



Adobe Summit 2017 Lab 3846:



Fuel your mobile apps with assets and content

Adobe Experience Manager Mobile

Anthony Rumsey and John Metzger



Table of Contents

<u>INTRODUCTION</u>	4
<u>GOAL</u>	4
<u>OBJECTIVES</u>	4
<u>MODULE 1 – SETUP AND INTRODUCTION</u>	5
LESSON 1 – AEM BASICS	5
OVERVIEW	5
OBJECTIVES	5
EXERCISE 1 – START AEM	5
EXERCISE 2 – INSTALL LAB CONTENT	6
EXERCISE 3 – SIGN IN TO AEM	6
EXERCISE 4 – CONTENT SERVICES	6
SUMMARY	7
<u>MODULE 2 – MOBILE APP MANAGEMENT</u>	8
LESSON 1 – APP SPACES	8
OVERVIEW	8
OBJECTIVES	8
EXERCISE 1 – CREATE A NEW SPACE	8
LESSON 2 – APP CATALOG	8
OVERVIEW	8
OBJECTIVES	8
EXERCISE 1 – CREATE A NEW APP	8
EXERCISE 2 – EDIT APP META DATA	9
EXERCISE 3 – ADD SPACES TO APP	11
SUMMARY	12
<u>MODULE 3 – EXTEND EXISTING CONTENT</u>	13
LESSON 1 – PAGES	13
OVERVIEW	13
OBJECTIVES	13
EXERCISE 1 – EXPLORE SITE CONTENT (OPTIONAL)	13
EXERCISE 2 – CREATE NEW PAGES ENTITY	13
EXERCISE 3 – EDIT PAGES ENTITY	14
LESSON 2 – ASSETS	14
OVERVIEW	14
OBJECTIVES	14
EXERCISE 1 – CREATE NEW ASSETS ENTITY	15
EXERCISE 2 – EDIT ASSETS ENTITY	15



SUMMARY	16
<u>MODULE 4 – MANAGING MODELS</u>	17
LESSON 1 – GROUP MODELS	17
OVERVIEW	17
OBJECTIVES	17
EXERCISE 1 – CREATE A NEW GROUP MODEL	17
EXERCISE 2 – EDIT MODEL PROPERTIES	18
LESSON 2 – PRIMITIVE MODELS	18
OVERVIEW	18
OBJECTIVES	18
EXERCISE 1 – CREATE A NEW MODEL	18
EXERCISE 2 – EDIT MODEL PROPERTIES	19
EXERCISE 3 – EDIT MODEL DEFINITION	19
SUMMARY	20
<u>MODULE 5 – MANAGING ENTITIES</u>	21
LESSON 1 – ENTITY CREATION	21
OVERVIEW	21
OBJECTIVES	21
EXERCISE 1 – CREATE NEW ENTITY	21
EXERCISE 2 – EDIT ENTITY IMAGE	21
EXERCISE 3 – EDIT ENTITY DATA	21
EXERCISE 4 – ENTITY REFERENCES	22
EXERCISE 5 – ENTITY PREVIEW	23
LESSON 2 – ENTITY PUBLISHING	23
OVERVIEW	23
OBJECTIVES	23
EXERCISE 1 – PUBLISH ENTITY	24
EXERCISE 2 – PUBLISH SPACE	24
SUMMARY	24
<u>MODULE 6 – MOBILE APPLICATION (OPTIONAL)</u>	25
LESSON 1 – RUN MOBILE APPLICATION	25
OVERVIEW	25
OBJECTIVES	25
EXERCISE 1 – INSTALL ADDITIONAL CONTENT	25
EXERCISE 2 – PUBLISH CONTENT	25
EXERCISE 3 – RUN APP	25
EXERCISE 4 – PUBLISH CHANGES	26
SUMMARY	26



Introduction

Adobe Experience Manager Mobile provides a complete toolset to rapidly build, manage, and deliver mobile apps that integrate with the Adobe Marketing Cloud so you can measure and optimize app performance. Leverage existing content from multiple sources including the easy-to-use authoring environment of AEM and deliver updates without requiring app rebuilds or resubmissions. Integrate with third-party APIs to extend functionality and connect to critical business systems. Use built-in messaging and analytics to communicate with your audience and track user activity and engagement.

Goal

Explore the content services features introduced with Adobe Experience Manager Mobile to easily deliver content to existing mobile applications. Leverage an existing investment in AEM by extending the use of pages and assets in a mobile app. Discover the data modeling capabilities introduced in AEM 6.3 that enable authors to easily create their own structured content.

