Throughout this course we have created two levels for each exercise. This should allow everybody, from a starter to a more seasoned programmer, to find a challenge in these exercises.

We will provide an existing Visual Studio 2019 solution that can be used during this course that will have snippets of code combined with a “task list” to work on.

However, if you prefer a clean solution you can start a new solution following the guidelines in the supplied document “***4-Series and VC-4 C# Development Instructions”***

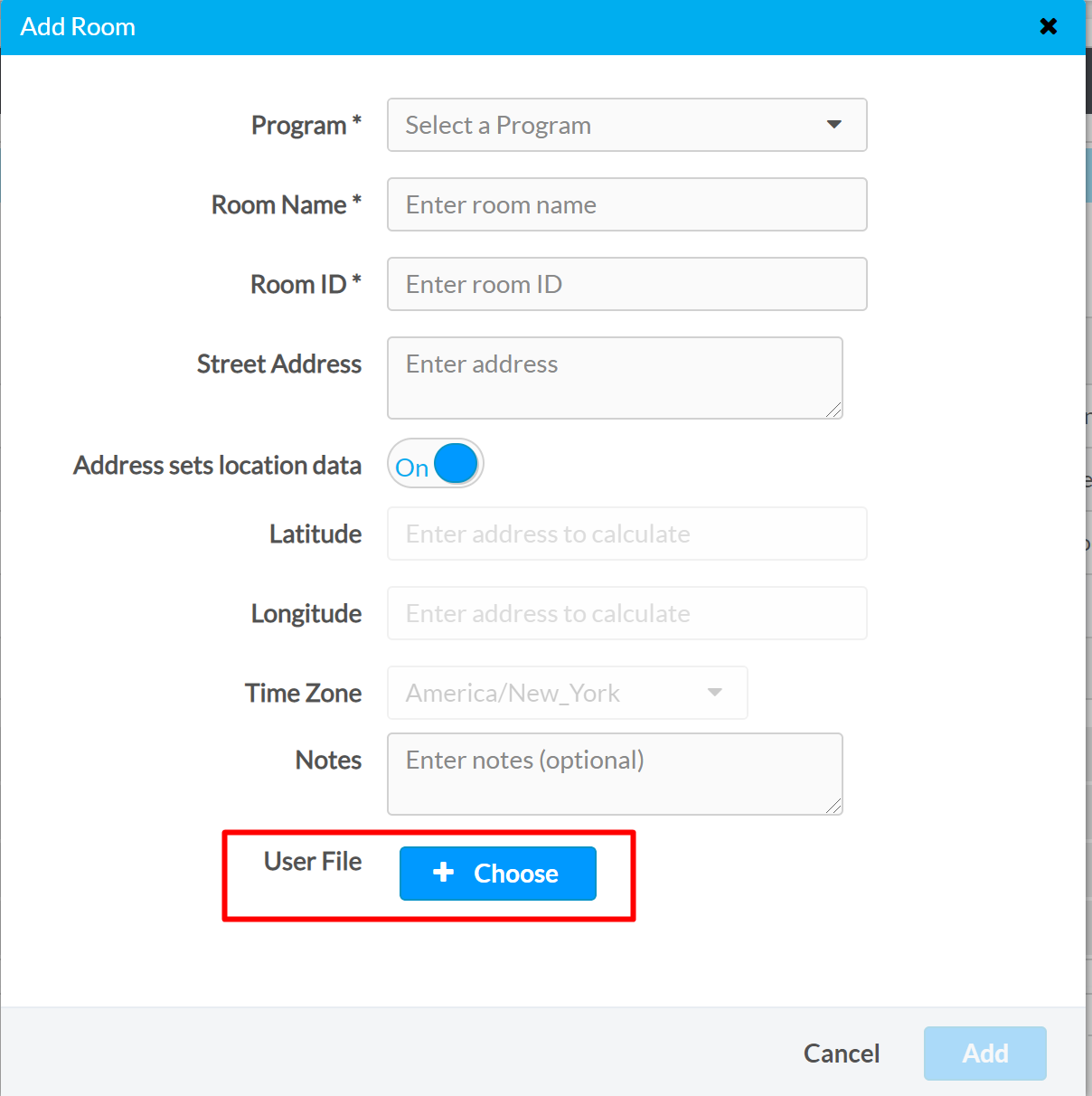
# Exercise 1 – Dynamic Registration

This first Lab will focus around the dynamic registration and configuration of devices.

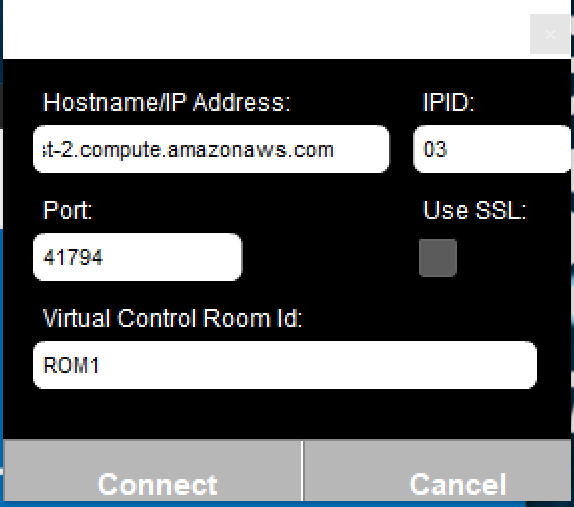
We have already set up a base solution the handles loading a configuration file. For anybody that has also followed the MSS-501/502 course the classes we have used to read in the config file will look very familiar.

The configuration file (*config.json*) is already provided in the **Documentation** folder. Please use this config file throughout the course.

