WSHFC Multifamily Developer Portal System Documentation

Contents

Security

Profiles

Setup -> Users -> Profiles

In order to control security for the new External Users of the Multifamily Development portal, new profiles were created:

- Multifamily Community User
- Multifamily Community User Login
- The difference between the two is which type of licensing used (named licensing vs per login licensing). Other than that the two profiles are identical.

For internal users, all security is controlled using existing profiles. The primary focus and testing was performed using these profiles:

- MFHCF
- System Administrator

Permission Sets

Allows a multifamily user to be a community admin and manage other users. Grants the Community Admin permission. The custom permission is called "Multifamily Community Admin". See the Community Admin section.

Sharing Settings

Setup -> Security -> Sharing Settings

Security for external users had not been configured prior to the start of this project. The "Default External Access" for many objects was configured to keep data secure and private. The below table illustrates the security granted by object:

Object	Permissions Needed	Org Wide Sharing	Granted by
Accounts	Read/Edit – related to Funding Contact Roles	Private (Original Sharing Settings were Public Read/Write)	Flow/Ownership
Allocation Round	Read	Public Read Only	Sharing Settings – Public Read Only
AMI Set Up Data	Read	Controlled by Parent	Controlled by Parent Application Set Up Data Record
Application	Read/Edit	Private	Flow/Ownership
Application Set Up Data	Read	Public Read Only	Sharing Settings – Public Read Only
Asset	Read (read/edit if created by application)	Controlled by Parent	Permission to Site->Project
Asset Funding	Read/Edit	Controlled by Parent	Permission to Asset->Site->Project Permission to Funding->Project

Bond Issue	Read	Private (Original Sharing Settings were Public Read/Write)	Flow/Ownership
Bond Issue Funding	Read/Edit	Controlled by Parent	Permission to Funding->Project Permission to Bond Issue
Bonds	Read	Controlled by Parent	Permission to Bond Issue
Checklist	Read	Public Read Only	Sharing Settings – Public Read Only
Checklist Item	Read	Controlled by Parent	Permission to Checklist
Checklist Response	Read/Edit	Controlled by Parent	Permission to Application
Contacts	Read/Edit – related to Funding Contact Roles	Private (Original Sharing Settings were Public Read/Write)	Flow/Ownership
Funding	Read/Edit	Controlled by Parent	Permission to Project
Funding Contact Roles	Read/Edit	Controlled by Parent	Permission to Funding->Project
LIH Set Asides	Read/Edit	Controlled by Parent	Permission to Funding->Project
Project	Read/Edit	Private (Original Sharing Settings were Public Read/Write)	Flow/Ownership
Project Update	Read/Edit	Controlled by Parent	Permission to Project
Site	Read (read/edit if created by application)	Controlled by Parent	Permission to Project
Special Populations	Read/Edit	Controlled by Parent	Permission to Funding->Project

Sharing via flows

A sharing setting of Private indicates that only the owner of that record and users that have explicitly received sharing to the record can see that data. In the Multifamily Developer Portal, the person who initiates the application becomes the owner of those records and as a result has access to them.

One of the requirements of the project is that multiple users be able to have access to the same application. As a result it is necessary that explicit sharing be granted to these additional users. This happens using the following method:

1. A new user is invited to the community to participate on a particular application. This happens through the "Invite others to collaborate on this application" button on the Application page in the Multifamily Developer Portal.
2. When that invitation happens, a record is created in the **Community Invitations** object.
3. The party that receives the invitation is then able to log in to the community through the invitation. This may be an existing or new user.
4. Logging in using an invitation updates the Community Invitation record and sets the **Invitation Used By** field to the user who used the invitation and sets the **Code Used** to true thus invalidating it for future use. The **Source Record Id** field on the Community Invitation is the ID of the application record to which the user should have access.
5. As soon as the Code Used and Invitation Used By fields are set, the **Community Invitation – After Save** flow is triggered. This runs an **Asynchronous** record create to create a record in the Application Permission table which indicates the Application and the User. Asynchronous means that it is run in a separate execution context from whatever else triggered the process to run. This is done because assigning permissions can be a CPU intensive process and running asynchronously helps to avoid hitting Salesforce system limits.
6. Creating a record in the **Application Permission** table triggers the **Application Permission – After Save** flow to run. This flow does the following:
    a. Grants permission to the Application
    b. Grants permission to the Project related to the application.
        i. Done through a Subflow called **Project – Add Sharing**
    c. Grants permission to any Funding Contact Roles related to the application.

Note: Internal users may grant permission to an application to an External User by manually adding a record to the Application Permission table. Simply entering the User and the Application is all that is required. This will cause step 6 to run above and grant the necessary permissions.

These are the scenarios where permissions need to be granted:

1. During initial application creation
    a. May need to grant access to projects if application is associated to existing project(s).
2. During registration with invitation code OR manually if admin says reset sharing
    a. Grant permission to application
    b. Grant permission to project(s)
    c. Grant permission to Accounts
    d. Grant permission to Contacts
    e. Grant permission to Bond Issues
3. During creation of a new Contact Funding Role
    a. Grant permission to Account for related users
    b. Grant permission to Contact for related users
4. During creation of a Bond Issue Funding
    a. Grant permission to Bond Issue for related users

In order to achieve the above the following flows are in place:

1. Application – After Save
    a. When an Application is created or the Owner is changed, a new record is created in Application Permission object for External Users only.
2. Application Permission – After Save
    a. When an Application Permission record is created, grants access to the Application and all Related records
3. Project - Add Sharing
    a. Called as a subflow by Application Permission – After Save
4. Funding Contact Role – After Save
    a. After a funding contact role is created or when the contact is changed, this grants permission to external users who are related to the application related to the funding for the funding contact role.
5. Bond Issue Funding – After Save
    a. When a new bond issue funding record is created, grants permission to the bond issue to all external users related to the application that is related to the funding that the bond issue is for.
6. Community Invitation – After Save
    a. When a new record is updated in the community invitation object such that it has the user and application filled in this triggers the creation of a new Application Permission record which in turn triggers the Application Permission – After Save flow.

Invitation Only Allocation Rounds

The following metadata was created to support invitation only allocation rounds and the implementation is quite simple and straightforward:

- Added picklist value to Accepting Applications for "Invitation Only".
- Added Allocation_Round_Invitation__c custom object + fields.
- Modified New Application Flow
- Modified EC Allocation Round - Display Accepting Applications Flow

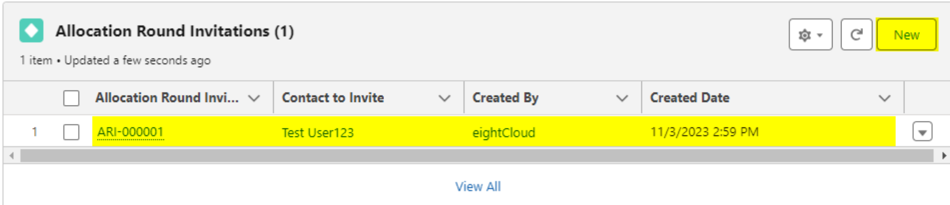
Both flows were similarly modified as follows:

1. Added a Get Allocation Round Invitation records element.
2. Loop through invitations to create a collection of Allocation Round Ids
3. Update the get of Allocation Round records to be where either
    a. Accepting applications = Yes
    b. OR By Invitation Only AND Id In [collection of invites]

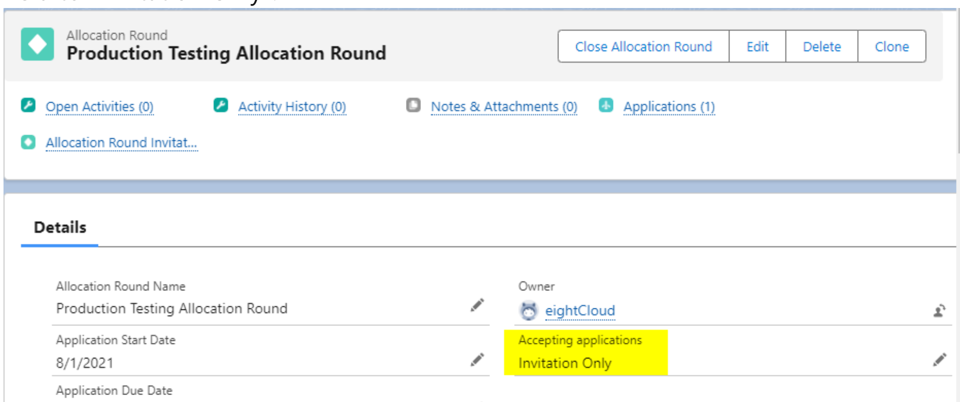
Steps to Set up an Allocation Round as Invitation Only:

1. Set up an allocation round as normally would happen.

2. Then create Allocation Round Invitation records associated to the allocation round (via the related list) for any contacts that should be able to access the allocation round.



