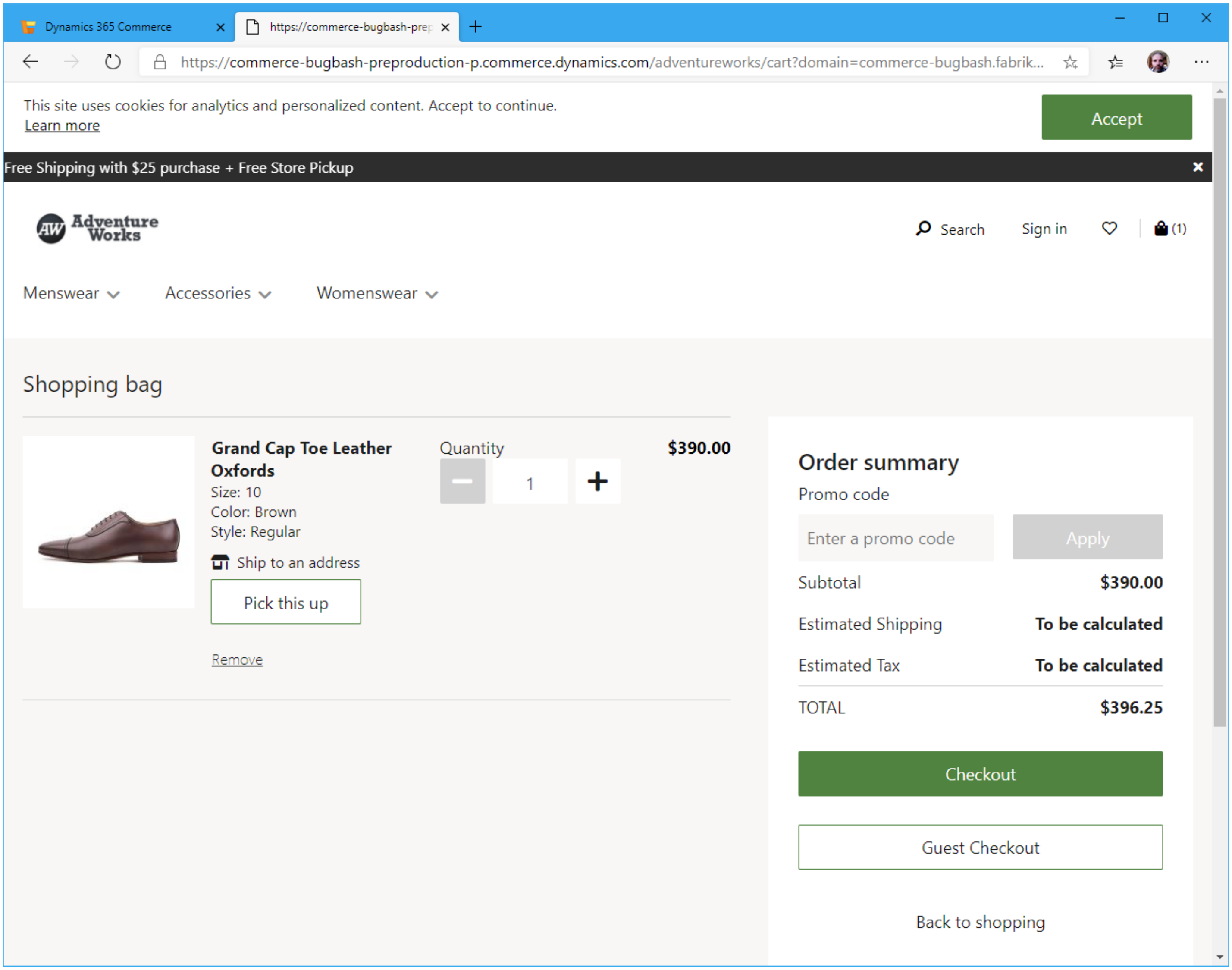
Image settings: □ Disable Lazy Load

Image Quality: 80

Custom CSS class name:

Click "Finish editing" and publish.

Now go back to your home page and preview it. Select a product and add it to the cart. The new cart page should now show up. You can also click the cart button at the top to get to the cart at any time.



This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Free Shipping with $25 purchase + Free Store Pickup

Shopping bag

Grand Cap Toe Leather Oxfords
Size: 10
Color: Brown
Style: Regular
Ship to an address

Quantity: 1

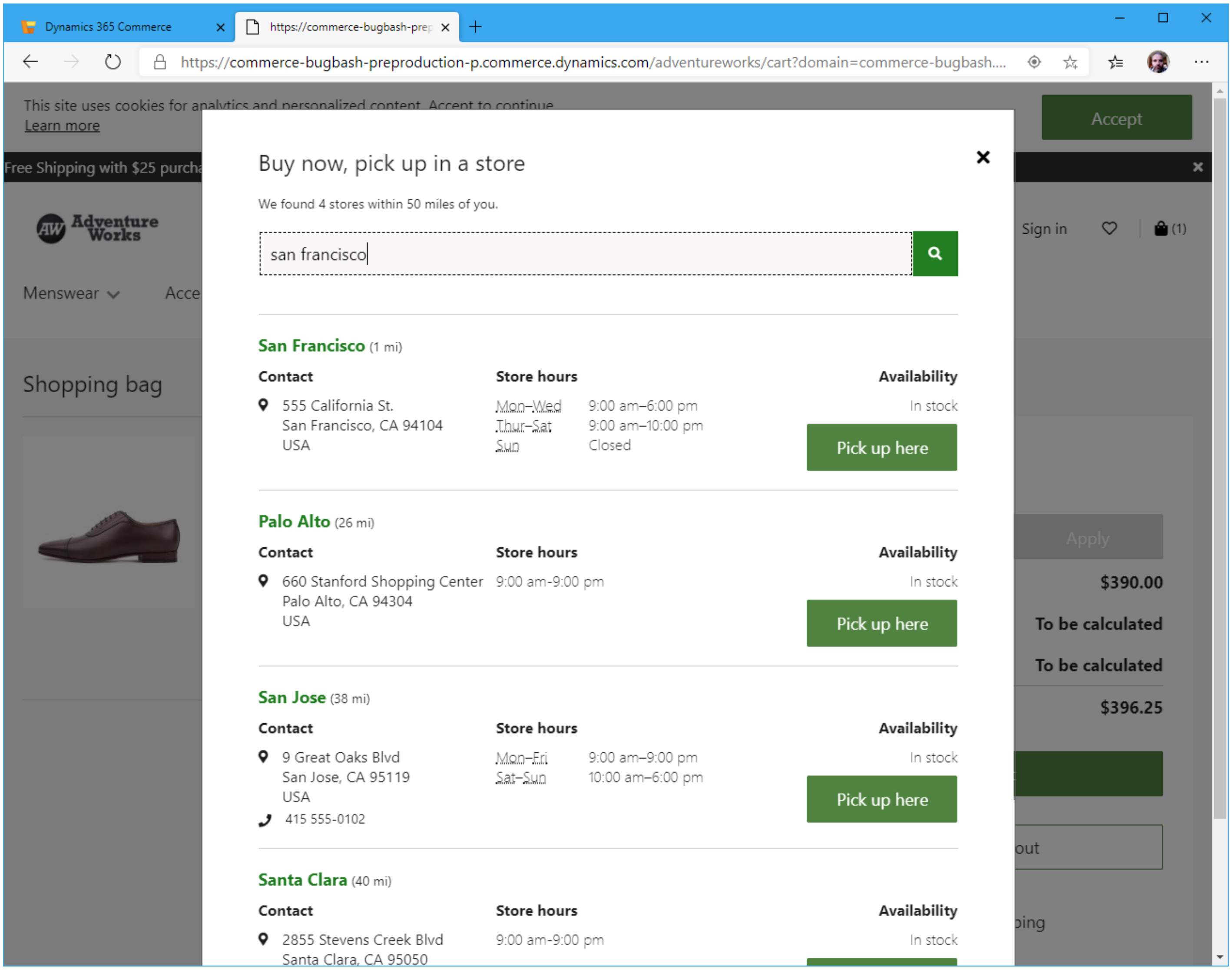
$390.00

Order summary

Promo code: Enter a promo code

Subtotal	**$390.00**
Estimated Shipping	**To be calculated**
Estimated Tax	**To be calculated**
TOTAL	**$396.25**

The “Pick this up” button should work to look up inventory at one of the stores. If you are using the demo data products that start with 9xxxx then most products will have inventory in the “San Francisco” retail store, you can search for it in the UI as shown below.


Dynamics 365 Commerce
https://commerce-bugbash-prep
https://commerce-bugbash-preproduction-p.commerce.dynamics.com/adventureworks/cart?domain=commerce-bugbash....
This site uses cookies for analytics and personalized content. Accept to continue.
Learn more
Accept
Free Shipping with $25 purcha
Adventure Works
Sign in
(1)
Menswear
Acce
Shopping bag
Apply
$390.00
To be calculated
To be calculated
$396.25
out
ping

Buy now, pick up in a store

We found 4 stores within 50 miles of you.

san francisco

San Francisco (1 mi)

Contact	Store hours		Availability
555 California St. San Francisco, CA 94104 USA	Mon–Wed Thur–Sat Sun	9:00 am–6:00 pm 9:00 am–10:00 pm Closed	In stock Pick up here

Palo Alto (26 mi)

Contact	Store hours	Availability
660 Stanford Shopping Center Palo Alto, CA 94304 USA	9:00 am-9:00 pm	In stock Pick up here

San Jose (38 mi)

Contact	Store hours		Availability
9 Great Oaks Blvd San Jose, CA 95119 USA 415 555-0102	Mon–Fri Sat–Sun	9:00 am–9:00 pm 10:00 am–6:00 pm	In stock Pick up here

Santa Clara (40 mi)

