AUTODESK UNIVERSITY 2014

CO5168 Virtual Installation: Augmented Realty of Install Details

Tim Riefenberg – CHOATE Construction – Virtual Construction Specialist

CO5168 A difficult task comes with constructing a complex building: properly showing those who are building how to build the building. The next stage comes with the further advancement of 3D modeling: erecting building parts and assemblies on a mobile device. This class is targeted toward designers, fabricators, and installers who want to learn how to properly demonstrate the way to build their products through augmented realty animation. They can then deliver these animations to the on-site installers who can virtually see components come together in their physical areas. This class will focus on using models from Revit software and/or Inventor software and recreating a building animation utilizing Augmented Reality software.

Learning Objectives

At the end of this class, you will be able to:

- Learn how to export pieces and parts into an Augmented Reality Viewer
- Learn about animating an erection of an building assembly
- Learn how to more accurately show a detail on a building assembly
- Learn how to properly convert geometry and materials into an Augmented Reality Viewer

About the Speaker

Tim received his bachelor's degree from Savannah College of Art and Design (S.C.A.D.) in both Architecture and Visual Effects in 2005. He is currently working for CHOATE Construction as a Virtual Construction Specialist in Atlanta, Georgia. Where he provides BIM implementation and other field based technologies. Tim is a regular attendee of Autodesk University and has in the past spoken at Revit Technology Conference USA. Tim's real passion is to inspire younger youths to follow a career in Architecture or Engineering; either through the use of mentorship and or being a guest speaker at local schools.

Contact Info: triefenberg@choateco.com

Screencast User Id: Triefenberg

Youtube: triefenberg

Overall Class Summary

Learn how to take models from Revit and view them in Augmented Realty. This class will be an overview on the processes needed to achieve this task. With the combination of lessons learned and best practices ever user will be able to recreate this process on their own. Through a live demonstration as well as an detailed how to in this document

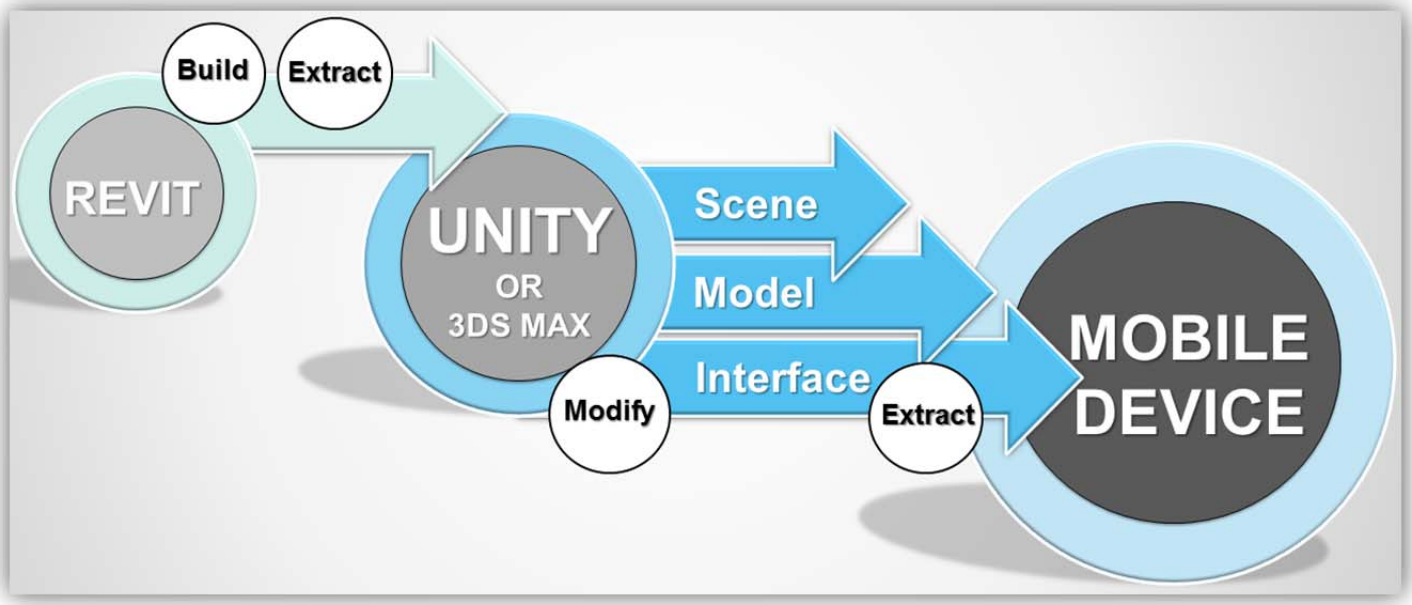
Use for Augmented Realty

What is Augmented Realty? Well simply put it is a merger of Realty and a Virtual Environment. With the use of a mobile viewer such as an smartphone or tablet one can utilize the devices camera to view this virtual model overlay. Since the technology has been around for several years, there are a lot of apps that showcase a game or object. Well the next step is to take your model and showcase it. Currently there are companies that produce this service for a fee. Well not every company is willing to spend more then what they already have. So, the solution was clear, how do I use readily available software that my company has already purchased? Well today you'll find out!