3. Finally, when ready to make the allocation round available to invitees, set the Accepting Applications field to "Invitation Only".



Steps to Test as Contact:

1. Find a contact who has access to the Multifamily developer portal.
2. Log into the Community.
3. On the home page, verify that only the allocation rounds that are accepting applications = Yes are showing plus any that the contact has been invited to and the allocation round has a status of "Invitation Only"
4. As staff, add/remove invitation records and the re-login as the contact to verify the home page updates for the contact as appropriate.
5. On the New Application tab complete the same test and verify the correct Allocation Rounds show up.

Duplicate Rules

Setup -> Data -> Duplicate Management -> Duplicate Rules

Duplicate rules were interfering with the ability for a user with the Multifamily Community User profile to complete an application. As these users do not have access to the internal Salesforce interface, nor do they have access to all records in Salesforce, it is not feasible for these users to be asked to check for or manage duplicates. As a result the duplicate processes were modified for External users:

1. **Account Matches** duplicate rule was modified to exclude records created through Self Registration where the Profile was equal to Multifamily Community User.
2. **Contact Matches** duplicate rule was modified to exclude records created through Self Registration where the Profile was equal to Multifamily Community User.
3. **Asset Dupes** duplicate rule was modified to exclude Reporting or Alerting on anything done by the Multifamily Community User. However a second duplicate rule for Multifamily Community Users was then created.

4. **Asset Dupes Multifamily External** duplicate rule Reports on duplicates created by Multifamily Community Users.

Checklist Functionality

In order to facilitate capturing all of the items necessary for an application, a checklist functionality was developed. This functionality is composed of these major parts:

- Objects
    - Checklist
    - Checklist Item
    - Checklist Response
    - Application
- Lightning Components
    - EC_ChecklistMainCmp
    - ec_checklistTable
    - ec_checkListDocList
    - ec_checklistFileUploadDisplay

In order to create a new checklist the following steps should be taken:

1. Set up the new checklist parent.
    a. Go to Checklists. Create New.
    b. Give the Checklist a Name.
    c. Click Save.
2. Click New under Allocation Round
    a. Fill out the New Allocation Round form.
    b. Make sure it is associated to the new Checklist just created.
    c. Associate to the Allocation record as necessary.
    d. Set Accepting Applications to "No"
    e. Set Status to "Draft"
    f. Set Display on Community = FALSE
    g. Click Save.
3. Create checklist items.
    a. From the Checklist record click "New".
    b. Select the type of item you wish to create.
    c. Flow
        i. Here's an example :
            1. Checklist Item Name: This is the name that the applicant will see. Give it a descriptive name (limit 80 characters).
                a. Example: "Project Information"

2. Description: This is a longer text description of the checklist item. The applicant will also see this. Here you can explain what the applicant must do.
    a. Example: "Please provide information about the number of units, square feet, set asides, and other items."
3. Category: Choose the category that this item will fall into. (These categories can be further refined as necessary).
    a. Example: General
4. Display Order: This is the order that the checklist item will display to the end user. If you have 5 checklist items and you put this as 1, it will display first. If you number multiple items with the same number, the system will arbitrarily choose the order within the items numbered the same.
    a. Example: 1
5. Custom Flow API Name: This is the API name of a flow that an administrator or developer has built to capture and update information in the system. A prerequisite to using this checklist item type is that a flow is built ready to go. For our example we have pre-created a flow:
    a. Example: EC_Project_Information
6. Custom Flow Variable:
    a. Example: recordId (should default to this)
7. Click Save.

d. Laserfiche Upload checklist item
    i. Give the Checklist Item Name a value. This is the value the user will see when they are working through the application.
    ii. If desired provide a Description.
    iii. Give the Checklist item a Category.
    iv. Optionally enter a Display Order to control which order the item shows up in the list of items.
    v. The Laserfiche Base URL should already be filled out with this value and this likely does not need to change: https://portal.wshfc.org/Forms/
    vi. Laserfiche Form Title will need a value which will the portion of the URL that specifies the form. In our example: Title_Report
    vii. The Custom Flow API Name should already be filled out with this value and this likely does not need to change: EC_Laserfiche_Upload
    viii. The Custom Flow Variable should already be filled out with this value and this likely does not need to change: recordId

e. Component checklist item:
    i. Click New Checklist Item again.
    ii. Select Component. Component is only for custom development screens so this option will only be chosen if a custom component has been built.
    iii. Give the Checklist Item a name. This is what the end user will see.
    iv. Optionally give the checklist item a description.

v. Optionally give the checklist item a Display Order.
vi. Give the item a Custom Component Name.
1. This value should be provided by the developer of the component.
2. In our example we will use: ec_sitesBuildings
vii. Click Save.
f. Upload
i. This is for Salesforce file upload. This is not Laserfiche. This should mostly not be used as all documents should be stored in Laserfiche.

4. When ready to "Go Live" with an allocation round:
   a. Navigate back to the allocation round record.
   b. Click Edit.
      i. Set Accepting Applications to "Yes"
      ii. Set Status to "Open"
      iii. Set Display on Community = TRUE
   c. Click Save.
5. When ready to remove an allocation round (i.e. no longer accepting applications):
   a. Navigate back to the allocation round record.
   b. Click Edit.
      i. Set Accepting Applications to "No"
      ii. Set Status to "Closed"
      iii. Set Display on Community = FALSE
   c. Click Save.

Components

The following custom components have been developed to streamline the data entry of information in the application process. These components can be referenced as checklist items with type of Component:

- ec_sitesBuildings
  - This component allows capturing information about Sites and Assets.
- ec_unitSquare
  - This component allows capturing units, square footage special population and set aside information by project, site and asset (building).
- ec_sitesBuildingsAndUnitSqFt
  - This component was developed to combine the ec_sitesBuildings and ec_unitSquare components into a cohesive two screen interface.

Flows for Checklist

The following flows have been developed to streamline the data entry of information in the application process. These flows can be referenced as checklist items with type of Flow:

- EC_Project_Information
  - Allows entering basic information about a project such as name and location.

- EC_Funding_Information
    - Allows entering details about a project including the amount of volume cap being requested.
- Funding_Contact_Roles
    - Allows entering information about a project team and partner contacts in the funding contact roles object.
- EC_Application_Fees
    - Allows calculating an application's fee and provides via email the wiring instructions.