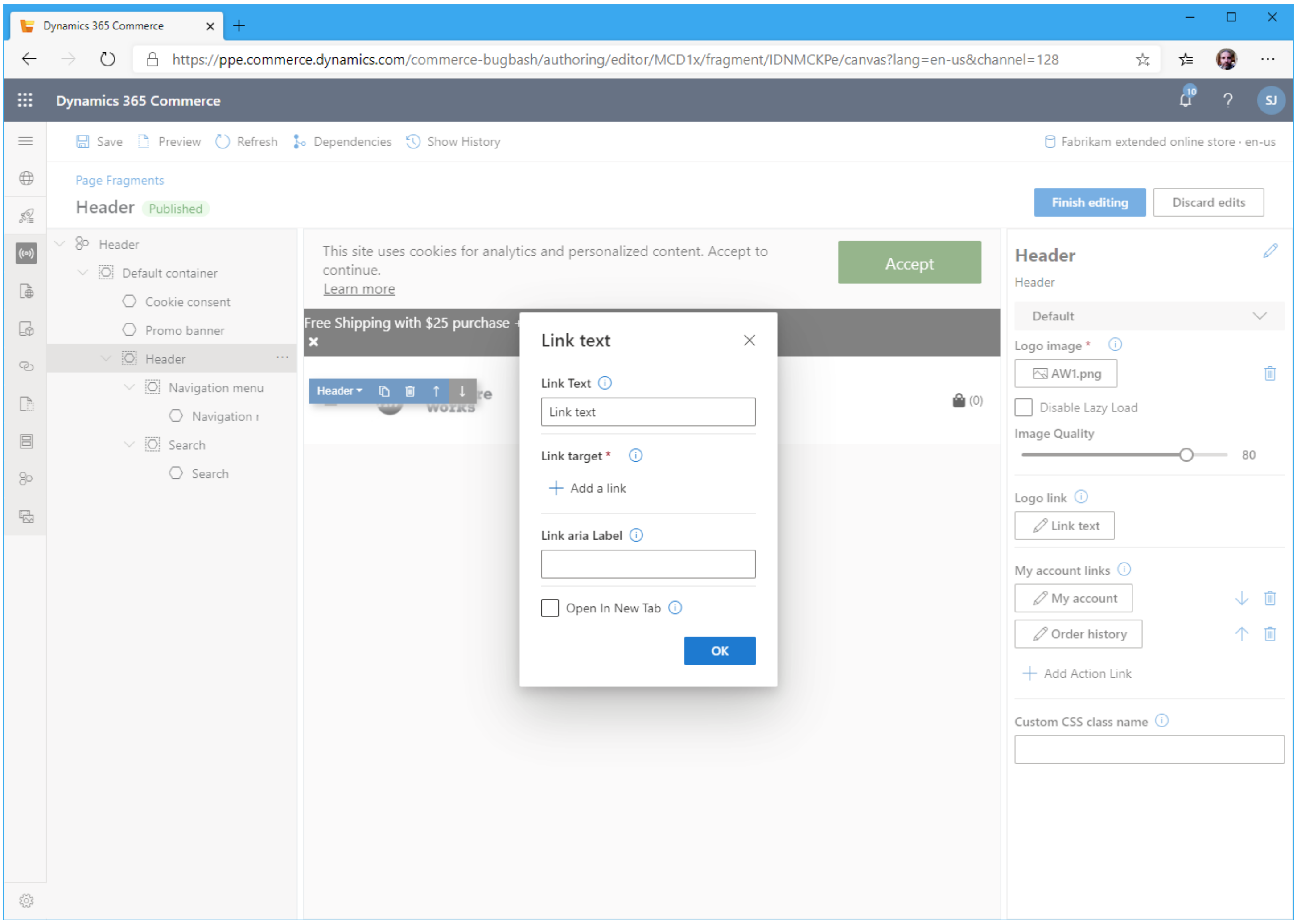
Contact	Store hours	Availability
2855 Stevens Creek Blvd Santa Clara, CA 95050	9:00 am-9:00 pm	In stock

APPENDIX C - CONFIGURE CONTENT PAGES

Now that we have a Home page, Category page and Product Details page we need to fix up a few links.

We'll start with linking the logo to navigate pack to the home page. Navigate back to the Header Fragment and select "**Edit**".

Select the "**Header**" module then select the "Logo Link" configuration property which will pop up a link picker.



Dynamics 365 Commerce
https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/fragment/IDNMCKPe/canvas?lang=en-us&channel=128
Dynamics 365 Commerce
SJ
Save
Preview
Refresh
Dependencies
Show History
Fabrikam extended online store · en-us
Page Fragments
Header
Published
Finish editing
Discard edits
Header
Default container
Cookie consent
Promo banner
Header
Navigation menu
Navigation i
Search
Search
This site uses cookies for analytics and personalized content. Accept to continue.
Learn more
Accept
Free Shipping with $25 purchase
Header
(0)

Link text

Link Text ⓘ

Link text

Link target * ⓘ

+ Add a link

Link aria Label ⓘ

☐ Open In New Tab ⓘ

OK

Header

Header

Default

Logo image * ⓘ

AW1.png

☐ Disable Lazy Load

Image Quality 80

Logo link ⓘ

Link text

My account links ⓘ

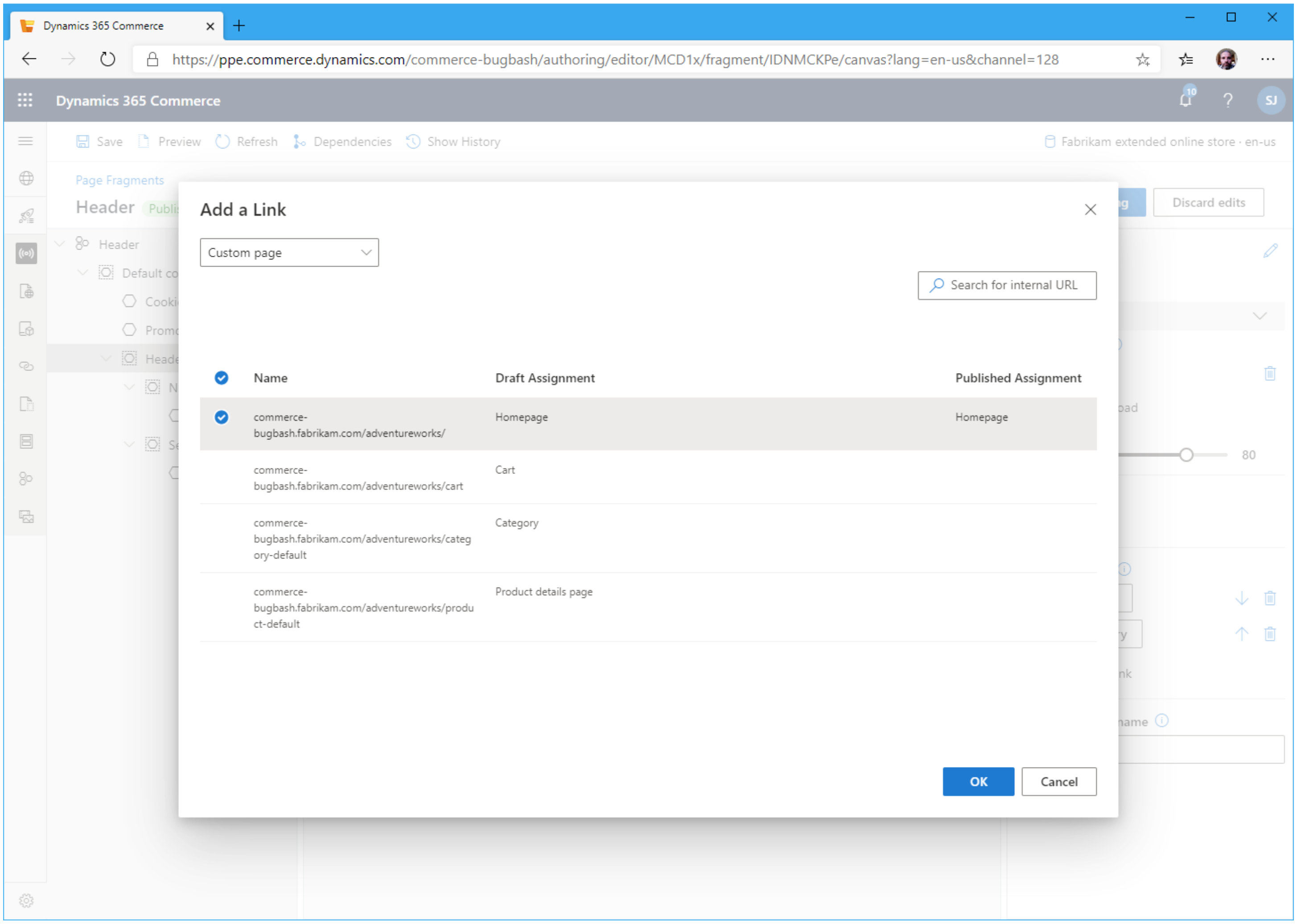
My account

Order history

+ Add Action Link

Custom CSS class name ⓘ

Select “Add a link” and choose the “Homepage” and OK. Click “Finish editing” and “Publish” then preview to test it out.



https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/fragment/IDNMCKPe/canvas?lang=en-us&channel=128

Add a Link

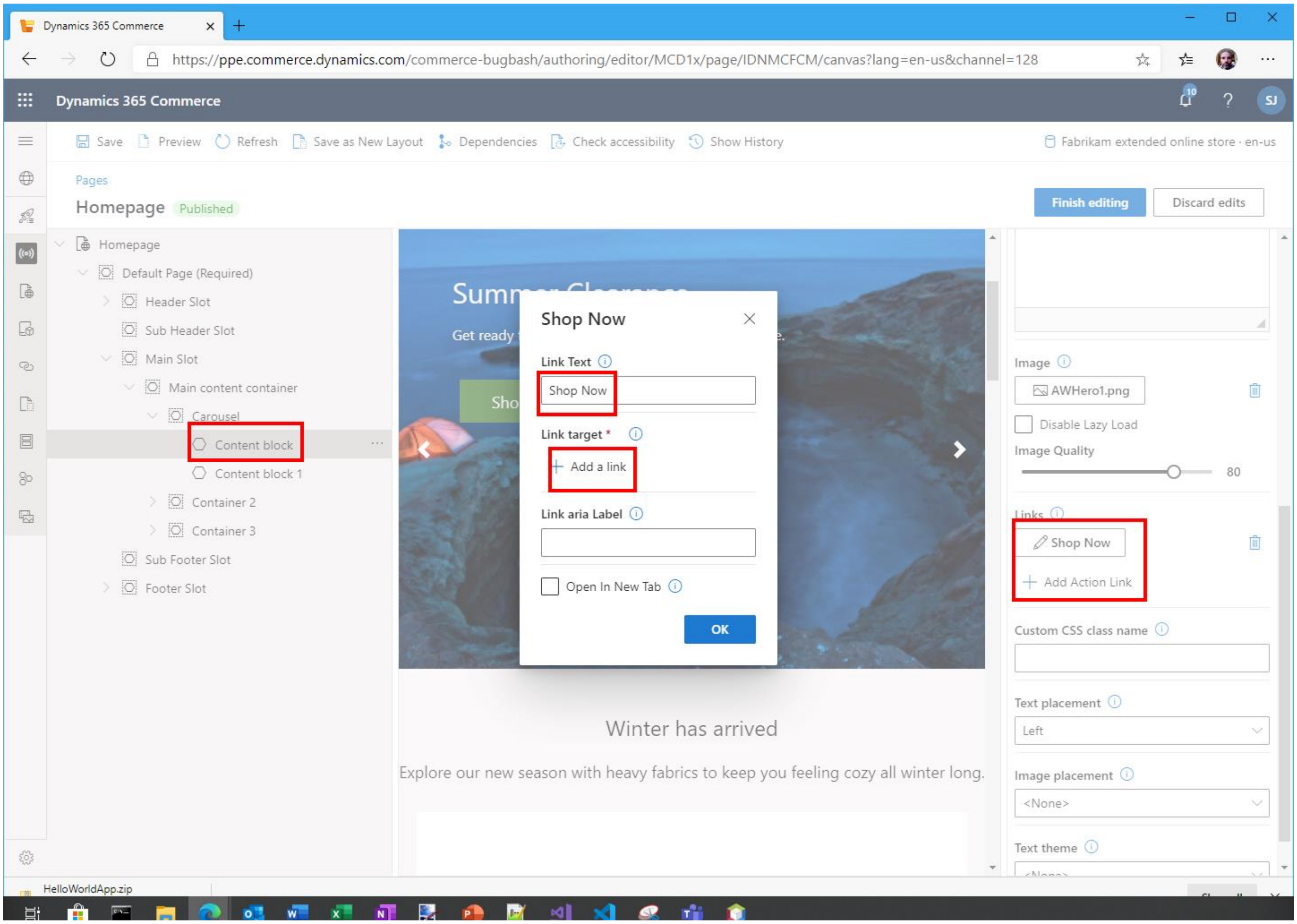
Custom page

Name	Draft Assignment	Published Assignment
commerce-bugbash.fabrikam.com/adventureworks/	Homepage	Homepage
commerce-bugbash.fabrikam.com/adventureworks/cart	Cart	
commerce-bugbash.fabrikam.com/adventureworks/category-default	Category	
commerce-bugbash.fabrikam.com/adventureworks/product-default	Product details page	