Class Highlight

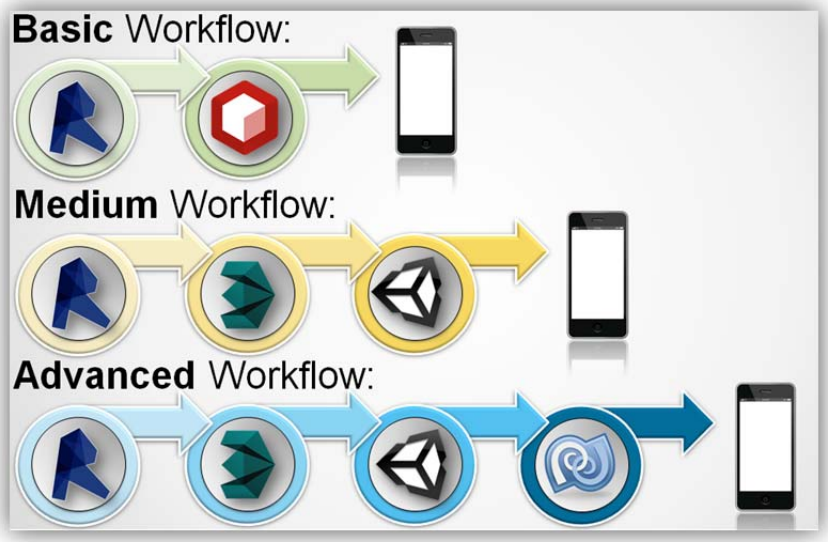
Everyone attending class today will be able to create their very own AR viewer that can be installed on any mobile device, regardless of skillset.

Process Overview



A brief overview of the process to be learned:

Class Structure Overview



Three workflows to learn today: Basic, Medium and Advanced

Workflow Overview:

Augmented Realty: Start to Finish

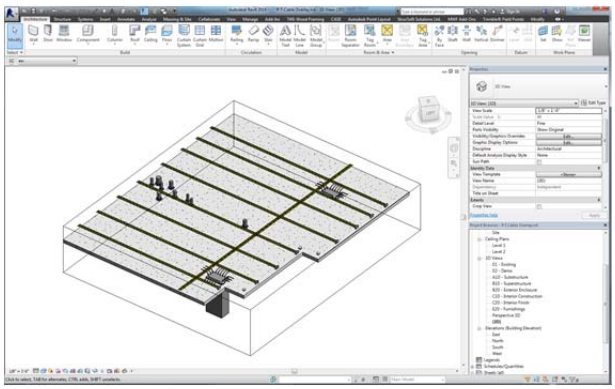
1. Workflow Basic:

Taking a Model from Revit into Augment(Commercially Free Viewer)

1. First open Revit; then choose the Model in which you would like to showcase.
2. Navigate to a 3D view, utilizing the section box, include the area in which you would like to showcase.

In this example I'm focusing on an Elevated Post Tension Slab.

Note: When utilizing a mobile viewer there are limitations on the amount of complexity shown. If a model has a lot of pipe it will be converted to polygons and could have millions of faces. It's good to review areas first and sometimes replace a 16 sided cylinder with a 4 sided polygon box.