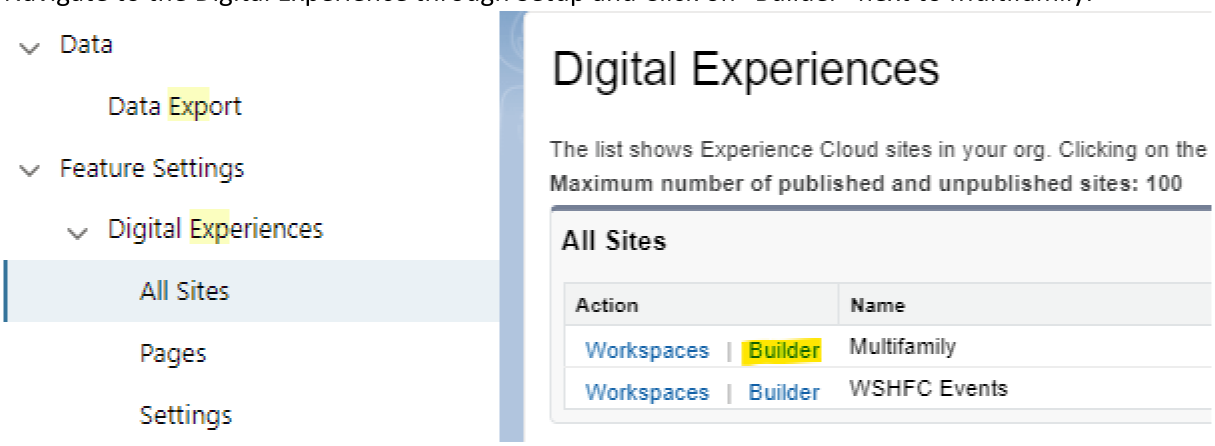
Community

Setup -> Feature Settings -> Digital Experiences -> All Sites -> Multifamily

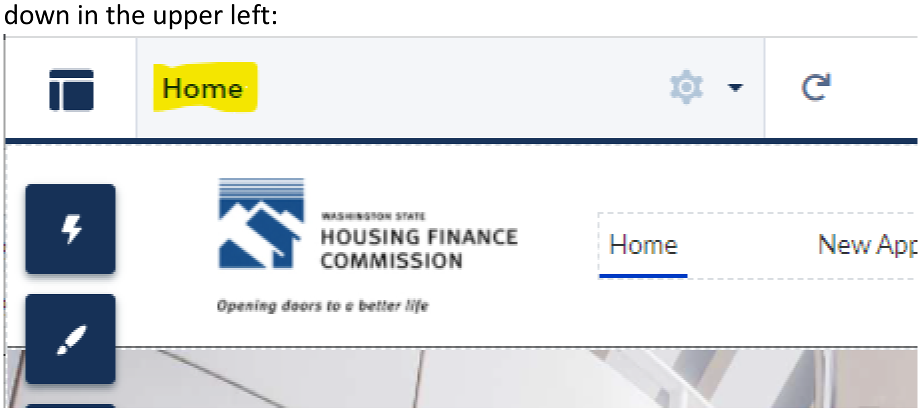
A new Salesforce Digital Experience (formerly "Community") was created to provide the portal to which applicants can log in. Below are instructions for basic upkeep of the tool.

Editing the Home Page

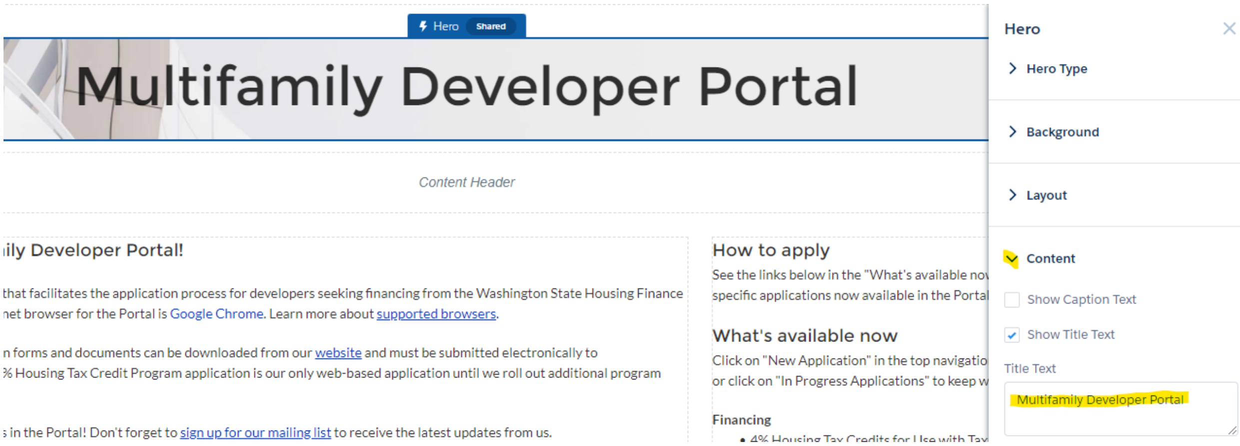
1. Navigate to the Digital Experience through Setup and Click on "Builder" next to Multifamily.



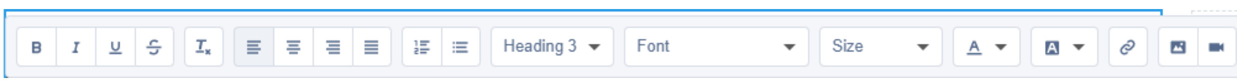
2. The tool should start you on the Home page but just in case not, select "Home" from the drop down in the upper left:



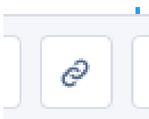
3. The Title of the Home page can be edited by clicking on the title. In the Popup that displays click the arrow next to Content. Edit the contents of the title in the Title Text box:



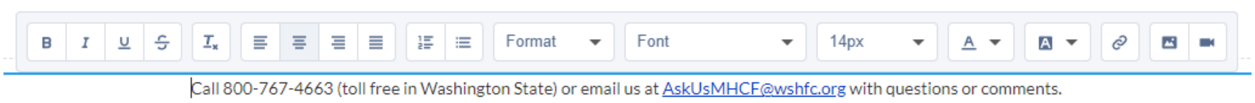
4. The text in both the left and right hand sides of the body area of the page are Rich Text components. They can be edited by clicking once on them and then clicking the Edit Content button (alternately you can double click). When the edit bar shows, you know you are in edit mode:



5. The Link Icon can be used to add a link to any piece of text:

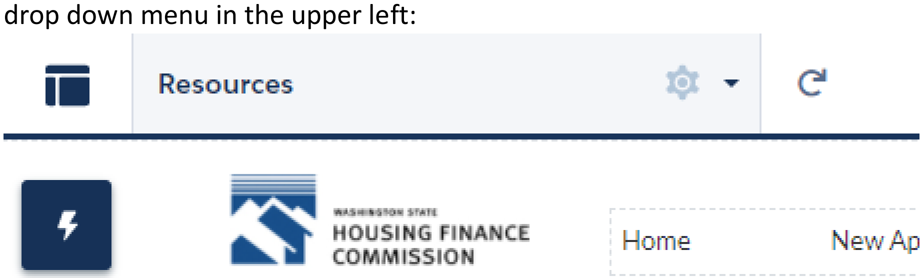


6. The footer likewise is also a rich text component and can be edited similarly. However note that the footer is a global footer which means that any edits made here apply to all pages in the Multifamily Developer Portal.