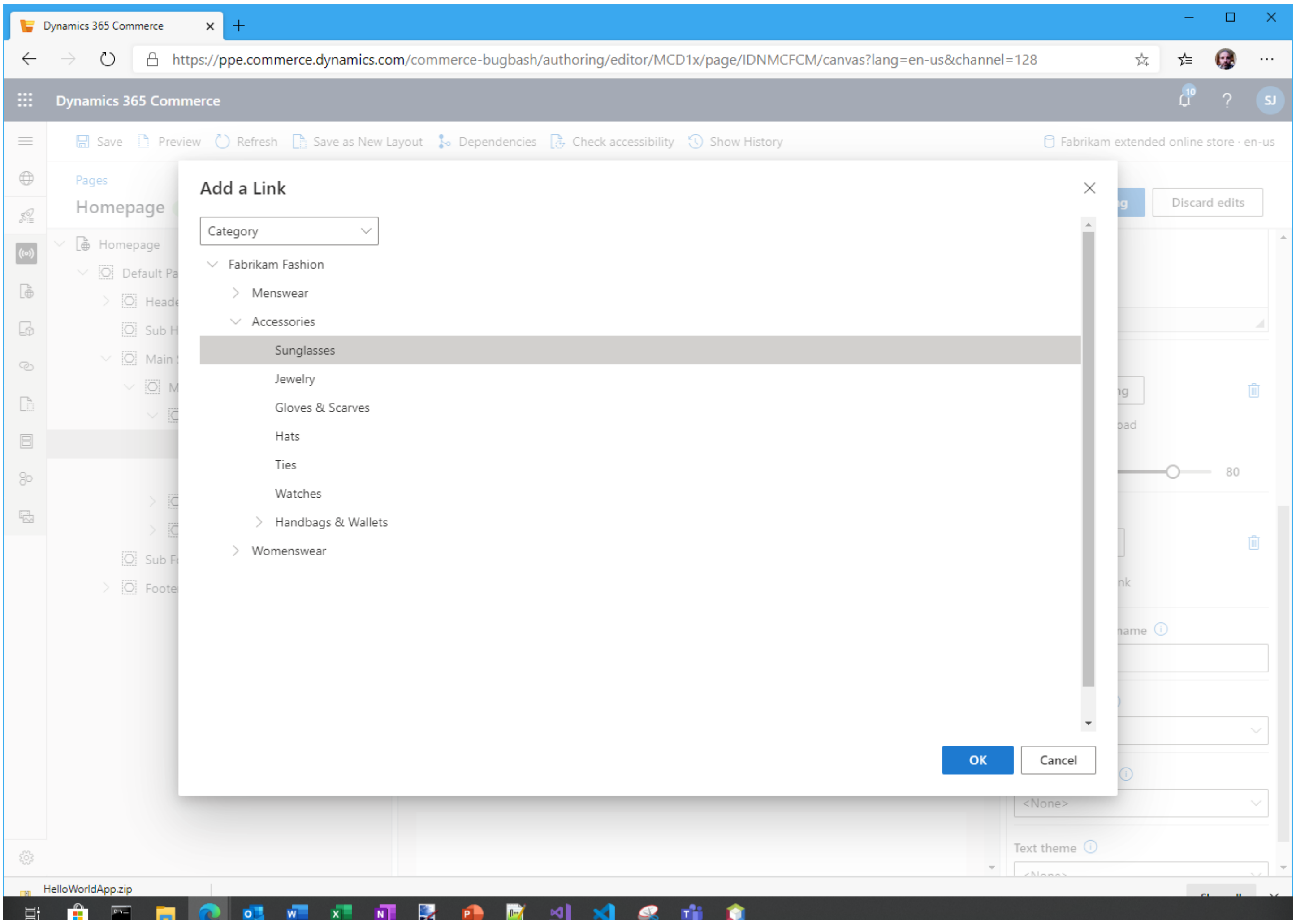
If you now preview any page and navigate around, the logo in the header will always take you back to the home page. This is a great example to see how useful fragments are, we modified it once and every page picks up the change since it’s used inside each of the templates we’ve created so far.

Next, we'll fix up links on the home page. Go back to the "**Pages**" tab and select the "**Homepage**" and "**edit**" the page.

Select the first "Content block" module and add an action link if one is not there and set the name to "**Shop Now**" and click the "**Add a link**" button to bring up the link picker.

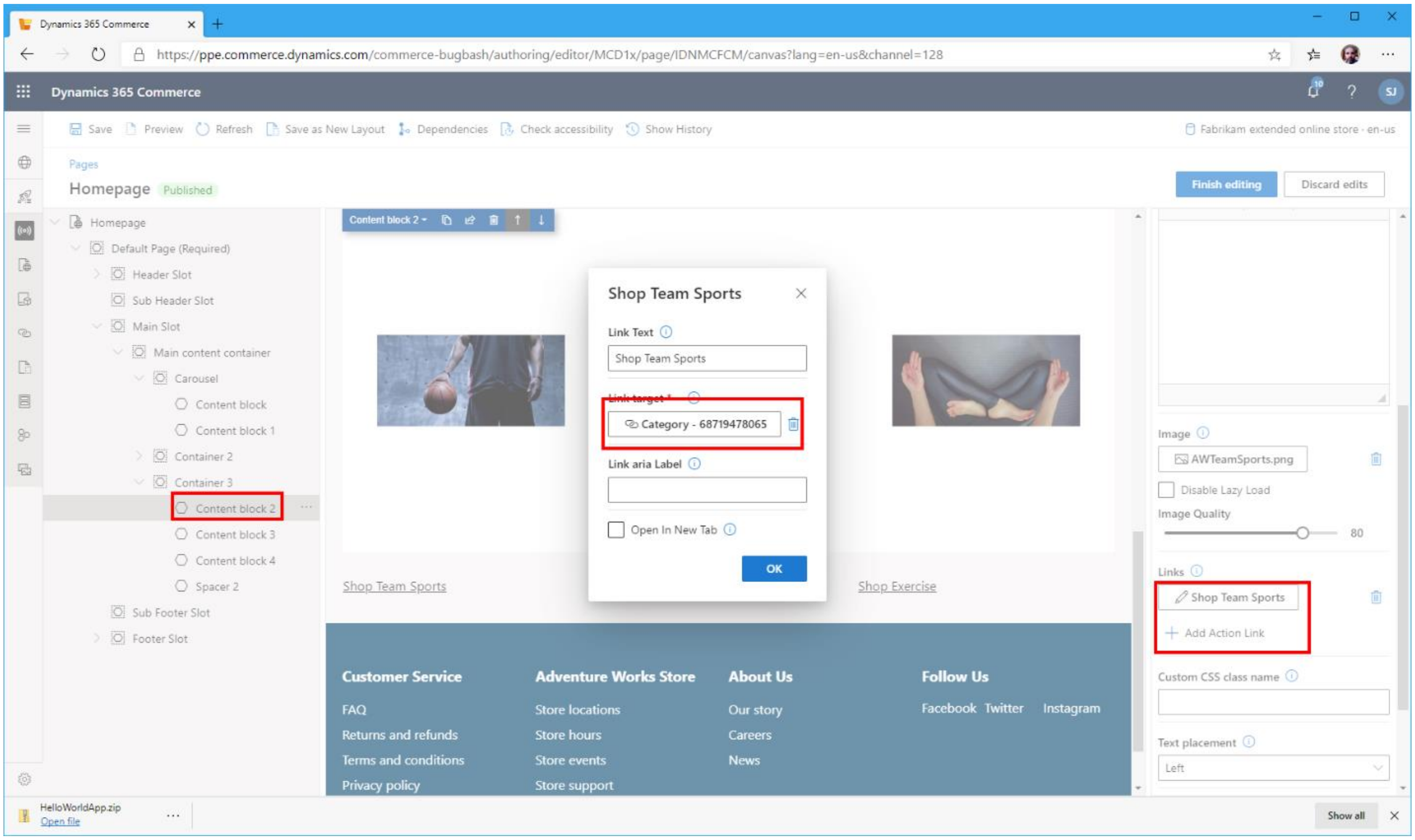


Select “**Category**” from the drop down to show category links which will then show the channel category hierarchy. Select one and click “OK”.



Similarly, select the second “**Content Block**” and select another category link.

We will next link the three content placement item modules to individual categories.



Shop Team Sports

Link Text: Shop Team Sports

Link target *: Category - 68719478065

Link aria Label:

□ Open In New Tab

Once complete, click “Finish editing” and “Preview” to test out the buttons. When satisfied, publish the page.

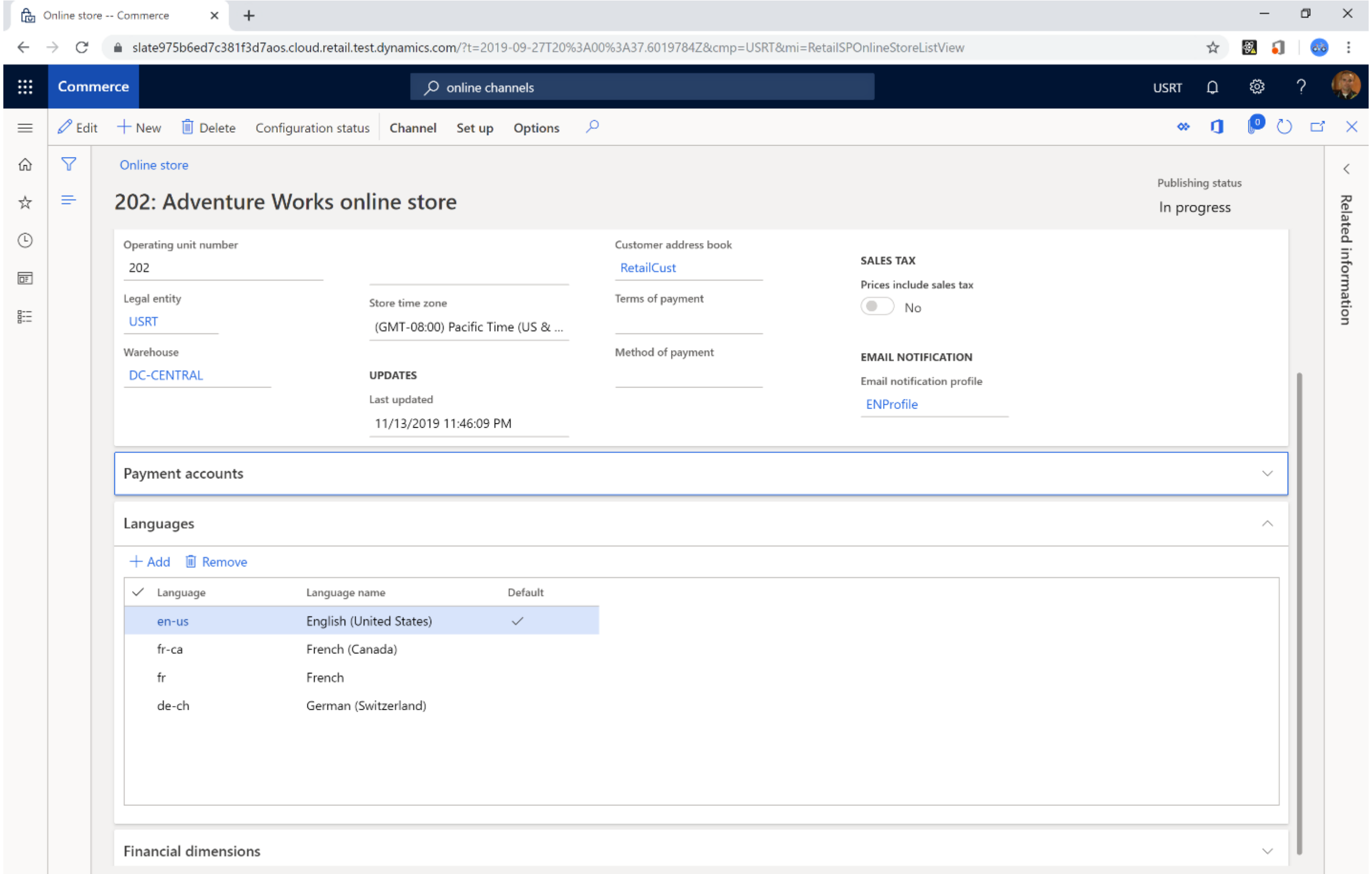
There are some links we have not configured yet like the static links in the footer. We’ll leave this as an exercise. Each page would need to be created and you should be able to re-use the “Marketing” template we created and used for our home pages to create new pages. Once the page is created and checked in you can add the link for each item in the footer as well as the “Contact” link we created in the Header module.