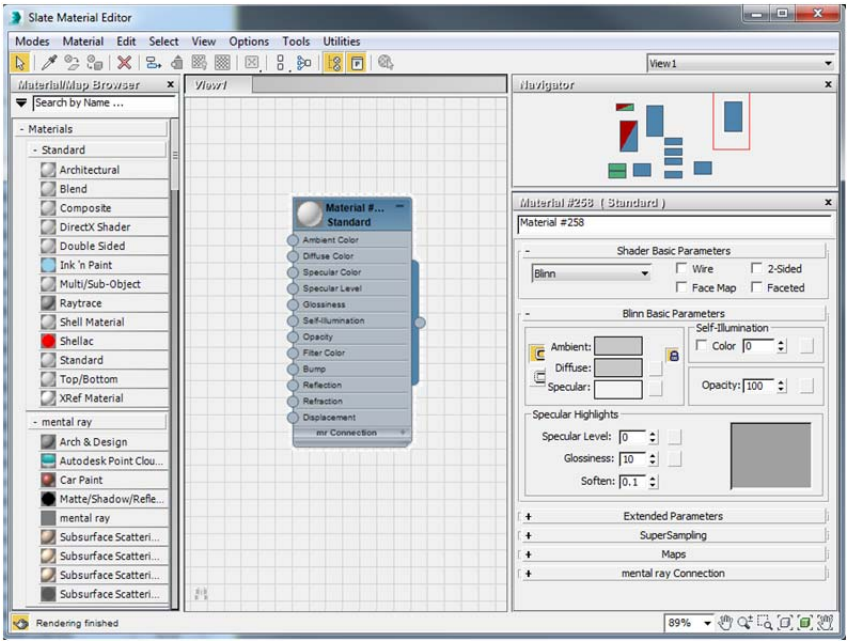
3. The Next step is to export the geometry out of Revit. Either by .fbx or .dwg format.

   (Later we will be importing this model into Unity is why we choose one format or the other)

   (Yes, 3ds Max also has a link to Revit project which is great, but sometimes you'll have to either change objects or recreate items to appropriate showcase in 3ds Max or Unity)

4. Open the most current Version of Autodesk 3Ds Max – Shown is 2015
5. Import the Geometry into your preferred scene

(A good working habit is to produce a scene template that has already the preferred lighting and material settings, Since most Augmented Realty Viewers have limitations on your device less complex a material is the better)



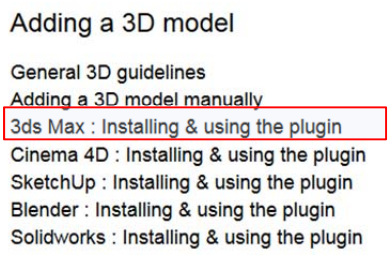
6. Reapply a **STANDARD** material to your objects in the scene.
7. Then adjust the lighting; either use the Standard Daylight System or a One Directional Light. – Less Lights the Better!
8. Install the Augment Plug-in

Plug in Is available from the Augment Website:

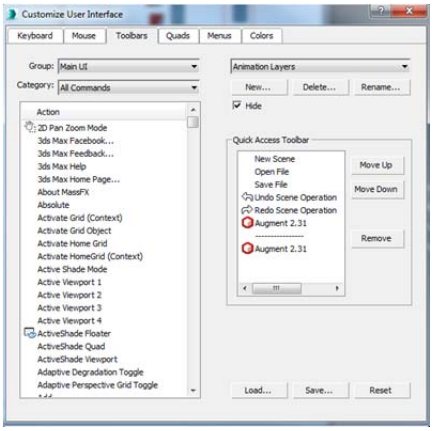
Currently Available for 3ds Max, Sketchup, Solidworks

http://augmentedev.com/

*Then Navigate to the **HELP** tab at the top of the website*



9. Once the Plug-In has been installed you simply drag the .MZP file format into 3ds Max Maxscript dialog box.
10. A window will appear asking you to Install the Augment 2.1 Plug-in
11. To load the Plug-in into you current Toolbar



Navigate to Customize

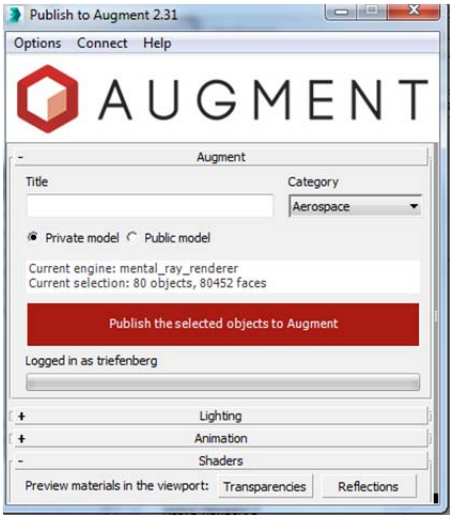
Then Choose Customize User Interface

In the Command list navigate to the Augment Plug-In; Then choose add to Quick Access Toolbar.

1.1 Installed Augment Plug-in

When the plug-in has been successfully installed, you'll be asked to create a free log-in on their site.

http://augmentedev.com/



You'll can create your own marker as well as showcase the model either privately or publically.

1.2 Uploading the Model to Augment

Isolate your scene to have the least amount of geometry and the most simplified textures applied.

In the Augment Lighting Section choose to include Bake Lighting,

The Publish the Selected Objects to Augment

2. Workflow Medium:

Taking a Model from Revit into a very basic Augmented Viewer provided from Unity and Qualcomm. This may include a lot of initial steps, downloading pieces and parts, but when you have the system down the workflow is very easy.