Objectives

- Launch and review AEM Mobile
- Create a new entry in the mobile app catalog
- Explore the AEM Mobile dashboard
- Edit metadata of a mobile app
- Re-use pages in a mobile application
- Re-use assets in a mobile application
- Create new data models
- Author structured content
- Publish structured content



Module 1 – Setup and Introduction

This module introduces the Content Services app catalog of Adobe Experience Manager Mobile.

Lesson 1 – AEM Basics

Overview

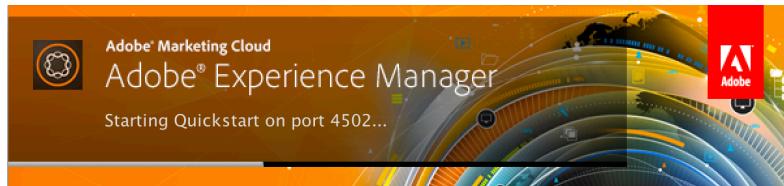
Adobe® Experience Manager helps you organize, create, and manage the delivery of creative assets and other content across your digital marketing channels, including web, mobile, email, communities, and video.

Objectives

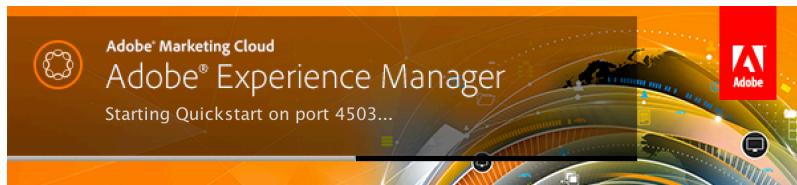
- Start an AEM author instance
- Start an AEM publish instance
- Install content packages to AEM
- Navigate to the Content Services app catalog

Exercise 1 – Start AEM

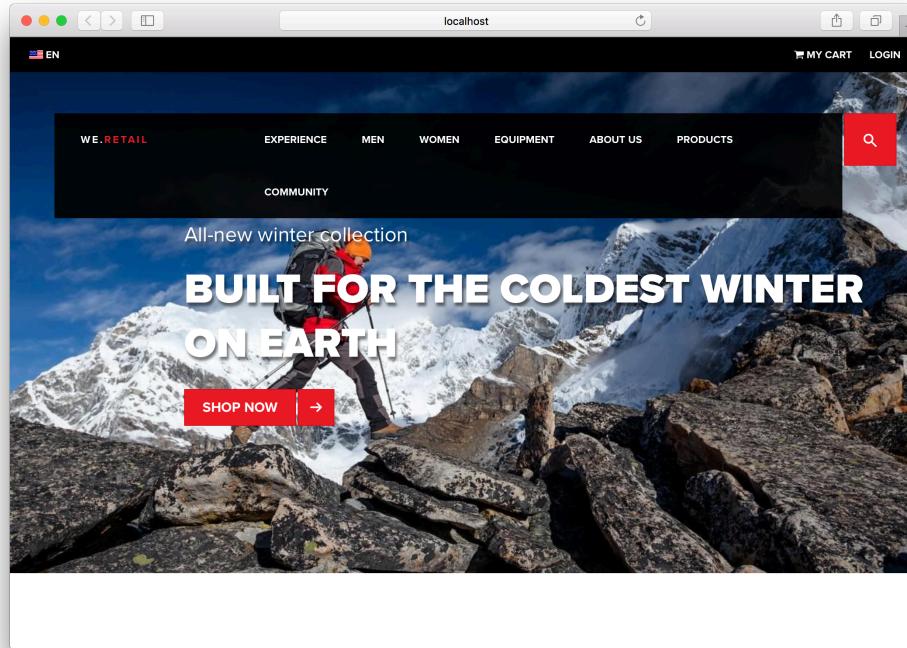
1. Find and open the **AEMM-Labs** folder on the desktop
2. Open the author folder and double-click the quickstart jar (**aem-author-4502.jar**) to start the AEM author server
 - a. When the AEM author server has started your browser window/tab will automatically open to <http://localhost:4502/> (this may take a few minutes)
 - b. Move on to Step 3 while you wait for the author server to finish starting



3. Open the publish folder and double-click the quickstart jar (**aem-publish-4503.jar**) to start the AEM publish server
 - a. When the AEM publish server has started your browser window/tab will automatically open to <http://localhost:4503/> (this may take a few minutes)



4. Wait for the author server to start (login screen will be shown)
5. Wait for the publish server to start (We.Retail Site will be shown)
 - a. If the browser asks to user your current location click 'Allow' to dismiss the dialog



Exercise 2 – Install lab content

1. Open the **L3846** folder under **AEMM-Labs** on the desktop
2. Double click on **install-start.command** and wait for the package installation to complete

Exercise 3 – Sign in to AEM

1. Navigate to the AEM Author login in a browser (<http://localhost:4502/>)
2. Enter the user name '**admin**'
3. Enter the password '**admin**'
4. Click '**Sign In**'