There are also some pages we have not shown how to build like the sign-in page, accounts management, checkout, wish list, … You should be able to look at the Fabrikam starter kit set of pages, templates and fragments to further build these out.

APPENDIX D - LOCALIZATION

You may want to localize your site to support multiple languages and markets.

From within Headquarters makes sure the online channel supports the locales you would like to support on your e-Commerce site. From within Headquarters navigate to “Online channels” then select the channel you are using. Expand the “Languages” section and add new locales as needed.



Online store -- Commerce
slate975b6ed7c381f3d7aos.cloud.retail.test.dynamics.com/?t=2019-09-27T20%3A00%3A37.6019784Z&cmp=USRT&mi=RetailSPOnlineStoreListView
Commerce
online channels
USRT
Edit
New
Delete
Configuration status
Channel
Set up
Options
Related information

Online store

202: Adventure Works online store

Publishing status: In progress

Operating unit number: 202

Legal entity: USRT

Warehouse: DC-CENTRAL

Store time zone: (GMT-08:00) Pacific Time (US & ...

UPDATES

Last updated: 11/13/2019 11:46:09 PM

Customer address book: RetailCust

Terms of payment:

Method of payment:

SALES TAX

Prices include sales tax: No

EMAIL NOTIFICATION

Email notification profile: ENProfile

Payment accounts

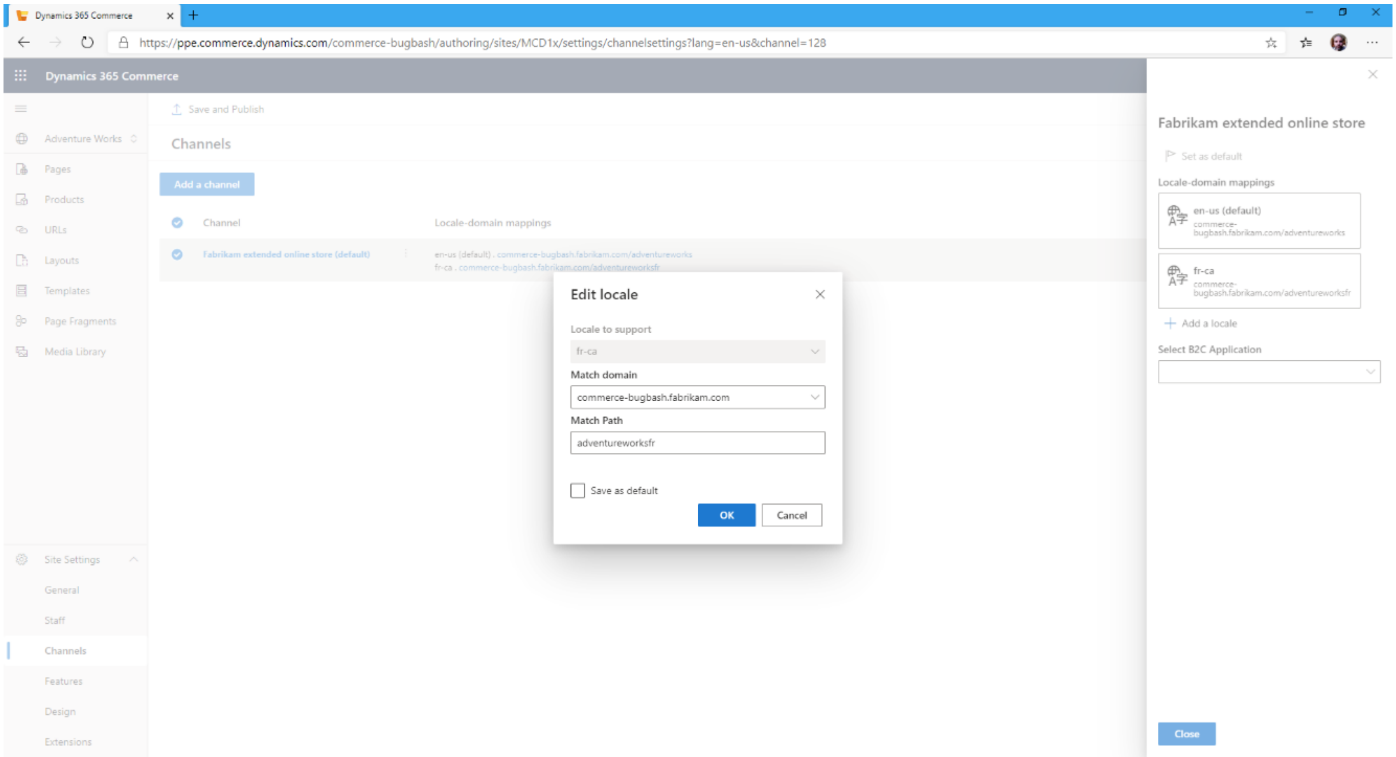
Languages

+ Add 🗑 Remove

Language	Language name	Default
en-us	English (United States)	✓
fr-ca	French (Canada)	
fr	French	
de-ch	German (Switzerland)	

Financial dimensions

From within the Site Builder tool, select the "**Channels**" tab under the "**Site Settings**" and select the Channel and click the "**Add a locale**" button. You should see all available locale's in the "Locale to support" drop down.



Channel	Locale-domain mappings
Fabrikam extended online store (default)	en-us (default) . commerce-bugbash.fabrikam.com/adventureworks fr-ca . commerce-bugbash.fabrikam.com/adventureworksfr

Edit locale

Locale to support: fr-ca

Match domain: commerce-bugbash.fabrikam.com

Match Path: adventureworksfr

□ Save as default

Fabrikam extended online store

Set as default

Locale-domain mappings

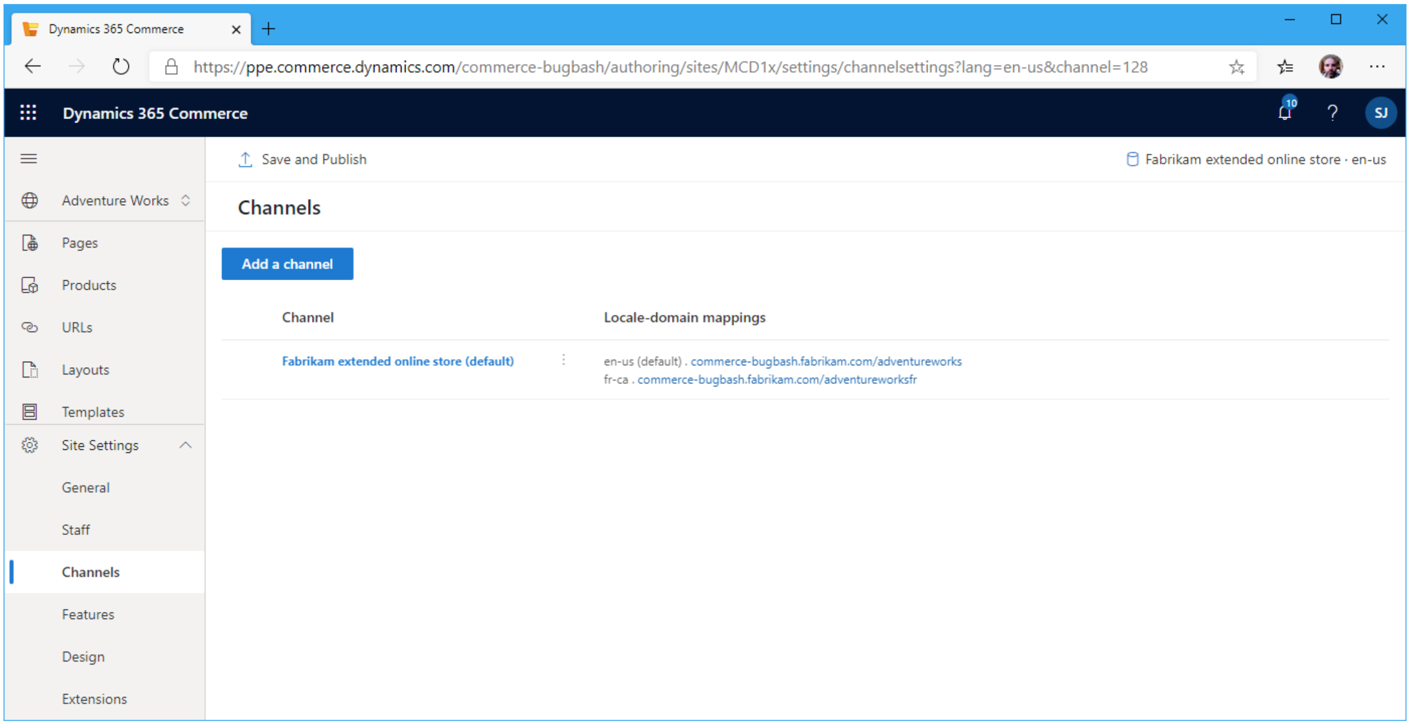
en-us (default)
commerce-bugbash.fabrikam.com/adventureworks