Since we are dynamically loading this config file, don’t forget to add it as a **User File** when creating your room in VC-4:



When using the provided Xpanel to test functionality, make sure you point to the hostname or IP address of the supplied VC-4 instance in combination with the ROOMID by selecting “Options” 🡪 “Host Settings”, or pressing Ctrl+H



For exercise 1, we will be using a single “Management” class (SystemManager.cs) that will handle device registration as well as device configuration.

We have again provided an XPanel that can be used, that simply consists of two Dynamic List objects that we want to setup correctly based on the provided JSON config file.

**Level 1**

From the *SystemManager.cs* file, please implement the dynamic registration of the touchpanels as they have been defined in the configuration file.

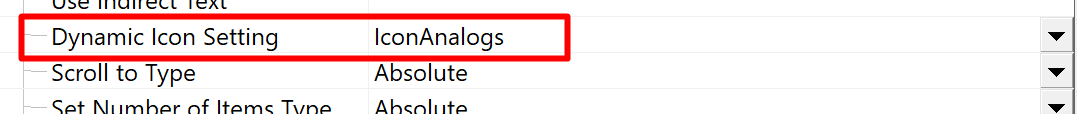
A lot of the heavy lifting is already done in the *TouchpanelUI.cs* file. You can create an instance of it in *SystemManager* to use it.

After you have registered the touchpanels, you will have to set them up.

Functionality we are looking for:

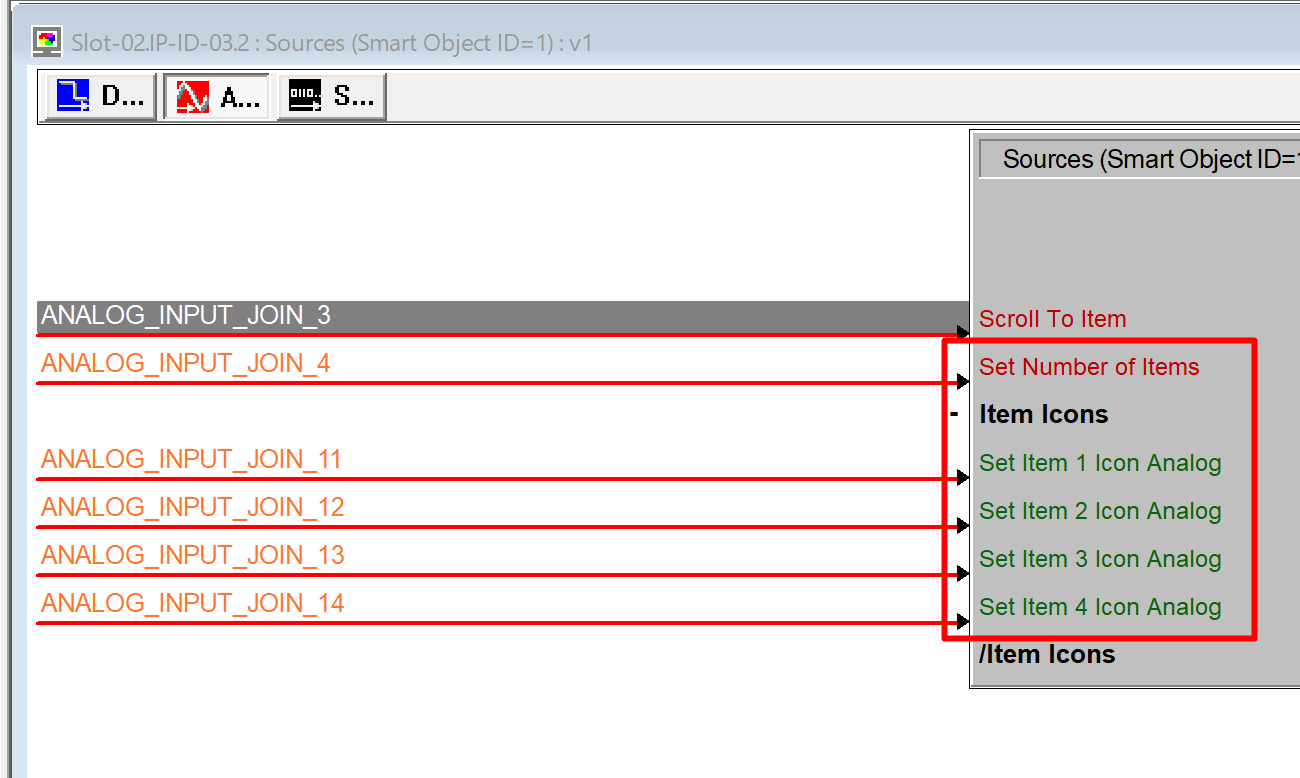
* There is currently one source defined in the config file. Expand the config file to add more sources.
* There is currently one destination defined in the config file. Expand the config file to add more destinations.
* Set up the two Smart Objects (Smart Object ID 1 & 2) to be dynamically configured using the information from the configuration file.
  + Change the size of the Dynamic List object to match the number of sources/destinations defined
  + Change the label for all sources/destinations to be dynamic
  + Change the icon for all sources/destinations to be dynamic

**Hint**: The icons can be set up using “Dynamic Icon Setting”



We have provided you a document called **“*Icons\_State\_ID”*** in the Documentation folder.

**Hint:** In SIMPL Windows, these are the properties you would use:



**Level 2**

After finishing the exercise for Level 1, you may continue with Level 2.

The additional exercise for this level is to add another *subsystem* to the configuration file that will allow you to dynamically register DM-NVX devices.

Functionality we are looking for:

* A way to define which NVX device you want to register:
  + DM-NVX350(-C)
  + DM-NVX351(-C)
* Being able to dynamically load and register these devices to your system.