Editing the Resources Page

1. Editing the Resources page is similar to the Home page, but first you must select it from the drop down menu in the upper left:

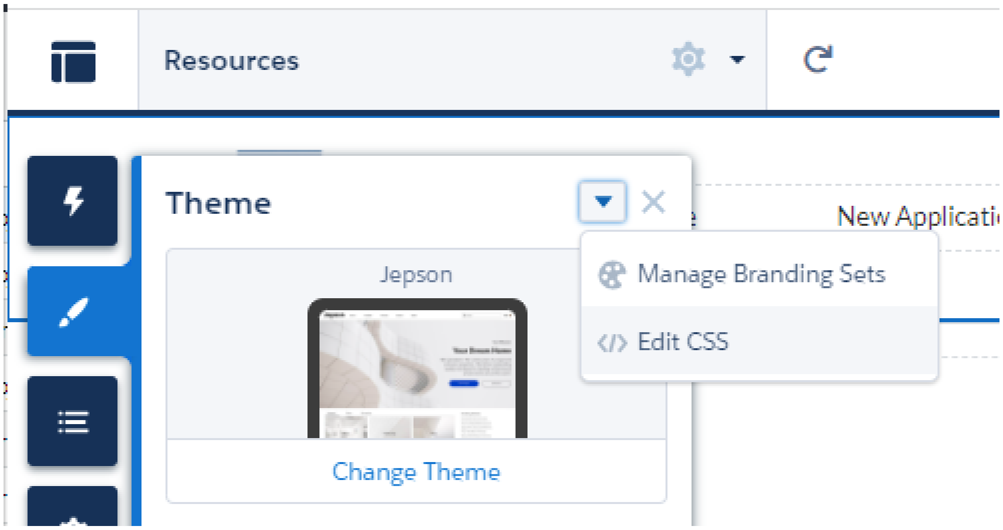


2. Then the contents of the page may be edited using the rich text editor similar to the Home page.

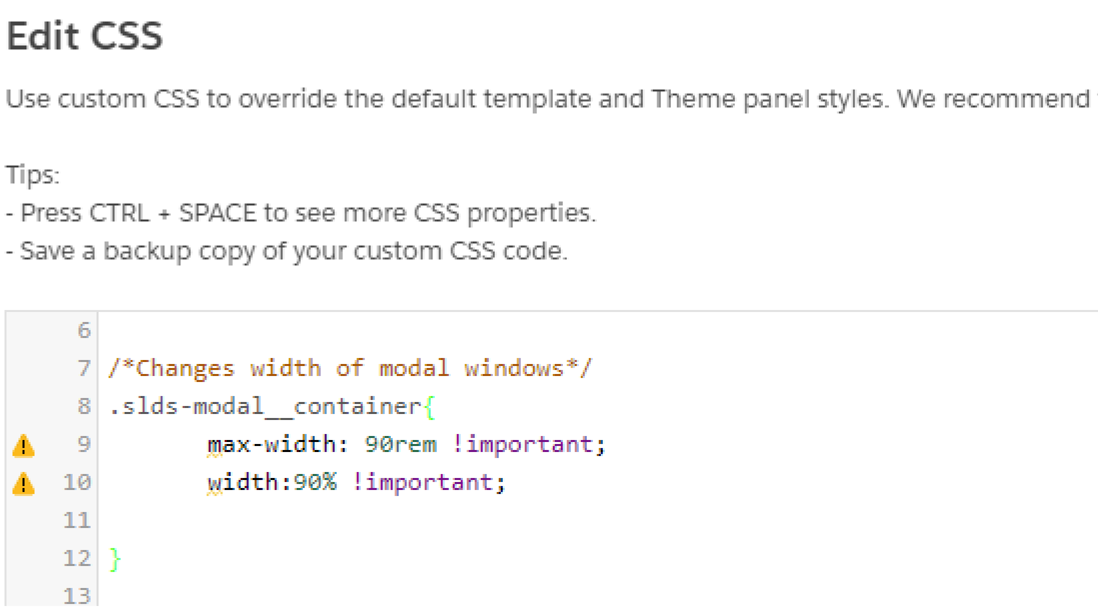
CSS

There is one minor changes that we made to the Multifamily Developer Portal through CSS (Cascading Style Sheets). These changes are overrides to the standard behavior. CSS Overrides may break if upgrading to a newer release of a digital experience so this should be tested if / when that happens.

1. To access the CSS overrides go to Theme -> arrow dropdown -> Edit CSS



2. Click the "Use Overrides" button
3. Note the overrides currently in place. Currently only the Modal container width override is being used:



4. This forces the pop up windows for flows to be wider than standard.

Self Registration

Apps -> Packaging -> Installed Packages -> selfreg (note that this will likely not appear in production as we will be deploying from Sandbox to production instead of from our accelerator library)

EightCloud provided a self registration accelerator to WSHFC. This self registration accelerator is fully open and accessible by WSHFC meaning that if code changes are necessary all of the code is installed, available and unlocked in the WSHFC org.

The package consists of these components:

Component Name	**Parent Object**	**Type**
Code Expired	Community Invitation	Custom Field
Code Used	Community Invitation	Custom Field
Date Code Sent	Community Invitation	Custom Field
Date Code Used	Community Invitation	Custom Field
Invitation Sent By	Community Invitation	Custom Field
Invitation Used By	Community Invitation	Custom Field
Invitee Email	Community Invitation	Custom Field
Single Use Code	Community Invitation	Custom Field
Source Record Id	Community Invitation	Custom Field
All	Community Invitation	List View
Community Invitation Layout	Community Invitation	Page Layout
ec_StatesList	Contact	Custom Field
ButtonPrimaryBackgroundColor	EC_CommunitySettings	Custom Field
ButtonPrimaryOutlineColor	EC_CommunitySettings	Custom Field
ButtonPrimaryTextColor	EC_CommunitySettings	Custom Field
ButtonSecondaryBackgroundColor	EC_CommunitySettings	Custom Field
ButtonSecondaryOutlineColor	EC_CommunitySettings	Custom Field
ButtonSecondaryTextColor	EC_CommunitySettings	Custom Field
CaptchaKey	EC_CommunitySettings	Custom Field
CaptchaVFURL	EC_CommunitySettings	Custom Field
CommunityName	EC_CommunitySettings	Custom Field
CreateAccountURL	EC_CommunitySettings	Custom Field
Enable Google Captcha	EC_CommunitySettings	Custom Field
FooterText	EC_CommunitySettings	Custom Field
FooterTextColor	EC_CommunitySettings	Custom Field
ForgotPasswordSuccessURL	EC_CommunitySettings	Custom Field
ForgotPasswordURL	EC_CommunitySettings	Custom Field
HelpText	EC_CommunitySettings	Custom Field
LoginURL	EC_CommunitySettings	Custom Field
PrimaryColor1	EC_CommunitySettings	Custom Field
PrimaryColor2	EC_CommunitySettings	Custom Field
Profile	EC_CommunitySettings	Custom Field
SelfRegisterOwner	EC_CommunitySettings	Custom Field
startURL	EC_CommunitySettings	Custom Field
State	EC_CommunitySettings	Custom Field
VFPageDomainURL	EC_CommunitySettings	Custom Field

EC_AccountSearchController	Apex Class
EC_CommunitySettings	Apex Class
EC_GenerateInvitationCtrl	Apex Class
EC_LightningCommRegisterControllerTest	Apex Class
EC_LightningCommunityRegisterController	Apex Class
EC_LightningLoginFormController	Apex Class
EC_AccountSearch	Aura Component Bundle
EC_AccountSearchEvent	Aura Component Bundle
EC_ForgotPassword	Aura Component Bundle
EC_GenerateInvitation	Aura Component Bundle
EC_Login	Aura Component Bundle
EC_SelfRegister	Aura Component Bundle
Community Invitation	Custom Object
EC_CommunitySettings	Custom Setting
State_Code_Global_List	Global Value Set
EC_CommunityLogo	Static Resource
SLDS246	Static Resource
EC_reCaptcha	Visualforce Page

Multiple Users for Same Application

One of the requirements of the project is that multiple users be able to have access to the same application. Special functionality was developed to allow this to happen and was included as part of the self registration accelerator that was provided.

1. A new user is invited to the community to participate on a particular application. This happens through the **Invite others to collaborate on this application** button on the Application page in the Multifamily Developer Portal.
    a. This button is a custom component. It can be added to any detail record page meaning it could potentially be used for other objects in the future.
2. When that invitation happens, a record is created in the **Community Invitations** object.
    a. This is a custom object that is part of the invitation package that was installed with Self Registration.
3. The party that receives the invitation is then able to log in to the community through the invitation. This may be an existing or new user.
4. Logging in using an invitation updates the Community Invitation record and sets the **Invitation Used By** field to the user who used the invitation and sets the **Code Used** to true thus invalidating it for future use. The **Source Record Id** field on the Community Invitation is the ID of the application record to which the user should have access.