fr-ca
commerce-bugbash.fabrikam.com/adventureworksfr

+ Add a locale

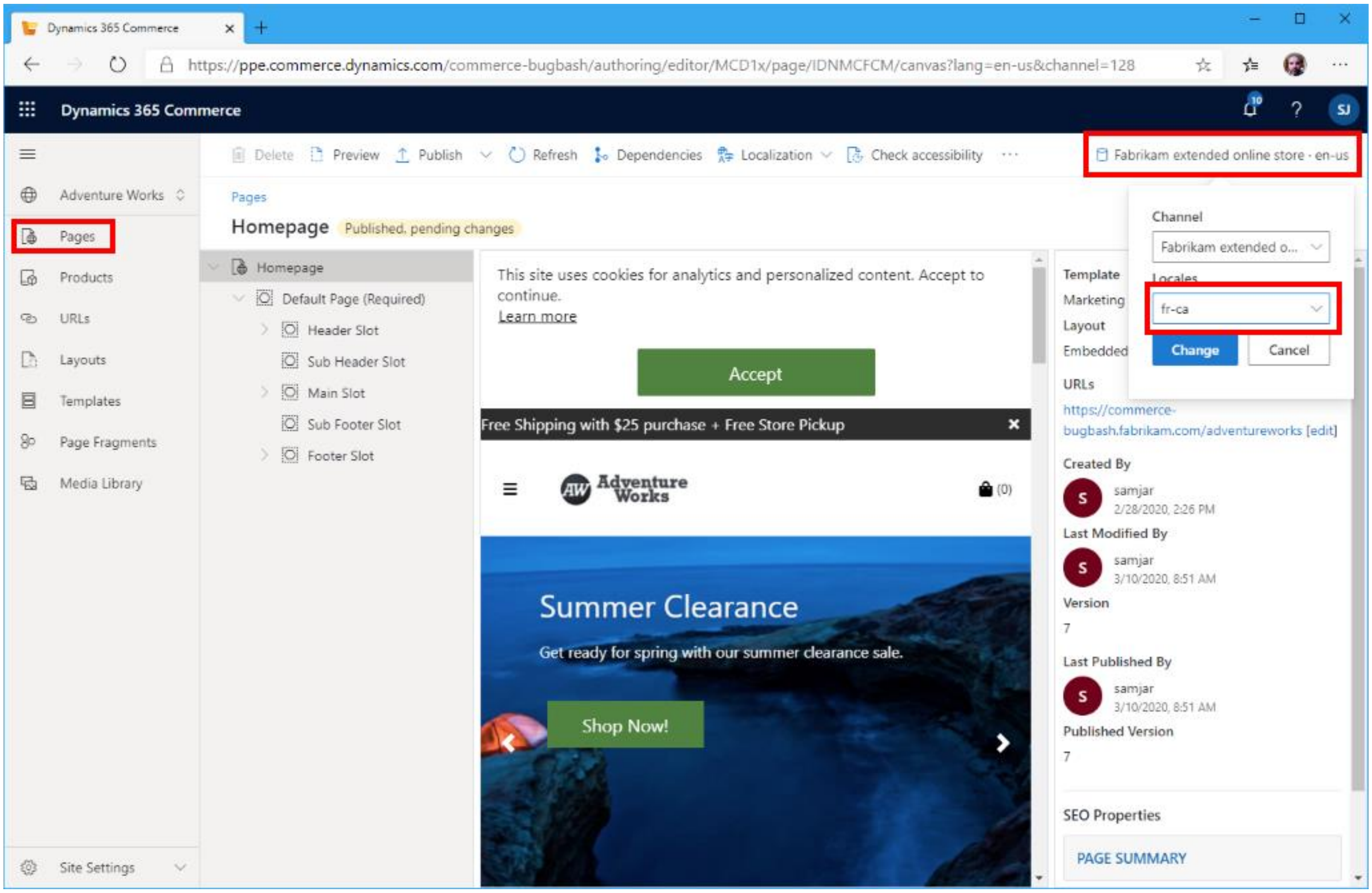
Select B2C Application

You should now see multiple Locale-domain mappings listed next to the channel as shown below. Once added, select "Save and Publish" at the top.



Channel	Locale-domain mappings
Fabrikam extended online store (default)	en-us (default) . commerce-bugbash.fabrikam.com/adventureworks fr-ca . commerce-bugbash.fabrikam.com/adventureworksfr

If you navigate back to a page in site builder, there is a drop down in the top right where you can switch the supported locale and configure the localized version of the page.


https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/page/IDNMCFCM/canvas?lang=en-us&channel=128

Homepage Published, pending changes

Channel: Fabrikam extended o...

Locales: fr-ca

This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Free Shipping with $25 purchase + Free Store Pickup

Summer Clearance

Get ready for spring with our summer clearance sale.

https://commerce-bugbash.fabrikam.com/adventureworks [edit]

Created By: samjar 2/28/2020, 2:26 PM

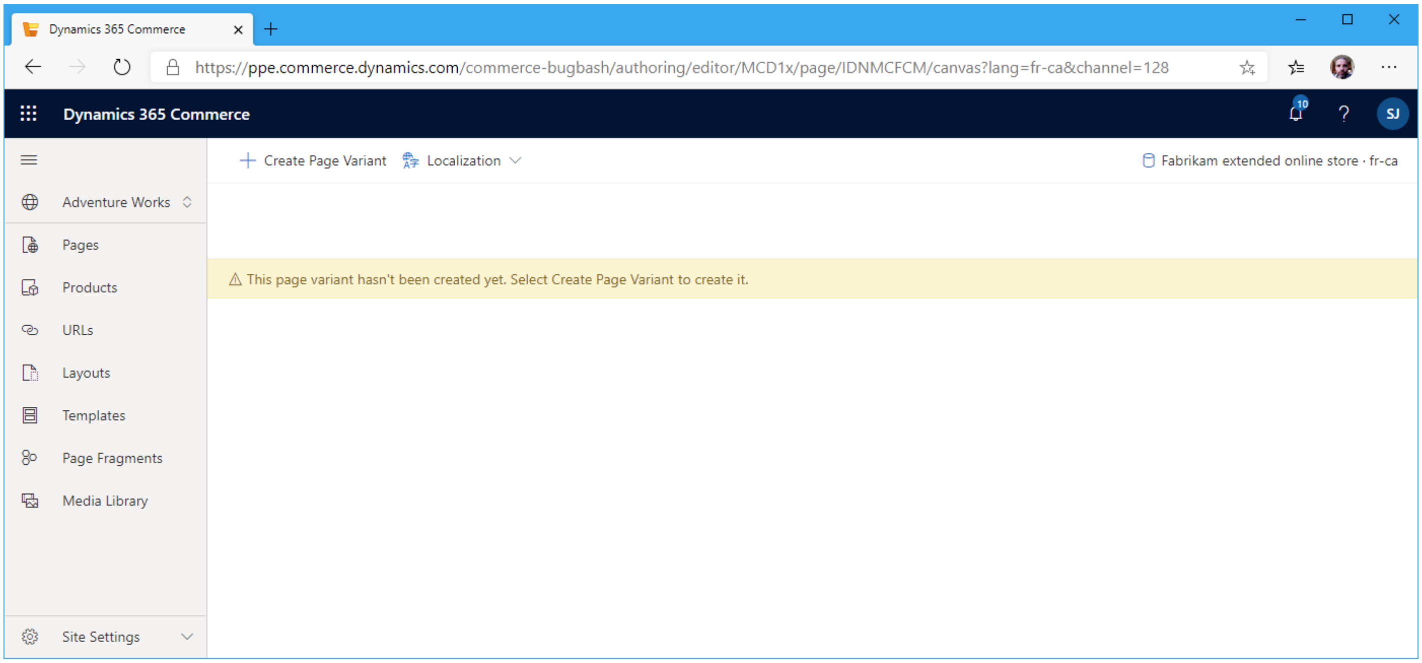
Last Modified By: samjar 3/10/2020, 8:51 AM

Version: 7

Last Published By: samjar 3/10/2020, 8:51 AM

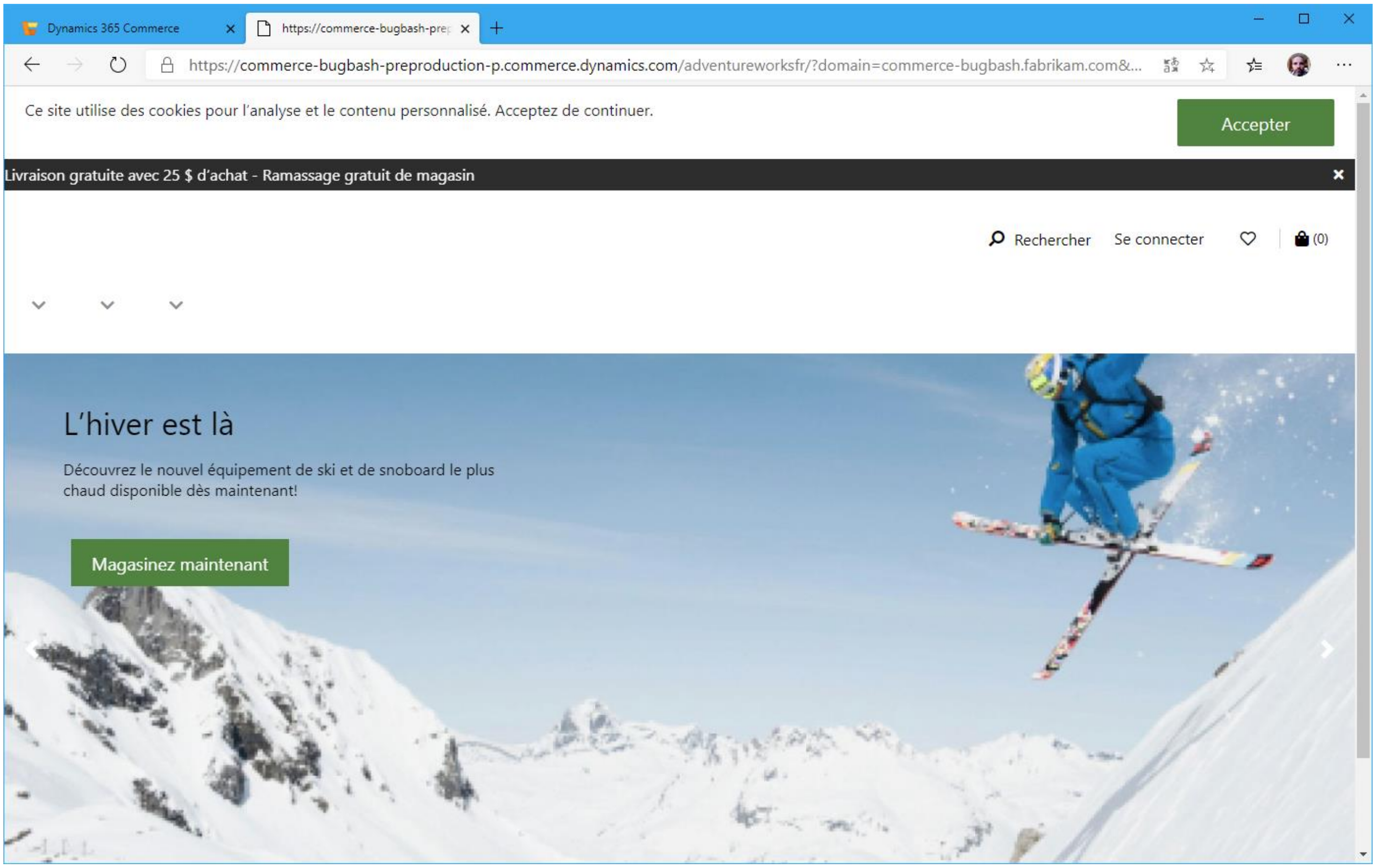
Published Version: 7

Select the locale you would like to configure and click the "**Change**" button. If it's the first time the page is being configured, you will see a message that the page variant hasn't been created yet. Select "**Create Page variant**" button.


https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/page/IDNMCFCM/canvas?lang=fr-ca&channel=128

⚠ This page variant hasn't been created yet. Select Create Page Variant to create it.

You can now configure the page as desired. The page can match the other locales or can be completely different if desired. Once complete, click “**Finish editing**” and publish the changes. Similarly, you’ll also need to create a localized variant of the Header and Footer fragments. Once those are complete and published the rendered page should look similar to the below image. Notice how the navigation hierarchy names are not showing up. This is because the names are not localized in Retail server.



Dynamics 365 Commerce
https://commerce-bugbash-pre
https://commerce-bugbash-preproduction-p.commerce.dynamics.com/adventureworksfr/?domain=commerce-bugbash.fabrikam.com&...

Ce site utilise des cookies pour l’analyse et le contenu personnalisé. Acceptez de continuer.