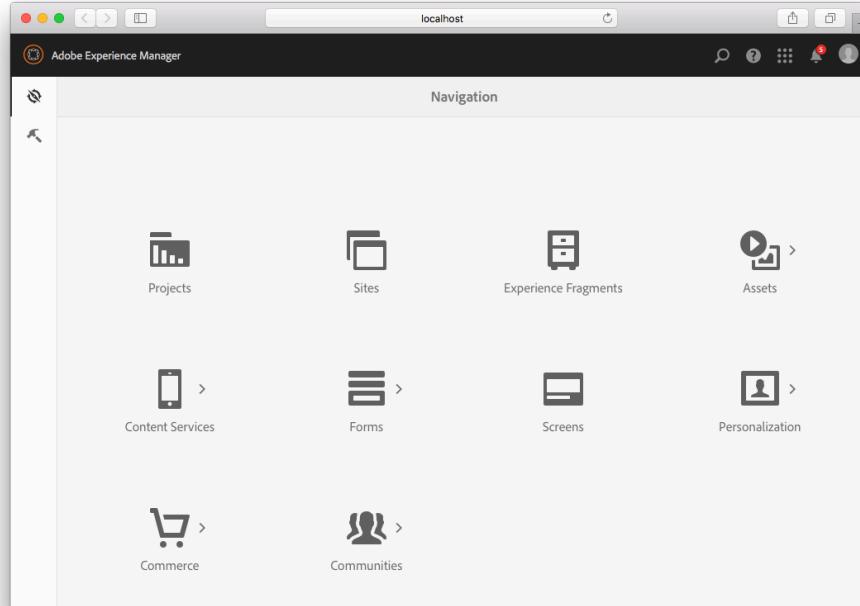


Exercise 4 – Content Services

1. The start page will appear initially after logging in to AEM (<http://localhost:4502/aem/start.html>)



- a. The start page can be accessed anytime by clicking on **Adobe Experience Manager** in the header bar.



2. Click on the **Content Services** capability



3. Open the **Apps** catalog



The app catalog will display all the mobile applications being managed by AEM and allows new mobile application entries to be created.

Summary

You are now familiar with the basics of Adobe Experience Manager and the Content Services App Catalog.



Module 2 – Mobile App Management

This module will introduce you to mobile app management and content services features in Adobe Experience Manager.

Lesson 1 – App Spaces

Overview

Spaces provide the ability to logically group similar content together that can then be easily consumed by several different mobile applications. Structured content under a space are called **Entities**. Entities can also be further grouped under language folders to assist with translation workflows.

Objectives

- Create a new space

Exercise 1 – Create a new space



1. From the AEM start page select **Content Services > Spaces**
 - a. <http://localhost:4502/libs/mobileapps/caas/admin/content/spaces.html/content/entities>
2. A couple entries will already exist in the Spaces console for **Drinks** and **Drink Ingredients** that the mixology app will be able to consume
3. Click on the **Create** button to create a new space
4. **Title:** Mixology
5. **Name:** mix
6. **Description:** Corporate news and assets
7. Click **Create**

Lesson 2 – App Catalog

Overview

AEM can manage a variety of mobile applications which addresses the need to have oversight over an organization's apps.

Objectives

- Create a new content mobile app
- Change meta data of app
- Add spaces to app

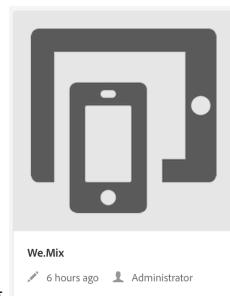
Exercise 1 – Create a new app

1. Navigate to the AEM app catalog by selecting **Content Services > Apps** from the start page
 - a. <http://localhost:4502/aem/apps.html/content/mobileapps>
2. Click the **Create** button
3. Choose **App** from the drop-down menu



Content Mobile App

4. Select **Content Mobile App**
5. Click **Next**
6. *Title: We.Mix*
7. *Name: we-mix*
8. Click **Create**
9. Click **Done**



10. A new entry will now appear in the app catalog

You have now created a new mobile app entry in AEM.

Exercise 2 – Edit app meta data

1. Navigate to the AEM Mobile dashboard by clicking on the **We.Mix** entry in the app catalog
 - a. <http://localhost:4502/libs/mobileapps/admin/content/dashboard.html/content/mobileapps/we-mix>



2. Explore the tiles on the mobile app dashboard

The dashboard for the app is created with several common tiles which can be customized to suit an organization's requirements.