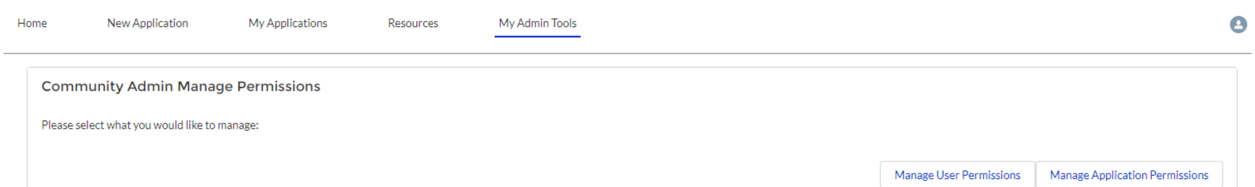
    a. This all happens through the Self-Registration/log in process and is included as part of the self registration accelerator that was installed.

The functionality below was custom developed apart from the accelerator to grant permissions to users:

5. As soon as the Code Used and Invitation Used By fields are set, the **Community Invitation – After Save** flow is triggered. This runs an **Asynchronous** record create to create a record in the Application Permission table which indicates the Application and the User. Asynchronous means that it is run in a separate execution context from whatever else triggered the process to run. This is done because assigning permissions can be a CPU intensive process and running asynchronously helps to avoid hitting Salesforce system limits.
6. Creating a record in the **Application Permission** table triggers the **Application Permission – After Save** flow to run. This flow does the following:
    a. Grants permission to the Application
    b. Grants permission to the Project related to the application.
        i. Done through a Subflow called **Project – Add Sharing**
    c. Grants permission to any Funding Contact Roles related to the application.

Community Admin

The community admin object is used to create a system where an individual can manage an account. By creating a record with a community user and an account in this object, the user is given the ability to see the My Admin Tools tab in the Digital Experience:



The solution allows the user to see other users related to the account and all of the applications related to the account and determine which user should have access to which applications. They may do this from a user perspective or from an application perspective.

Flows:

- Community_Admin_Manage_Permissions
    - This flow is the user interface that the individual interacts with to manage the permissions.
    - The outcome of this flow, is the creation or removal of Application Permission records.
- Community_Admin_After_Save
    - After a record is created in the Community Admin object, all applications related to the identified account are selected and that user is granted permission to all applications related to that account. Calls the Community Admin Change Event platform event if necessary.

- Community_Admin_Before_Delete
    - Calls the Community Admin Change Event platform event if necessary.
- Community_Admin_Change_Event
    - Adds (or removes) the Multifamily Community Admin permission set.

New Object:

- Community_Admin__c
- Community_Admin_Change__e (Platform Event)

New Custom Permission:

- Multifamily Community Admin
    - Used to grant permission to the My Admin Tools page in the community.

Application Set Up Data

Overview

There are two setup objects that are used to make the processes in the Multifamily Developer portal more configurable:

- Application Set Up Data
- AMI Set Up Data

Application Set Up Data is the parent object and AMI Set Up Data is a child object. Some Application Set Up Data records also have parent Application Set Up Data records.

The data stored in these two objects drive the options available to applicants in the following areas:

- Federal Set Asides
- State Set Asides
- County Specific logic
- Special Populations
- Geographic Credit Pools

Modifying the records in these objects change the options available in the application.

When using the **Sites, Buildings & Units** checklist item, there are several sets of Application Set Up Data that help drive the data in the component:

Almost all of the data is used on the second screen.

Additionally the Geographic Credit Pools data is used in the **Project Summary** Checklist Item.

Checklist

Category	Application Checklist Item	Description	Status	Attachments	Work on this section
General	Project Summary	Enter project summary information	Complete		Work on this section
General	Project Development	Enter bond and other development information	Complete		Work on this section
General	Sites, Buildings & Units	Enter site, building and unit information	Complete		Work on this section

Federal Set Asides

When filling out the section of the screen related to Minimum Low-Income Housing Commitments, the applicant is asked to provide information about the LIHTC Federal Minimum Set-Aside (4% and 9%) and the Tax-Exempt Bonds Minimum Federal Set-Aside (4% only).

Minimum Low-Income Housing Commitments

Instructions

To qualify for tax-exempt bond financing and low-income housing tax credits, the Project must satisfy minimum set-aside
choose the Average Income Test (AIT) if the Project is a re-syndication or has market rate units.

LIHTC Federal Minimum Set-Aside

A minimum of 40% of the Total Units will be rented to Residents with income at or below 60% or the Area Median Income.

Tax-Exempt Bonds Minimum Federal Set-Aside

A minimum of 40% of the Total Units will be rented to Residents with income at or below 60% or the Area Median Income.

The options available in these two menus are driven off of the data in the Application Set Up Data table.

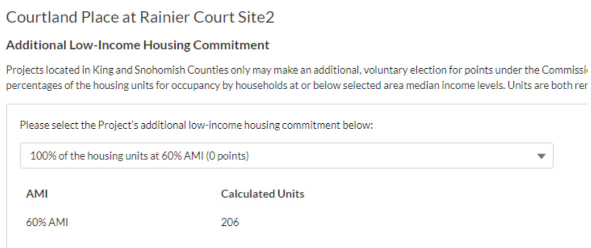
- The Data Type = "Federal Set Aside LIHTC" is used for the LIHTC Federal Minimum Set-Aside menu.
- The Data Type = "Federal Set Aside Tax-Exempt Bonds" is used for the Tax-Exempt Bonds Minimum Federal Set-Aside menu.
- The options displayed are different for 4% and 9% and the Application Type column is used to differentiate between the two.
- The order of the items in the menu is determined by the Display Order.
- The Start Date and End Date are used to determine which set of records to use. We only have one set of records, but if at some point in the future we wanted to have multiple sets of records based on date, we could add date ranges. It would use the Allocation Round Application Due Date to determine which set of records to use. *Note that this date range functionality hasn't been tested and may need to be modified/enhanced when it is used.*
- The Value Set Type column indicates the value that is stored in the database when an option is chosen.
  - Data is stored in Salesforce in
    - Funding__c.Federal_Election__c (Federal Election LIHTC)
    - Funding__c.Federal_Set_Aside_Tax_Exempt_Bonds__c
- The Value column indicates the value displayed to the end user.

	A	D	E	M	N	P	Q	R	S	U	V
1	Application_Type_	Data_Type__c	Display_Order_	Name	Number_	Parent_Application_Set_Up_Data_	Percent_	Start_Date_	End_Date_	Value_Set_Type__c	Value__c
81	4%	Federal Set Aside LIHTC	1	AD-00001				1/1/2022		20% at 50% AMI	A minimum of 20% of the Total Units will be rented to R
82	4%	Federal Set Aside LIHTC	2	AD-00002				1/1/2022		40% at 60% AMI	A minimum of 40% of the Total Units will be rented to R
83	4%	Federal Set Aside LIHTC	3	AD-00003				1/1/2022		Income Averaging	Average Income - A minimum of 40% of the Total Housir
84	9%	Federal Set Aside LIHTC	1	AD-00004				1/1/2022		20% at 50% AMI	A minimum of 20% of the Total Units will be rented to R
85	9%	Federal Set Aside LIHTC	2	AD-00005				1/1/2022		40% at 60% AMI	A minimum of 40% of the Total Units will be rented to R
86	9%	Federal Set Aside LIHTC	3	AD-00006				1/1/2022		Income Averaging	Average Income - A minimum of 40% of the Total Housir
87	4%	Federal Set Aside Tax-Exempt Bonds	1	AD-00007				1/1/2022		20% at 50% AMI	A minimum of 20% of the Total Units will be rented to R
88	4%	Federal Set Aside Tax-Exempt Bonds	2	AD-00008				1/1/2022		40% at 60% AMI	A minimum of 40% of the Total Units will be rented to R
195											