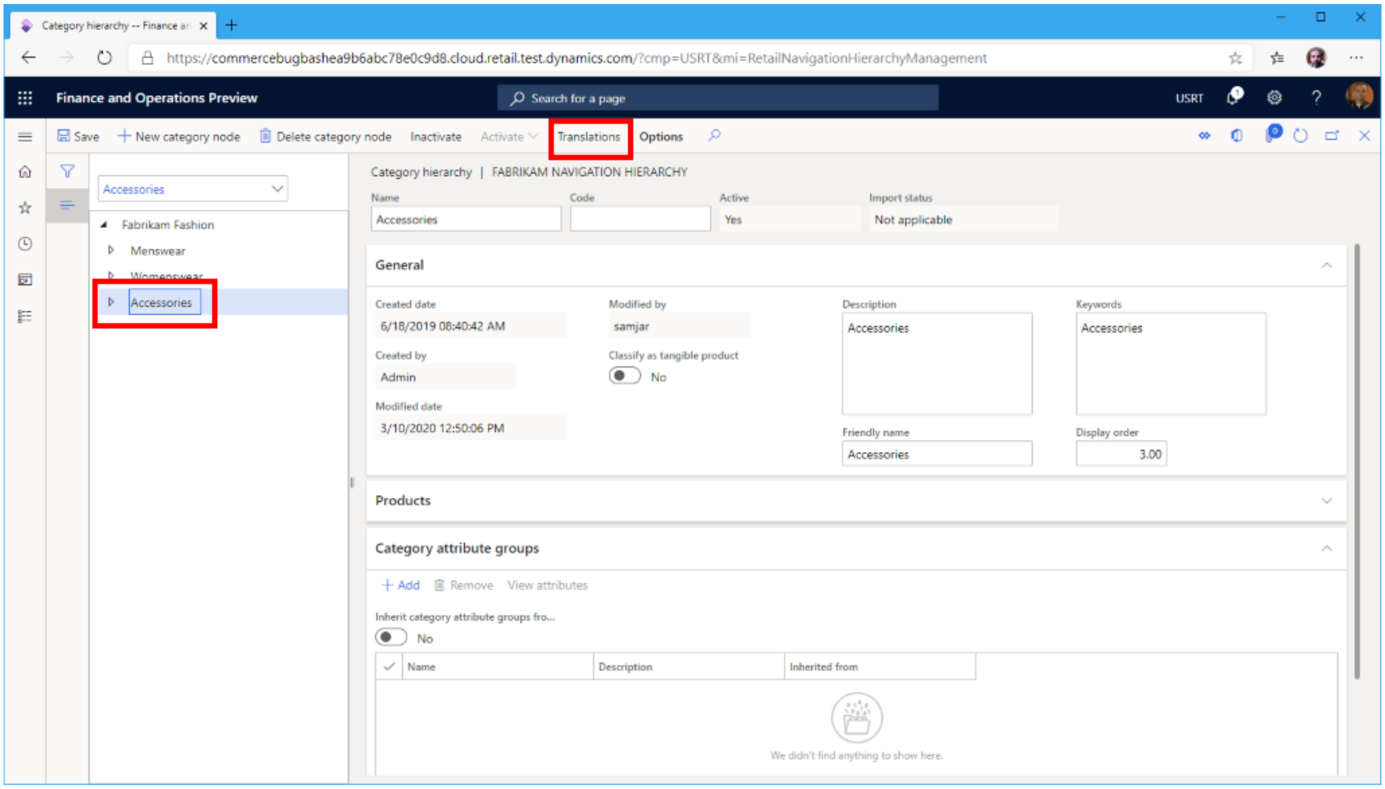
Livraison gratuite avec 25 $ d’achat - Ramassage gratuit de magasin

Rechercher
Se connecter
(0)

L’hiver est là

Découvrez le nouvel équipement de ski et de snoboard le plus chaud disponible dès maintenant!

To localize the category names, launch the Headquarters site and search for “**Commerce navigation categories**” and select the channel hierarchy you are using. Select “Edit category hierarchy” button at the top. Select each node and click the “Edit” button, followed by the **Translations** link.

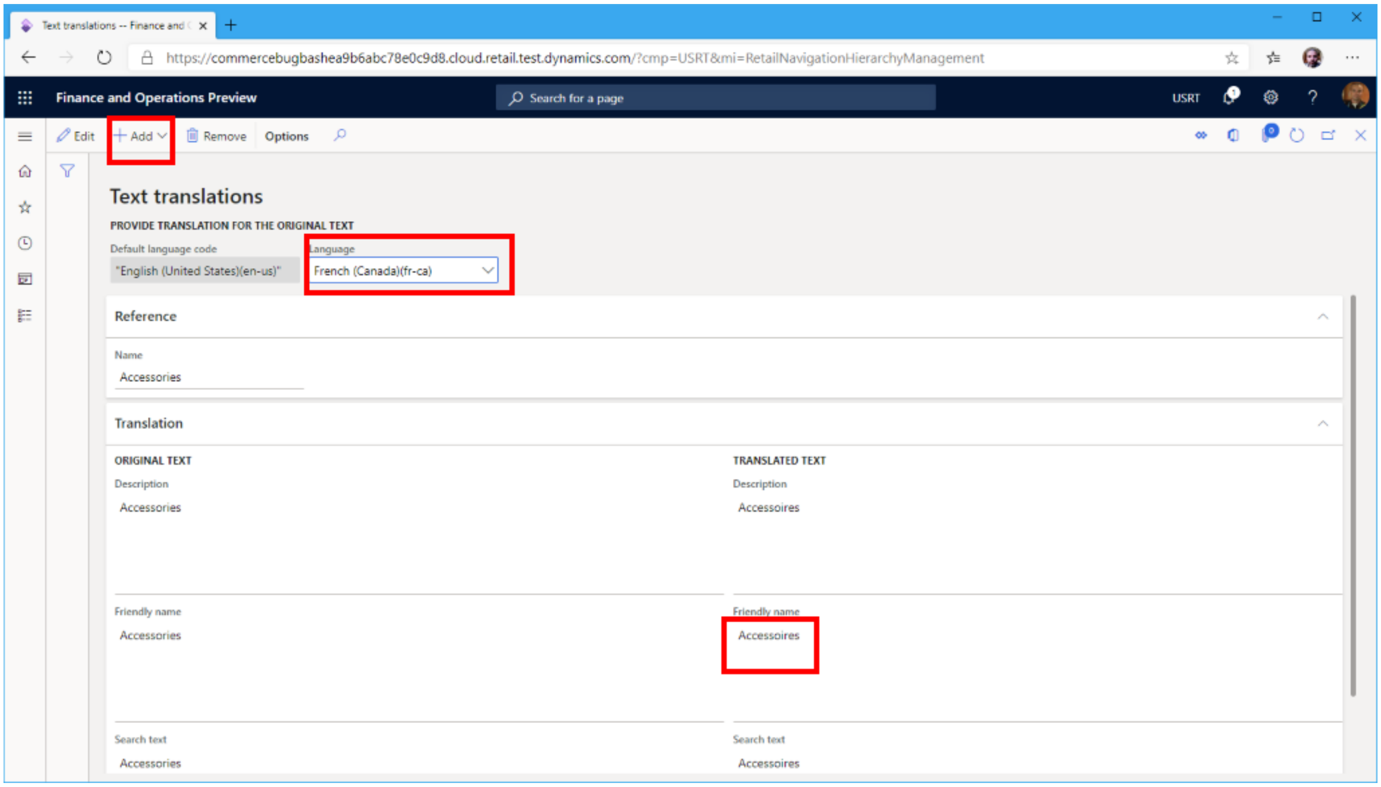

Category hierarchy -- Finance an
https://commercebugbashea9b6abc78e0c9d8.cloud.retail.test.dynamics.com/?cmp=USRT&mi=RetailNavigationHierarchyManagement
Finance and Operations Preview
Search for a page
USRT
Save
New category node
Delete category node
Inactivate
Activate
Translations
Options
Accessories
Fabrikam Fashion
Menswear
Womenswear
Accessories
Category hierarchy | FABRIKAM NAVIGATION HIERARCHY
General
Products
Category attribute groups
Add
Remove
View attributes
Name
Description
Inherited from
We didn't find anything to show here.

Name: Accessories
Code:
Active: Yes
Import status: Not applicable

Created date: 6/18/2019 08:40:42 AM
Modified by: samjar
Description: Accessories
Keywords: Accessories
Created by: Admin
Classify as tangible product: No
Modified date: 3/10/2020 12:50:06 PM
Friendly name: Accessories
Display order: 3.00

Inherit category attribute groups fro...: No

Select the appropriate locale in the “**Language**” drop down or if it is not listed add it with the “Add” button at the top. In the Translated text section, change the “**Friendly name**” of the category to the localized version.


Text translations -- Finance and O
https://commercebugbashea9b6abc78e0c9d8.cloud.retail.test.dynamics.com/?cmp=USRT&mi=RetailNavigationHierarchyManagement
Finance and Operations Preview
Search for a page
USRT
Edit
Add
Remove
Options

Text translations

PROVIDE TRANSLATION FOR THE ORIGINAL TEXT

Default language code: "English (United States)(en-us)"

Language: French (Canada)(fr-ca)

Reference

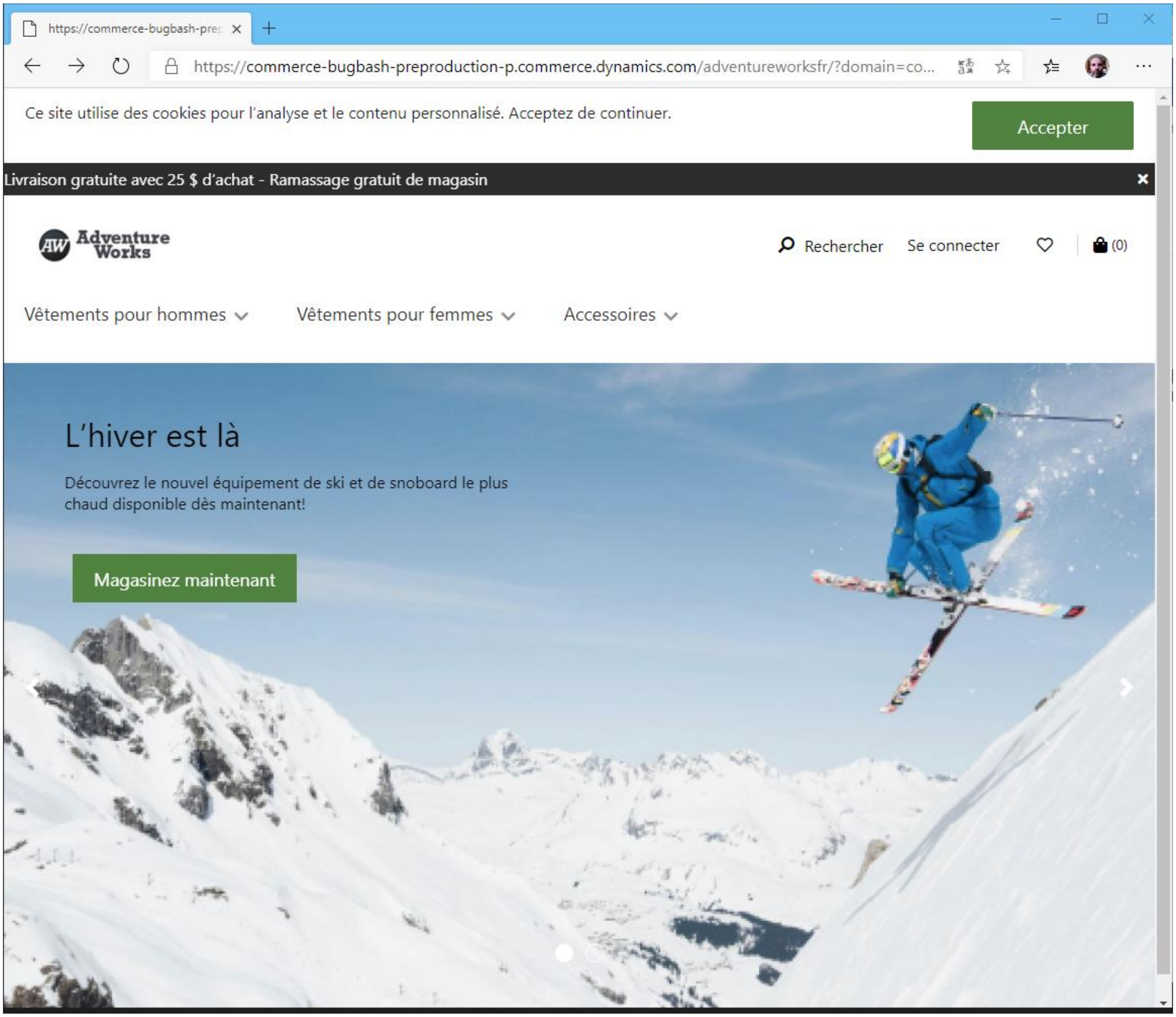
Name: Accessories

Translation

	ORIGINAL TEXT	TRANSLATED TEXT
Description	Accessories	Accessoires
Friendly name	Accessories	Accessoires
Search text	Accessories	Accessoires

When complete click save and to ensure all the data gets sync'd to the cloud scale unit.