- a. **Manage App:** manage meta data associated with the app
- b. **Manage Entity Content:** manage content services spaces associated with app
- c. **Manage Page Content:** manage all authored page content
- d. **Manage Content Packages:** manage the release of content updates to devices
- e. **Manage Cloud Services:** manage connections to a variety of cloud services

3. Expand the **Manage App** tile by clicking in the bottom right corner

4. Click on to edit the app icon

- a. Type **icon** in the search bar and press **Enter**



- b. Select the **WE.MIX icon.png** and click on **Select**

5. Change the app description

6. Edit the remaining meta data

- a. **ID: com.adobesummit.mixology**
- b. **Version: 1.0.0**
- c. **Platforms: ios**



7. Click **Save & Close**

Exercise 3 – Add spaces to app

Indicate which spaces a mobile application will require access to.

1. Expand the **Manage Entity Content** tile by clicking  in the bottom right corner
2. Click on **Add Space**
3. Select all available spaces from the **Add Space** dialog by selecting each item from the drop-down list
4. Each space will appear under the drop-down list once it has been selected

Add Space X

Include which groups of Entities to have access to from this app.

Space(s) * ?

▼

X Drinks X Drink Ingredients X Mixology

Cancel Add

5. Click on **Add**
6. Return to the dashboard by clicking on  and selecting **We.Mix**
7. The Manage Entity Content tile now displays the three spaces associated with the app



The screenshot shows the AEM interface with the title 'We.Mix' in the top navigation bar. On the left, there's a 'Manage App' panel for 'We.Mix' (we.mix.app), showing details like Version: 1.0, Update URL: http://localhost:4503/, and Platforms: iOS and Android. Below this is a 'DESCRIPTION' section with the text 'Elevating the experience of discovering and appreciating the art of mixology.' To the right, there are two lists: 'Manage Entity Content' and 'Manage Page Content'. The 'Manage Entity Content' list includes entries for 'Mixology' (Corporate news and assets, modified 6 hours ago by Administrator), 'Drinks' (Ready, set, mix!, modified 7 hours ago by Administrator), and 'Drink Ingredients' (Individual ingredients used in drink recipes, modified 7 hours ago by Administrator). The 'Manage Page Content' list shows one entry for 'English' with 1 file, modified 'Never', and an 'INCLUDE IN BUILD' checkbox checked.

Summary

You created a new entry for a mobile app in the AEM Mobile app catalog as well as a new space for grouping entities that can be consumed by a mobile application.

Need to catch up? Double click **install-module2.command** to install all the exercises from this module.



Module 3 – Extend Existing Content

This module will introduce you to built-in support in AEM content services to leverage existing pages and assets for mobile applications.

Lesson 1 – Pages

Overview

Fully leverage your investment in AEM by extending the use of existing news articles that have been authored in the We.Mix site.

Objectives

- Create a new entity using built-in Pages model
- Edit configuration of page entity
- Preview generated data

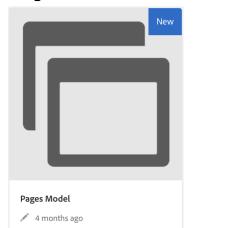
Exercise 1 – Explore Site Content (optional)

1. Navigate to the **Sites** console
 - a. <http://localhost:4502/sites.html/content>
2. Continue navigating to *We.Mix / Welcome to We.Mix / News*
3. Select a Page under News
 - a. In Column View () a page is selected by clicking on the thumbnail image
4. Click on **Edit** in the action bar
5. View the page content
6. Close the Editor browser tab when done viewing the page content

The next exercise will proceed to re-use these news articles for delivery by content services.

Exercise 2 – Create new Pages entity

1. Navigate to **Content Services > Spaces**
2. Click on the **Mixology** entry created in the previous module
3. Click the **Create** button
4. Click on **Entity** in the drop-down

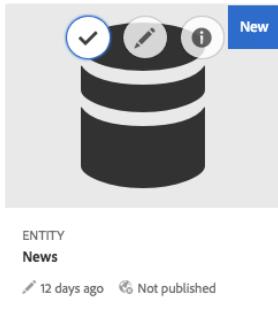


5. Select the Pages Model
6. Click **Next**
7. **Title: News**
8. **Name: news**
9. Click **Create**
10. Click **Done**

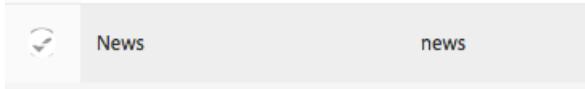


Exercise 3 – Edit Pages entity

1. Select the **News** entity created in the previous exercise from the list
 - a. Selecting a list item in the AEM touch interface card view can be achieved by hovering over the card and clicking on the checkmark icon.



- b. Selecting a list item in the AEM touch interface list view can be achieved by clicking on the thumbnail image in the left column.