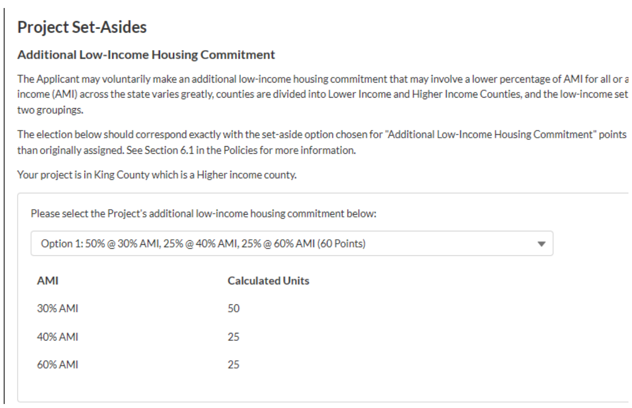
State Set Asides with AMI Set Up Data and County Specific Logic

When filling out the section of the screen related to Additional Low-Income Housing Commitments, the applicant is asked to provide information about the project's additional low-income housing commitment. The interface varies slightly with different instructions and also some county based logic based on which type of application it is:

4% Application:



9% Application:



The options available in this menu are driven off of the data in the Application Set Up Data table.

- The Data Type = "Counties by Type" is used to help determine which set of State Set Asides to display.
- The Data Type = "State Set Asides" is used for the Additional Low Income Housing Commitment menu.
- For both data types, the options/logic displayed are different for 4% and 9% and the Application Type column is used to differentiate between the two.
- The order of the items in the menu is determined by the Display Order.
- The Start Date and End Date are used to determine which set of records to use. We only have one set of records, but if at some point in the future we wanted to have multiple sets of records based on date, we could add date ranges. It would use the Allocation Round Application Due Date to determine which set of records to use. *Note that this date range functionality hasn't been tested and may need to be modified/enhanced when it is used.*

- Counties by Type

	Application_Type_	Data_Type__c	Display_Order_	Name	Number_	Parent_Application_Set_Up_Data_	Percent_	Start_Date_	End_Date_	Value_Set_Type__c	Value__c
2	4%	Counties by Type	1	AD-00009				1/1/2022		King & Snohomish	King
3	4%	Counties by Type	2	AD-00010				1/1/2022		King & Snohomish	Snohomish
4	4%	Counties by Type	3	AD-00158				1/1/2022		All Other Counties	Adams
5	4%	Counties by Type	4	AD-00159				1/1/2022		All Other Counties	Asotin
6	4%	Counties by Type	5	AD-00160				1/1/2022		All Other Counties	Benton
7	4%	Counties by Type	6	AD-00161				1/1/2022		All Other Counties	Chelan

	Application_Type_	Data_Type__c	Display_Order_	Name	Number_	Parent_Application_Set_Up_Data_	Percent_	Start_Date_	End_Date_	Value_Set_Type__c	Value__c
41	9%	Counties by Type	1	AD-00011				1/1/2022		Lower	Adams
42	9%	Counties by Type	2	AD-00012				1/1/2022		Lower	Asotin
43	9%	Counties by Type	3	AD-00013				1/1/2022		Higher	Benton
44	9%	Counties by Type	4	AD-00014				1/1/2022		Lower	Chelan
45	9%	Counties by Type	5	AD-00015				1/1/2022		Lower	Clallam
46	9%	Counties by Type	6	AD-00016				1/1/2022		Higher	Clark
47	9%	Counties by Type	7	AD-00017				1/1/2022		Lower	Columbia
48	9%	Counties by Type	8	AD-00018				1/1/2022		Lower	Cowlitz
49	9%	Counties by Type	9	AD-00019				1/1/2022		Lower	Douglas
50	9%	Counties by Type	10	AD-00020				1/1/2022		Lower	Ferry
51	9%	Counties by Type	11	AD-00021				1/1/2022		Higher	Franklin
52	9%	Counties by Type	12	AD-00022				1/1/2022		Higher	Garfield

  - The Value Column indicates a county name. This is used to match against the county provided in the Application. Because of the various situations we can encounter with an application, the county name may come from one of these places:
    - Site__r.County__r.Name
      - Used if the project is a portfolio project and is entering data by Site.
    - Application__r.Project_Update__r.County__r.Name
      - Used if the project was pre-existing and we don't want to update the actual project record.
    - Funding__r.County__c
      - Used if the existing project address is fine and there is no project update record which typically happens if this is a brand new project.
  - The Value Set Type column indicates the categorization of the county:
    - For 4% the categorization is either it is in King & Snohomish counties or it is not.
    - For 9% the categorization indicates if the county is in a higher income or lower income county.
- State Set Asides and AMI Set Up Data

	Application_Type_	Data_Type__c	Display_Order_	Name	Number_	Parent_Application_Set_Up_Data_	Percent_	Start_Date_	End_Date_	Value_Set_Type__c	Value__c	C
156	4%	State Set Asides	1	AD-00195				1/1/2022		All Other Counties	100% of the housing units at 60% AMI (0 points)	
157	4%	State Set Asides	1	AD-00157				1/1/2022		King & Snohomish	100% of the housing units at 60% AMI (0 points)	
158	4%	State Set Asides	2	AD-00050				1/1/2022		King & Snohomish	70% of the housing units at 60% AMI, 30% at 50% AMI (2 points)	
159	4%	State Set Asides	3	AD-00051				1/1/2022		King & Snohomish	50% of the housing units at 60% AMI, 50% at 50% AMI (4 points)	
160	4%	State Set Asides	4	AD-00052				1/1/2022		King & Snohomish	30% of the housing units at 60% AMI, 70% at 50% AMI (6 points)	
161	9%	State Set Asides	1	AD-00053				1/1/2022		Higher	Option 1: 50% @ 30% AMI, 25% @ 40% AMI, 25% @ 60% AMI (60 Points)	
162	9%	State Set Asides	1	AD-00070				1/1/2022		Lower	Option 1: Not Available in Lower Income Counties	
163	9%	State Set Asides	2	AD-00054				1/1/2022		Higher	Option 2: 50% @ 30% AMI, 50% @ 50% AMI (60 Points)	
164	9%	State Set Asides	2	AD-00071				1/1/2022		Lower	Option 2: Not Available in Lower Income Counties	
165	9%	State Set Asides	3	AD-00055				1/1/2022		Higher	Option 3: 50% @ 30% AMI, 30% @ 50% AMI, 20% @ 60% AMI (58 Points)	
166	9%	State Set Asides	3	AD-00072				1/1/2022		Lower	Option 3: Not Available in Lower Income Counties	
167	9%	State Set Asides	4	AD-00056				1/1/2022		Higher	Option 4: 25% @ 30% AMI, 50% @ 40% AMI, 25% @ 60% AMI (56 Points)	
168	9%	State Set Asides	4	AD-00073				1/1/2022		Lower	Option 4: 25% @ 30% AMI, 50% @ 40% AMI, 25% @ 60% AMI (60 Points)	
169	9%	State Set Asides	5	AD-00057				1/1/2022		Higher	Option 5: 50% @ 30%, 25% @ 50% AMI, 25% @ 60% AMI (56 Points)	
170	9%	State Set Asides	5	AD-00074				1/1/2022		Lower	Option 5: Not Available in Lower Income Counties	
171	9%	State Set Asides	6	AD-00058				1/1/2022		Higher	Option 6: Not Available in Higher Income Counties	

  - The Value Set Type column matches to the county categorization determined by the Counties by Type data set.
  - The Value column indicates which options to display to the end user. The options are different based on Application Type and Value Set Type.
  - There are AMI Set Up Data records related to each State Set Aside record. The AMI Set Up Data determines which LIH Set Aside records to create.