- Search for "Channel categories and product attributes", select the channel and click the "Publish channel updates" button.
- Search for "Channel Database" and select the appropriate Channel Database ID from the list then run a "Full data sync" 9999.


https://commerce-bugbash-prep
https://commerce-bugbash-preproduction-p.commerce.dynamics.com/adventureworksfr/?domain=co...
Accepter
Adventure Works
Rechercher
Se connecter
(0)
Vêtements pour hommes
Vêtements pour femmes
Accessoires
Magasinez maintenant

Ce site utilise des cookies pour l'analyse et le contenu personnalisé. Acceptez de continuer.

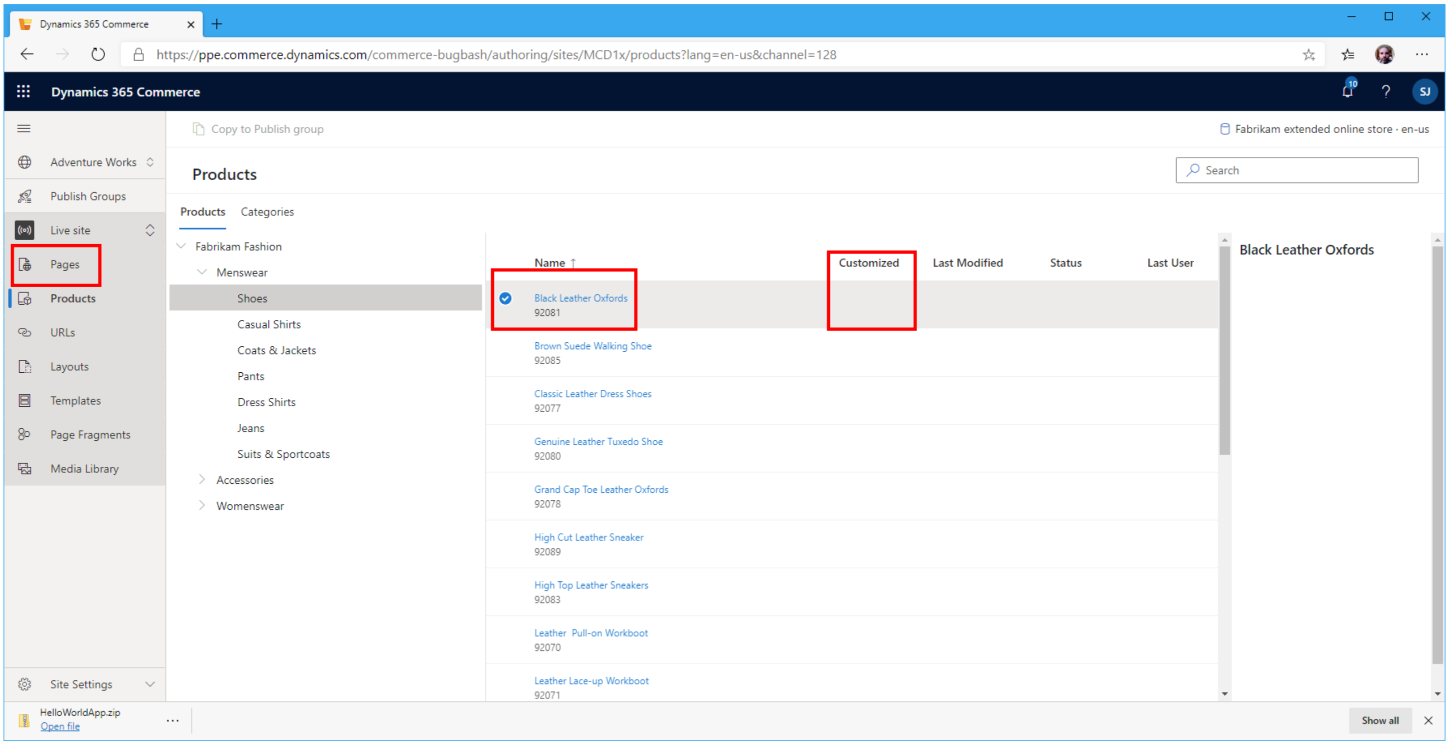
Livraison gratuite avec 25 $ d'achat - Ramassage gratuit de magasin

L'hiver est là

Découvrez le nouvel équipement de ski et de snoboard le plus chaud disponible dès maintenant!

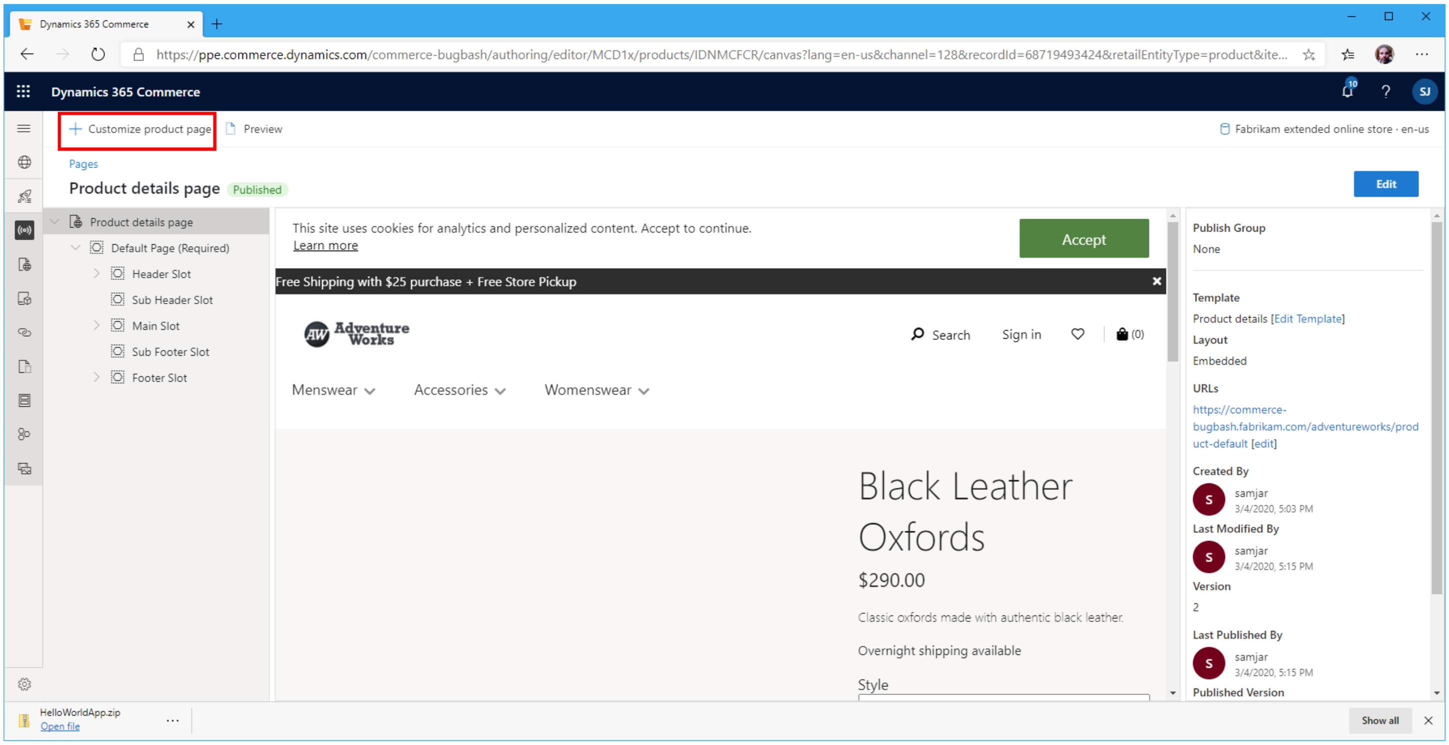
APPENDIX E - CUSTOMIZING A PRODUCT DETAIL PAGE

By default, all product details pages derive from the same root page, there are times when we want to make a product special and stand out. This is where customizing a page comes in. To customize a page, navigate to the "Products" tab and find the product you want to customize, and notice the customized column should be empty.



Name ↑	Customized	Last Modified	Status	Last User
Black Leather Oxfords 92081				
Brown Suede Walking Shoe 92085				
Classic Leather Dress Shoes 92077				
Genuine Leather Tuxedo Shoe 92080				
Grand Cap Toe Leather Oxfords 92078				
High Cut Leather Sneaker 92089				
High Top Leather Sneakers 92083				
Leather Pull-on Workboot 92070				
Leather Lace-up Workboot 92071				

Select the link to the product. This will load the product in the page editor, select the top "Customize product page" button.



This site uses cookies for analytics and personalized content. Accept to continue.
Learn more

Free Shipping with $25 purchase + Free Store Pickup

Black Leather Oxfords

$290.00

Classic oxfords made with authentic black leather.