2. Click on **Edit** in the action bar
3. Browse to the News page in the We.Mix site to reference its content

- a. Click on  in the path picker
 - b. Navigate to *We.Mix / Welcome to We.Mix / News*
 - c. Select **News**

- d. Click on the check mark button to pick the selected path 

4. Expand **Save & Close** button by clicking on 
5. Click on **Save** in the drop-down that appears
 - a. If you click Save & Close instead don't worry. You can just select the entity and return to editing again.
6. Click on **Preview** tab
7. Examine the resulting JSON that is generated for the page path that was previously chosen
8. Change the depth and examine the JSON again. What changed?
 - a. Hint: Scroll down and compare the contents of the child page objects (*pucker-up, three-mocktails, etc.*)
9. Click **Cancel** to return to the Content Services Spaces console

An optional module at the end of this lab will explore consuming content services data in a mobile app.

Lesson 2 – Assets

Overview

Fully leverage your investment in AEM by extending the use of existing Assets content.

Objectives

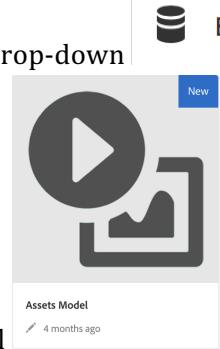
- Create a new entity using built-in Assets model



- Edit configuration of asset entity
- Preview generated data

Exercise 1 – Create new Assets entity

1. Navigate to **Content Services > Spaces**
2. Click on the **Mixology** entry created in the previous module
3. Click the **Create** button

4. Click on **Entity** in the drop-down
- 
- Entity

5. Select the Assets Model
6. Click **Next**
7. *Title:* **Featured Drinks**
8. *Name:* **featured-drinks**
9. Click **Create**
10. Click **Done**

Exercise 2 – Edit Assets entity

1. Select the **Featured Drinks** entity created in the previous exercise from the list
2. Click on **Edit** in the action bar
3. Browse for the assets this entity should reference
 - a. Click on  to open asset picker
 - b. Enter **drink-images** in search bar and press **Enter**
 - c. Select several images
 - d. Click on **Select** pick the selected assets
4. Expand **Save & Close** button by clicking on 
5. Click on **Save** in the drop-down that appears
 - a. If you click Save & Close instead don't worry. You can just select the entity and return to editing again.
6. Click on **Preview** tab
7. Examine the resulting JSON that is generated for the list of assets that were previously chosen



8. Click on that is beside the entity's URL
9. Select **Open on Author**
10. Examine the generated JSON in the new browser tab/window that is opened
11. What happens when navigating to an **href** for one of the assets?
12. Close the browser tab/window when done
13. Click **Cancel** to return to the Content Services Spaces console

A JSON viewer extension has been added to the Chrome browser for this lab. This extension makes it easier to preview the data generated by AEM content services.

Summary

The built-in page and asset models allow existing content from Sites and Assets to be quickly added to content service entities for delivery to a mobile application.

Need to catch up? Double click **install-module3.command** to install all the exercises from this module.



Module 4 – Managing Models

Data modeling enables complex data structures to be authored with content services. A model defines the set of data types which will be associated with an entity. A data model can also describe relationships between models for grouping and data integrity. Several models have already been installed for this lab but two additional models for drink recipes still need to be created.

Lesson 1 – Group Models

Overview

An important feature of data modeling is the ability to enforce data integrity. Group models allow a certain structure to be enforced during entity creation. Group models cannot define any custom properties.

This lesson will create a new group model for drink recipes. The model will enforce that only drink recipe models can be created as children.

Objectives

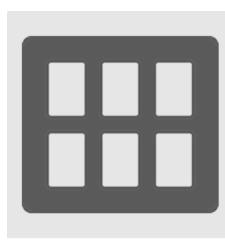
- Create a new group model
- Edit the group properties

Exercise 1 – Create a new group model

1. Open the AEM start page (hint:  Adobe Experience Manager)
2. Click on the **Tools**  button
3. Click on **General** (selected by default)

The screenshot shows the AEM General Tools interface. In the top navigation bar, there is a 'Tools' button with a gear icon and the text 'General'. Below the navigation, there is a sidebar with a 'Models' section. The 'Models' section has a sub-section titled 'Manage Content Services data model templates'. There is a small icon of a document with a grid pattern next to the section title.

4. Click on **Models**
 - a. The browser will navigate to:
<http://localhost:4502/libs/mobileapps/caas/admin/content/models.html/conf>
5. Click on **We.Mix** folder
6. Click on **Create**



7. Select **Entity Group**
8. Click on **Next**
9. **Model Title: Drink Recipe Group**

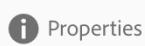


10. Click on **Create**

11. Click on **Done**

Exercise 2 – Edit model properties

1. Select the **Drink Recipe Group** created in the previous exercise



2. Click on **Properties** button in the action bar
3. Add a new item under **Allowed Children**
 - a. /conf/we-mix/settings/mobileapps/models/drink-recipe
4. Click on **Save & Close**

You have now created an entity group model for the drink recipe model that will be created in the next lesson.