	AMI_Value__c	AMI__c	Name	Percent_of_Units__c	Application_Set_Up_Data_Name
2	50% AMI	50	AM-00070	50	State Set Asides (0.04) King & Snohomish; 50% of the housing units at 60% AMI, 50% at 50% AMI (4 points)
3	50% AMI	50	AM-00072	70	State Set Asides (0.04) King & Snohomish; 30% of the housing units at 60% AMI, 70% at 50% AMI (6 points)
4	30% AMI	30	AM-00001	50	State Set Asides (0.09) Higher; Option 1: 50% @ 30% AMI, 25% @ 40% AMI, 25% @ 60% AMI (60 Points)
5	40% AMI	40	AM-00002	25	State Set Asides (0.09) Higher; Option 1: 50% @ 30% AMI, 25% @ 40% AMI, 25% @ 60% AMI (60 Points)
6	60% AMI	60	AM-00003	25	State Set Asides (0.09) Higher; Option 1: 50% @ 30% AMI, 25% @ 40% AMI, 25% @ 60% AMI (60 Points)

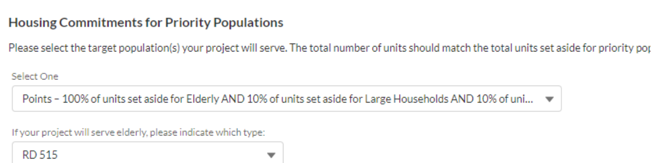
    - The AMI Value and AMI fields indicate the percentage of AMI.

        - The Percent of Units field indicates how many units should be counted in each AMI category.
    - Data is stored in Salesforce in
        - LIH_Set_Aside__c.Set_Aside_Type__c = Tax Credits
        - LIH_Set_Aside__c.Active__c = True
        - LIH_Set_Aside__c.Record_Source__c = Units and Sq Ft
            - Used to identify which records this process created
        - LIH_Set_Aside__c.Description__c = the AMI_Set_Up_Data__c.AMI_Value__c
        - LIH_Set_Aside__c.Unit_Count__c = calculated from Units * percent of units
            - Units - Elderly is 100% of units which includes LIH, Common Area and Market Rate units. All other set-asides (special population and LIH set-asides) are calculated based off the LIH units only.
        - LIH_Set_Aside__c.Unit_Percent__c = AMI_Set_Up_Data__c.Percent_of_Units__c
    - In many cases, multiple LIH records are created based on the option chosen.

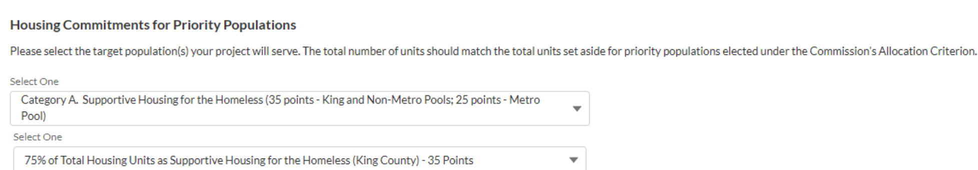
Special Populations

The interface for special populations is slightly different based on whether the application is for 4% or 9%.

4% Application:



9% Application:



There are several data sets used to drive this interface and they work differently for 4% than 9%.

4% Special Population Logic:

	A	D	E	M	N	P	Q	R	S	U	V	W
1	Application_Type_	Data_Type__c	Display_Order_	Name	Number_	Parent_Application_Set_Up_Data_	Percent_	Start_Date_	End_Date_	Value_Set_Type__c	Value__c	C
151	4%	Special Populations	2	AD-00088				1/1/2022			Threshold â€“ 10% of units set aside for Large Households AND 1	
152	4%	Special Populations	3	AD-00089				1/1/2022			Threshold â€“ 20% of units set aside for Persons with Disabilitie	
153	4%	Special Populations	4	AD-00090				1/1/2022			Points â€“ 20% of units set aside for Large Households AND 10%	
154	4%	Special Populations	5	AD-00091				1/1/2022		Elderly	Points â€“ 100% of units set aside for Elderly AND 10% of units se	
155	4%	Special Populations	6	AD-00154				1/1/2022		Elderly	Points â€“ 100% of units set aside for Elderly AND 20% of units se	
195												

1	Application_Type_	Data_Type__c	Display_Order_	Name	Number_	Parent_Application_Set_Up_Data_	Percent_	Start_Date_	End_Date_	Value_Set_Type__c	Value__c	C
137	4%	SP Records to Create		AD-00149		a628H000000nQOwQAM	20	1/1/2022		Large Household	20% of units set aside for Large Households	
138	4%	SP Records to Create		AD-00151		a628H000000nQOxQAM	100	1/1/2022		Elderly	100% of units set aside for Elderly	
139	4%	SP Records to Create		AD-00146		a628H000000nQOuQAM	10	1/1/2022		Large Household	10% of units set aside for Large Households	
140	4%	SP Records to Create		AD-00147		a628H000000nQOuQAM	10	1/1/2022		Disabled	10% of units set aside for Persons with Disabilities	
141	4%	SP Records to Create		AD-00150		a628H000000nQOwQAM	10	1/1/2022		Disabled	10% of units set aside for Persons with Disabilities	
142	4%	SP Records to Create		AD-00155		a628H000000nSJbQAM	100	1/1/2022		Elderly	100% of units set aside for Elderly	
143	4%	SP Records to Create		AD-00153		a628H000000nQOxQAM	10	1/1/2022		Disabled	10% of units set aside for Persons with Disabilities	
144	4%	SP Records to Create		AD-00156		a628H000000nSJbQAM	20	1/1/2022		Disabled	20% of units set aside for Persons with Disabilities	
145	4%	SP Records to Create		AD-00152		a628H000000nQOxQAM	10	1/1/2022		Large Household	10% of units set aside for Large Households	
146	4%	SP Records to Create		AD-00148		a628H000000nQOvQAM	20	1/1/2022		Disabled	20% of units set aside for Persons with Disabilities - 0 points	

- The 4% application uses the Data Type of "Special Populations" to determine the list of items to display.
- It uses the Value Set Type field to determine if the selected option has special processing considerations. In this case, whether the selected option involves elderly or not.
- If an option involves elderly, then an additional question is displayed asking what type of elderly.
- Based on the selected option there is an additional data set "SP Records to Create" that each have a parent record of one of the Special Population records. Multiple SP Records to Create records may be related to a single Special Population record thus causing the system to create multiple special population records in Salesforce.
- Data is created as follows:
    - Special_Populations__c. Record_Source__c = Units and Sq Ft
        - Used to identify which records this process created
    - Special_Populations__c.Active__c = True
    - Special_Populations__c.Site__c (populated if data entered by site)
    - Special_Populations__c.Description__c = Application_Set_Up_Data__c.Value_Set_Type__c
    - Special_Populations__c.Unit_Count__c = calculated from Units * Application_Set_Up_Data__c.Percent__c
        - Units - Elderly is 100% of units which includes LIH, Common Area and Market Rate units. All other set-asides (special population and LIH set-asides) are calculated based off the LIH units only.
    - Special_Populations__c.Unit_Percent__c = Application_Set_Up_Data__c.Percent__c
    - Special_Populations__c.Description_Value__c = Application_Set_Up_Data__c.Value__c

9% Special Population Logic:

1	Application_Type_	Data_Type__c	Display_Order_	Name	Number_	Parent_Application_Set_Up_Data_	Percent_	Start_Date_	End_Date_	Value_Set_Type__c	Value__c	C
147	9%	Special Population Categories	1	AD-00092				1/1/2022		None	No Points Taken	
148	9%	Special Population Categories	2	AD-00093	1			1/1/2022		Homeless Special Popul:	Category A. Supportive Housing for the Homeless (35 points -	
149	9%	Special Population Categories	3	AD-00094	1			1/1/2022		Farmworker Special Pop	Category B. 75% Farmworker Housing (35 points)	
150	9%	Special Population Categories	4	AD-00095	2			1/1/2022		Other Special Populatior	Category C. Housing Commitments for other Priority Populati	
195												