Overnight shipping available

Publish Group

None

Template

Product details [Edit Template]

Layout

Embedded

URLs

https://commerce-bugbash.fabrikam.com/adventureworks/product-default [edit]

Created By

samjar
3/4/2020, 5:03 PM

Last Modified By

samjar
3/4/2020, 5:15 PM

Version

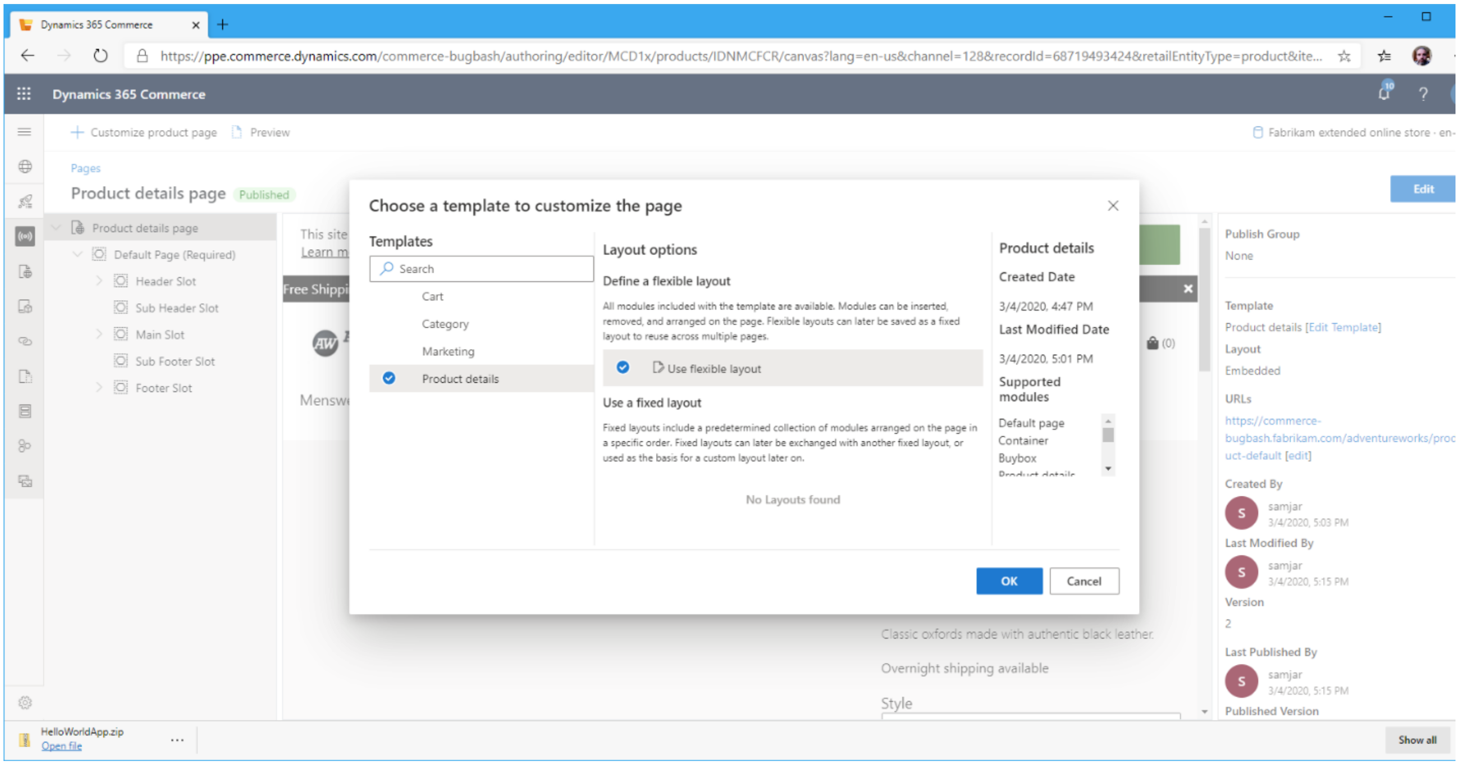
2

Last Published By

samjar
3/4/2020, 5:15 PM

Published Version

We will then be asked which Template we'd like the new page to adhere to. We could build out new templates, but we'll pick the Product details page we already created.



Choose a template to customize the page

Templates

Search

Cart

Category

Marketing

Product details

Layout options

Define a flexible layout

All modules included with the template are available. Modules can be inserted, removed, and arranged on the page. Flexible layouts can later be saved as a fixed layout to reuse across multiple pages.

Use flexible layout

Use a fixed layout

Fixed layouts include a predetermined collection of modules arranged on the page in a specific order. Fixed layouts can later be exchanged with another fixed layout, or used as the basis for a custom layout later on.

No Layouts found

Product details

Created Date

3/4/2020, 4:47 PM

Last Modified Date

3/4/2020, 5:01 PM

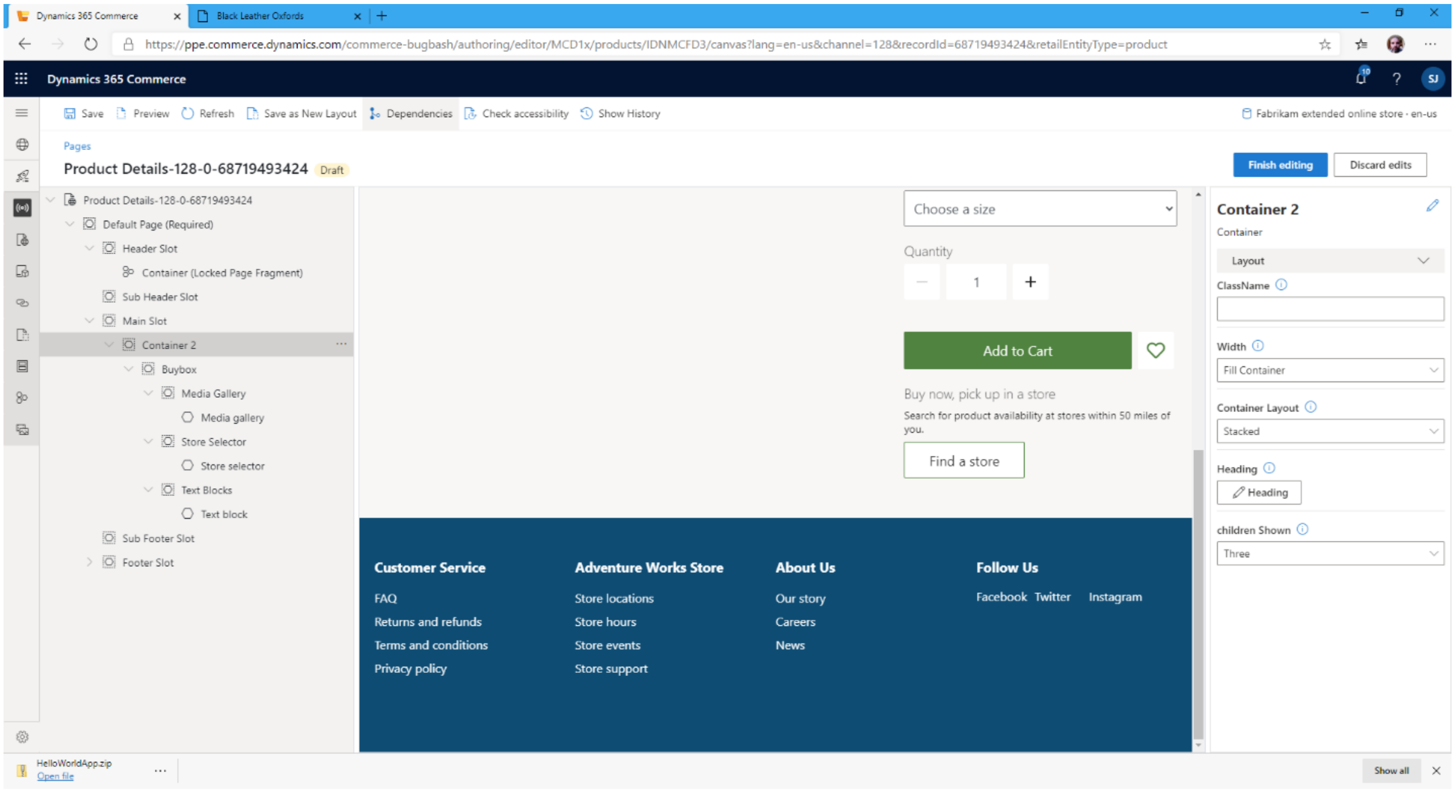
Supported modules

Default page
Container
Buybox
Product details

OK   Cancel

We can now edit the page as we like and make any modifications desired for this page. Make some changes and save/preview/check In and Publish when complete

Notice in below example, we have a streamlined PDP with no ratings/reviews or collections, but additional modules can also be added. Note the template may restrict to only a select set of modules, so you may need to modify the PDP template or create a new one for customized pages.

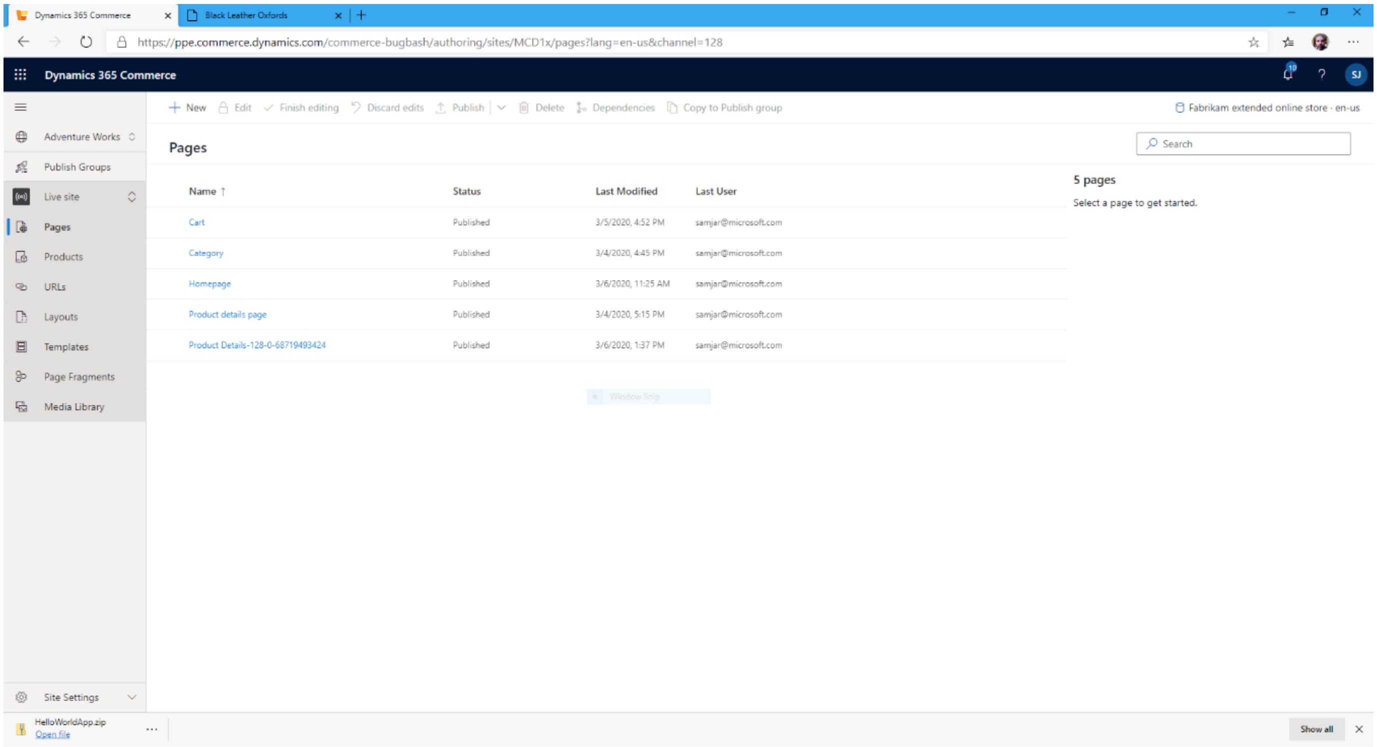


https://ppe.commerce.dynamics.com/commerce-bugbash/authoring/editor/MCD1x/products/IDNMCFD3/canvas?lang=en-us&channel=128&recordId=68719493424&retailEntityType=product

Product Details-128-0-68719493424 Draft

Container 2
Container
Width: Fill Container
Container Layout: Stacked
children Shown: Three

Select “Finish editing” and publish when complete. If you now navigate around your site, you should find that only the new PDP will load up as customized and all of the others will continue to use the default page. You’ll also find this new page listed when you select the “Pages” tab on the left side as shown below.



Pages

Name ↑	Status	Last Modified	Last User
Cart	Published	3/5/2020, 4:52 PM	samjar@microsoft.com
Category	Published	3/4/2020, 4:45 PM	samjar@microsoft.com
Homepage	Published	3/6/2020, 11:25 AM	samjar@microsoft.com
Product details page	Published	3/4/2020, 5:15 PM	samjar@microsoft.com
Product Details-128-0-68719493424	Published	3/6/2020, 1:37 PM	samjar@microsoft.com

5 pages

Select a page to get started.