Lesson 2 – Primitive Models

Overview

The primitive model type can be used to create a model that defines properties using the built-in set of data types. Additional model types can be created that limit the allowed data types or introduce more complex types. The primitive model type includes data types for primitive values and includes types for asset lists and content references.

This lesson will create a new model for a drink recipe. A drink recipe can only be added to a recipe group and will allow ingredients to be defined as children. The model will also define properties for whether the drink is alcoholic, a rating and a list of instructions.

Objectives

- Create a new model based on primitive model type
- Define availability of model
- Add data types to model

Exercise 1 – Create a new model

1. Return to the We.Mix Models console described in Lesson 1

a. <http://localhost:4502/libs/mobileapps/caas/admin/content/models.html/conf/we-mix>

2. Click on **Create**



TEMPLATE FOR PRIMITIVE DATA TYPES
Primitive Types Model

3. Select **Primitive Types Model**

4. Click on **Next**

5. *Model Title: Drink Recipe*

6. *Description: A recipe for a cocktail.*



7. Click on **Create**
8. Click on **Done**

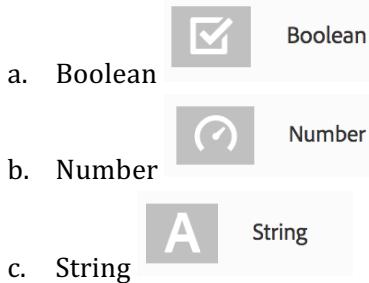
Exercise 2 – Edit model properties

1. Select the **Drink Recipe** model created in the previous exercise
2. Click on **Properties** button in the action bar
3. Click on **Upload Image** button
4. Choose **drink-recipe-thumb.png** from `~/Desktop/AEMM-Labs/L3846/images` folder
5. Edit the Model properties
 - a. *Model Type:* **DrinkRecipe**
6. Add a new item under **Allowed Children**
 - a. `/conf/we-mix/settings/mobileapps/models/ingredient-list-group`
7. Add a new item under **Allowed Parents**
 - a. `/conf/we-mix/settings/mobileapps/models/drink-recipe-group`
8. Click on **Save & Close**

You have now created a drink recipe model that enforces creation only under a recipe group and allows children ingredient list group models.

Exercise 3 – Edit model definition

1. Select the **Drink Recipe** model created in the previous exercise
2. Click on **Edit** button in the action bar
3. Drag the following data types anywhere onto the data model preview canvas



4. Click on **Boolean** field in canvas to edit its properties
5. Edit the properties for the **Boolean** field
 - a. *Field Text:* **Alcoholic**
 - b. *Property Name:* **alcoholic**
 - c. *Default:* **checked**
 - d. *Description:* **Does this recipe contain alcohol?**
6. Edit the properties for the **Number** field
 - a. *Field Label:* **Rating**
 - b. *Property Name:* **rating**
 - c. *Type:* **Long**
 - d. *Default Value:* **3**
 - e. *Min:* **1**
 - f. *Max:* **5**
 - g. *Step:* **1**



- h. *Required: checked*
 - i. *Description: The rating between 1 and 5*
7. Edit the properties for the **String** field
- a. *Multi-value: checked*
 - b. *Field Label: Instructions*
 - c. *Property Name: instructions*
 - d. *Max Length: 500*
 - e. *Required: checked*
 - f. *Description: The instructions for making the cocktail*
8. Click on **Save**
9. Click on **Cancel** (make sure you Save first!)

The screenshot shows the Data Model Editor interface on a Mac OS X system. The title bar says "localhost". The main window is titled "Drink Recipe". On the left, there are three fields: "Alcoholic" (Boolean type, checked), "Rating*" (Number type, value 3), and "Instructions*" (String type). On the right, there is a sidebar titled "Data Types" with a list of eight types: String, Number, Boolean, Date, Content Reference, Asset List, Section, and Hidden. The "String" type is currently selected. At the top right of the editor window are "Cancel" and "Save" buttons.

The drink recipe model is now ready to be used for entity creation.

Summary

Data modeling in content services allows complex data structures to be defined that will enable content authors to easily create entities while ensuring data integrity is enforced.

Need to catch up? Double click **install-module4.command** to install all the exercises from this module.



Module 5 – Managing Entities

This module will continue exploring entity management in AEM that was first introduced during the extending existing content module. Instead of using the built-in models, entities can also be created from custom models like those created in the previous module.