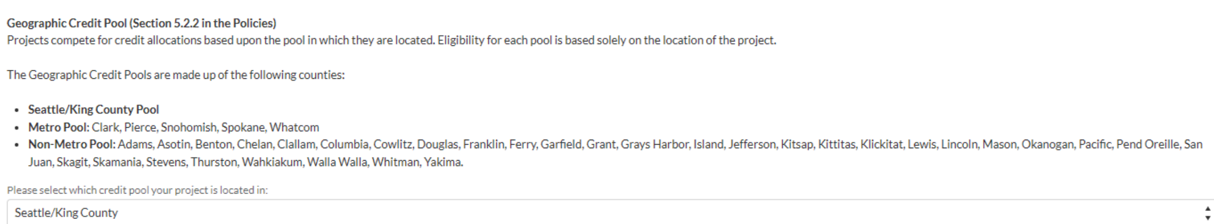
1	Application_Type_	Data_Type__c	Display_Order_	Name	Number_	Parent_Application_Set_Up_Data_	Percent_	Start_Date_	End_Date_	Value_Set_Type__c	Value__c	C
80	9%	Farmworker Special Populations	1	AD-00099			75	1/1/2022		Farmworker	75% of Total Housing Units for Farmworkers (Metro and	
128	9%	Homeless Special Populations	1	AD-00096			75	1/1/2022		Homeless	75% of Total Housing Units as Supportive Housing for th	
129	9%	Homeless Special Populations	2	AD-00097			50	1/1/2022		Homeless	50% of Total Housing Units as Supportive Housing for th	
130	9%	Homeless Special Populations	3	AD-00098			50	1/1/2022		Homeless	50% of Total Housing Units as Supportive Housing for th	
131	9%	Homeless Special Populations	4	AD-00144			25	1/1/2022		Homeless	25% of Total Housing Units as Supportive Housing for th	
132	9%	Other Special Populations	1	AD-00100			20	1/1/2022		Farmworker	20% of Total Housing Units for Farmworkers - 10 Points	
133	9%	Other Special Populations	2	AD-00101			20	1/1/2022		Large Household	20% of Total Housing Units for Large Households - 10 Pc	
134	9%	Other Special Populations	3	AD-00102			20	1/1/2022		Disabled	20% of Total Housing Units for Persons with Disabilities	
135	9%	Other Special Populations	4	AD-00103			20	1/1/2022		Homeless	20% of Total Housing Units for Permanent Housing for H	
136	9%	Other Special Populations	5	AD-00104			100	1/1/2022		Elderly	100% of Total Housing Units for Elderly - 10 Points	

- The 9% application uses the Data Type of "Special Population Categories" to determine the list of items to display.
    - The Value__c field is used to determine the options to show in the picklist.

    - It uses the Value Set Type field to determine if the selected option has special processing considerations. In this case, there is a secondary picklist that is displayed based on the options selected.
- The secondary picklist is driven off of the following Data Set Types:
    - Farmworker Special Populations
    - Homeless Special Populations
    - Other Special Populations
    - The Value field is used to determine the options to show in the secondary picklist.
- If an option involves elderly (Based on Value Set Type), then an additional question is displayed asking what type of elderly.
- If Other Special Populations option is chosen, then it gives the user the option to select two options.
- Data is then created as follows:
    - Special_Populations__c. Record_Source__c = Units and Sq Ft
        - Used to identify which records this process created
    - Special_Populations__c.Active__c = True
    - Special_Populations__c.Site__c (populated if data entered by site)
    - Special_Populations__c.Description__c = Application_Set_Up_Data__c.Value_Set_Type__c
    - Special_Populations__c.Unit_Count__c = calculated from Units * Application_Set_Up_Data__c.Percent__c
        - Units - Elderly is 100% of units which includes LIH, Common Area and Market Rate units. All other set-asides (special population and LIH set-asides) are calculated based off the LIH units only.
    - Special_Populations__c.Unit_Percent__c = Application_Set_Up_Data__c.Percent__c
    - Special_Populations__c.Description_Value__c = Application_Set_Up_Data__c.Value__c

Geographic Credit Pool

The geographic credit pool data is the only data set that is not used in the **Sites, Buildings & Units** checklist item. Instead this is used in the **Project Summary** checklist item for 9% applications.



Geographic Credit Pool (Section 5.2.2 in the Policies)
Projects compete for credit allocations based upon the pool in which they are located. Eligibility for each pool is based solely on the location of the project.

The Geographic Credit Pools are made up of the following counties:

- **Seattle/King County Pool**
- **Metro Pool:** Clark, Pierce, Snohomish, Spokane, Whatcom
- **Non-Metro Pool:** Adams, Asotin, Benton, Chelan, Clallam, Columbia, Cowlitz, Douglas, Franklin, Ferry, Garfield, Grant, Grays Harbor, Island, Jefferson, Kitsap, Kittitas, Klickitat, Lewis, Lincoln, Mason, Okanogan, Pacific, Pend Oreille, San Juan, Skagit, Skamania, Stevens, Thurston, Wahkiakum, Walla Walla, Whitman, Yakima.

Please select which credit pool your project is located in:
Seattle/King County

- The default value of the Geographic Credit Pool comes from the Geographic Credit Pool Application Set Up Data:

	A	D	E	M	N	P	Q	R	S	U	
1	Application_Type_	Data_Type__c	Display_Order_	Name	Number_	Parent_Application_Set_Up_Data_	Percent_	Start_Date_	End_Date_	Value_Set_Type__c	Value__c
89	9%	Geographic Credit Pool	1	AD-00105				1/1/2022		Non-Metro	Adams
90	9%	Geographic Credit Pool	2	AD-00106				1/1/2022		Non-Metro	Asotin
91	9%	Geographic Credit Pool	3	AD-00107				1/1/2022		Non-Metro	Benton
92	9%	Geographic Credit Pool	4	AD-00108				1/1/2022		Non-Metro	Chelan
93	9%	Geographic Credit Pool	5	AD-00109				1/1/2022		Non-Metro	Clallam
94	9%	Geographic Credit Pool	6	AD-00110				1/1/2022		Metro	Clark
95	9%	Geographic Credit Pool	7	AD-00111				1/1/2022		Non-Metro	Columbia
96	9%	Geographic Credit Pool	8	AD-00112				1/1/2022		Non-Metro	Cowlitz
97	9%	Geographic Credit Pool	9	AD-00113				1/1/2022		Non-Metro	Douglas
98	9%	Geographic Credit Pool	10	AD-00114				1/1/2022		Non-Metro	Ferry
99	9%	Geographic Credit Pool	11	AD-00115				1/1/2022		Non-Metro	Franklin
100	9%	Geographic Credit Pool	12	AD-00116				1/1/2022		Non-Metro	Garfield
101	9%	Geographic Credit Pool	13	AD-00117				1/1/2022		Non-Metro	Grant
102	9%	Geographic Credit Pool	14	AD-00118				1/1/2022		Non-Metro	Grays Harbor
103	9%	Geographic Credit Pool	15	AD-00119				1/1/2022		Non-Metro	Island
104	9%	Geographic Credit Pool	16	AD-00120				1/1/2022		Non-Metro	Jefferson
105	9%	Geographic Credit Pool	17	AD-00121				1/1/2022		King County	King

- Based on the County provided in the Application, the Value field is used to lookup the Value Set Type to get whether the default Geographic Credit Pool should be Metro, Non-Metro, or King County.
- The data is then stored in the Funding__c.Geographic_Pool__c field.