Lesson 1 – Entity Creation

Overview

Use the models created in the previous module to create a new drink recipe entity that includes references to other entities. Preview the generated data of the completed entity.

Objectives

- Create a new entity based on drink recipe model
- Edit properties of an entity
- Add references to other entities
- Preview completed entity

Exercise 1 – Create new entity

1. Navigate to **Content Services > Spaces**
2. Click on the **Drinks** space
3. A **Drink Recipes** entity group has already been installed that is based on the Drink Recipe Group model that was created in the previous module
4. Click on the **Drink Recipes** entity
5. Several drink recipes have already been pre-installed. Let's add one more!
6. Click on **Create** and select **Entity** from the drop-down
7. Select **Drink Recipe** model
 - a. Notice that this is the only model available based on the rules defined during model creation
8. Click on **Next** to edit the entity properties
 - a. **Title: Martini**
 - b. **Name: martini**
 - c. **Description: The quintessential drink of spies everywhere.**
9. Click on **Create**
10. Click on **Done**

Exercise 2 – Edit entity image

1. Select the **Martini** entity created in the previous exercise
2. Click on **Properties** button in the action bar
3. Click on the image edit button
4. Type **martini** in the search bar and press **Enter**
5. Select **martini.jpg** and click on **Select**
6. Click on **Save & Close**

Exercise 3 – Edit entity data

1. Select the **Martini** entity created in the previous exercise



2. Click on **Edit** button in the action bar
3. The displayed form is obtained from the associated model that was created in the previous module
4. Enter data values
 - a. *Alcoholic: checked*
 - b. *Rating: 5*
 - c. *Instructions (click Add button for each instruction)*
 - i. **Fill shaker with ice**
 - ii. **Combine ingredients**
 - iii. **Stir well to chill**
 - iv. **Strain into a chilled cocktail glass**
 - v. **Garnish with a twist of lemon peel or an olive as desired**
5. Click on **Save & Close**

Exercise 4 – Entity references

Now some ingredients need to be associated with this drink recipe. A drink ingredients space already exists so the recipe only needs to reference the desired ingredients instead of having to create new ingredients. The ability to reference existing content is a powerful feature of content services.

1. Click on the **Martini** entity list item to begin adding child entities.
2. Click on **Create > Entity** and select **Ingredient List Group**
3. Click **Next** to edit the properties
 - a. *Title: ingredients*
4. Click on **Create and Done**
5. Click on the **ingredients** entity that was just created to add child entities
6. Click on **Create > Entity** and select **Ingredient List Item**
7. Click **Next** to edit the properties
 - a. *Title: Vodka*
 - b. *Name: vodka*
8. Click on **Create and Done**
9. Select and Edit the **vodka** entity to reference an item from the drink ingredients space
 - a. *Ingredient: /content/entities/ingredients/vodka*
 - b. *Amount: 1.5 ounces*
10. Click on **Save & Close**
11. Create additional ingredient entries
 - a. Dry Vermouth
 - i. *Title: Dry Vermouth*
 - ii. *Name: dry-vermouth*
 - iii. *Ingredient: /content/entities/ingredients/vermouth*
 - iv. *Amount: 0.25 ounces*
 - b. Olives
 - i. *Title: Olives*
 - ii. *Name: olives*
 - iii. *Ingredient: /content/entities/ingredients/olives*
 - iv. *Amount: 2 olives*
12. Click on **Save & Close**



Exercise 5 – Entity preview

Previewing generated data during the authoring of structured entity content can be used to quickly ensure the output conforms to a desired schema and to ensure there are no errors before performing testing on a device.

1. Return to the **Drink Recipes** list
2. Select and Edit the **Martini** entity
3. Click the **Preview** tab
4. Change the Depth to **2**
5. Open the preview options by clicking on the drop-down chevron beside the preview URL
Click on **Open on Author** to view JSON in a new tab

The screenshot shows the AEM authoring interface for the 'Martini' entity. The 'Properties' dialog is open, and the 'Preview' tab is selected. The 'Selector' field contains 'caas'. The 'Depth' dropdown is set to 2. The JSON preview area shows the following structure:

```
{
  "title": "Martini",
  "lastModified": "Tue Mar 07 2017 13:25:49 GMT-0500",
  "description": "The quintessential drink of spies everywhere.",
  "image": {
    "href": "http://localhost:4502/content/entities/drinks/recipes/martini/caas.2.json",
    "path": "/content/entities/drinks/recipes/martini/jcr:content/image",
    "type": "foundation/components/image"
  },
  "ingredients": [
    {
      "name": "vodka",
      "title": "Vodka",
      "amount": "1.5 ounces",
      "lastModified": "Tue Mar 07 2017 13:31:40 GMT-0500",
      "ingredient": {
        "name": "vodka",
        "title": "Vodka",
        "path": "/content/entities/ingredients/vodka",
        "type": "Ingredient",
        "href": "http://localhost:4502/content/entities/ingredients/vodka.caas.json"
      }
    }
  ]
}
```

A context menu is open over the JSON preview, showing options: 'Copy to Clipboard', 'Open on Author' (which is highlighted), and 'Open on Publish'.

Lesson 2 – Entity Publishing

Overview

Entities need to be published just like pages on a site. The publishing of entities is the final step before exposing authored content to a mobile app.

Objectives

- Select single entity to publish
- Publish all entities under a space



Exercise 1 – Publish entity

1. Select and Edit the **Martini** entity
2. Click on the **Preview** tab
3. Choose a depth of **2**
4. Select **Open on Publish** from the preview options
 - a. <http://localhost:4503/content/entities/drinks/recipes/martini.caas.2.json>
5. Since this is a new entity it will not be available yet on the publish instance and a 404 error will be returned
 - b. Leave this tab open for the remainder of the lesson
6. Click on **Cancel** in the entity editor
7. Select the **Martini** entity from the drink recipes list



8. Click on **Quick Publish**
9. Ensure all the assets are selected and click on **Publish**
10. Return to the publish preview URL tab and refresh
11. The entity data should now be available on publish. Quick publish however did not include all the child ingredients.
12. Return to the drinks list and re-select the **Martini** entity



13. Click on **Publish Tree**
14. Return to the publish preview URL tab and refresh
15. The list of ingredients is now also available. There are still errors however since the ingredient content references have not been published.

Exercise 2 – Publish space

Spaces can be published by following the same steps used for an entity.

1. Navigate to the **Content Services > Spaces** console
 - a. <http://localhost:4502/libs/mobileapps/caas/admin/content/spaces.html/content/entities>
2. Select one of the three spaces
3. Click on **Publish Tree**
4. Repeat for the remaining two spaces
5. Return to the publish preview URL tab from the previous exercise and refresh
6. The ingredient error has been replaced by an **href** to the ingredient data

Summary

Content service entities can also be created from custom models which allows complex data structures to be authored. When entities are ready to be made available to a mobile application they need to be published like any other content stored in AEM.

Need to catch up? Double click **install-module5.command** to install all the exercises from this module.



Module 6 – Mobile Application (optional)

This module will view all the content created during this lab in a mobile application.

Lesson 1 – Run Mobile Application

Overview

A sample iOS project has been included with the lab that can be open and run in XCode. The sample mixology app has been configured to communicate with a local AEM instance on port 4503.

Objectives

- Publish all initial content required by mobile app
- Launch XCode and run mobile app
- Explore use of content services data in app
- Edit entity and publish
- View changes in mobile app

Exercise 1 – Install additional content

Additional content needs to be installed into the We.Mix app entry that was created during this lab.

1. Open the **L3846** folder under **AEMM-Labs** on the desktop
2. Double click on **install-module6.command** to install the additional content to the AEM author instance

Exercise 2 – Publish content

All the entity content required by the app has already been published by the previous module. One additional content package needs to be installed that allows the news content to be made available offline.

1. Navigate to the Content Services app catalog
 - a. <http://localhost:4502/aem/apps.html/content/mobileapps>
2. Click on the **We.Mix** app to open its dashboard
3. Click on **English** under the Manage Content Packages tile
4. Click on **Stage** in the action bar
5. Provide a title and click on **Done**
6. Click on **Publish** in the action bar
7. Provide a title and click on **Done**

Exercise 3 – Run app

1. Open the **mixology-app** folder under **~/Desktop/AEMM-Labs/L3846**
2. Double click on **Mixology.xcodeproj**
3. Select a desired simulator from menu
4. Under the Product menu select **Run**
5. Switch to the Simulator app in the dock once it opens



6. Explore the app
 - a. Tap on Drinks tab
 - b. Scroll down to find Martini
 - c. Tap on **Martini** list item
 - d. Confirm data is same as what was authored in the previous module

Exercise 4 – Publish changes

1. Edit the **Martini** entity in AEM
 - a. <http://localhost:4502/libs/mobileapps/caas/admin/content/spaces/editor.html/content/entities/drinks/recipes/martini>
2. Change the rating and instructions
3. Click on **Save & Close**
4. **Quick Publish** the **Martini** entity
5. Return to drinks list in the app
6. Pull down to refresh
7. Confirm rating has changed in list
8. Tap on **Martini** list item
9. Confirm instructions have changed



Summary

Congratulations! You have launched a mobile application that consumes published content services data from AEM.

Martini

Ingredients

Vodka	1.5 ounces
Dry Vermouth	.25 ounces
Olives	2 olives