Items needed in this section listed below:

- Autodesk Revit 2015
- Autodesk 3Ds Max 2015
- Unity Gaming Application 4.5 (Free)
- Vuforia 3.0 Developer (Free – account login required)

Location to access Current Software or Plug-ins Required:

Unity - *http://unity3d.com/unity/download*

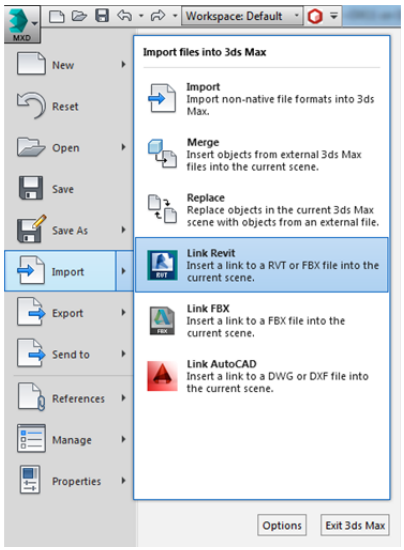
Vuforia -Qualcomm (Unity Asset) - *https://developer.vuforia.com/resources/sdk/unity*

Follow the websites guide for installation procedure, especially the Vuforia asset.

Place the asset in a centralized location, where you house all your Unity Assets.

1. In Revit Navigate to a 3D view, utilizing the section box, include the area in which you would like to showcase.
2. Export your Scene using the file format .FBX

Optional Process> If you're a newbie using Unity but are very comfortable using 3Ds Max follow these steps listed below here>



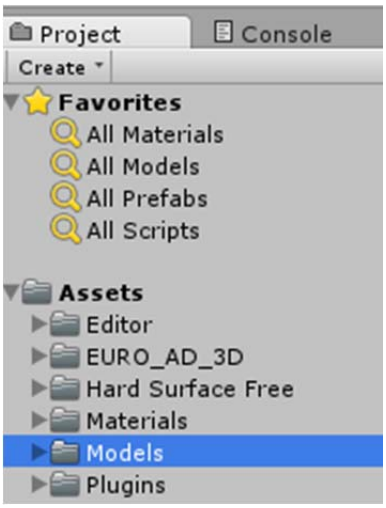
1. Export the Geometry out of Revit as an .FBX format or In 3Ds choose link Revit file as the import option.
2. Replace all the materials applied to all your objects with STANDARD shaders. Remember to create individual materials for categories of objects.
3. In 3Ds export the Selected items as an .OBJ file Format

Navigation in Unity as well as applying materials can be a challenge for a newbie, this step will let you apply your materials before bringing into Unity.

Continued Initial Step of Process Listed above

3. Open **Unity**

In Unity save your scene to an appropriate local folder, its key to save all your assets, materials, models and markers in the same scene folder for this project



4. In the Project folder under your Assets create a new folder called **Models**
5. When you double click the folder the right plan will list all the objects imported, simply Right Click and choose *Import New Asset*
6. Navigate and import your model
7. Replace the materials with Unity based materials, at this stage you can import free material assets available in the Unity asset store. Materials such as glass and metals are hard to view in the AR viewer.

2.1 Creating the QR marker

This process will create the local marker or clud based marker that will be the background image map. This Image will be the base point for viewing your model in the AR viewer.

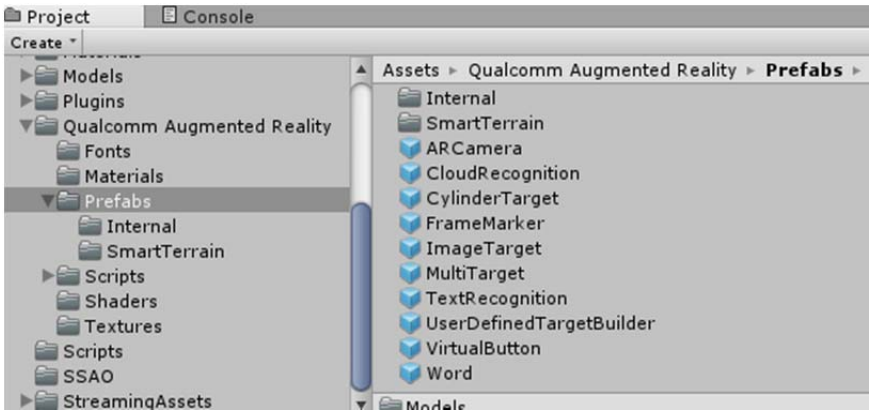
1. Install the Vuforia Asset available for Unity

Which is available at the following link:

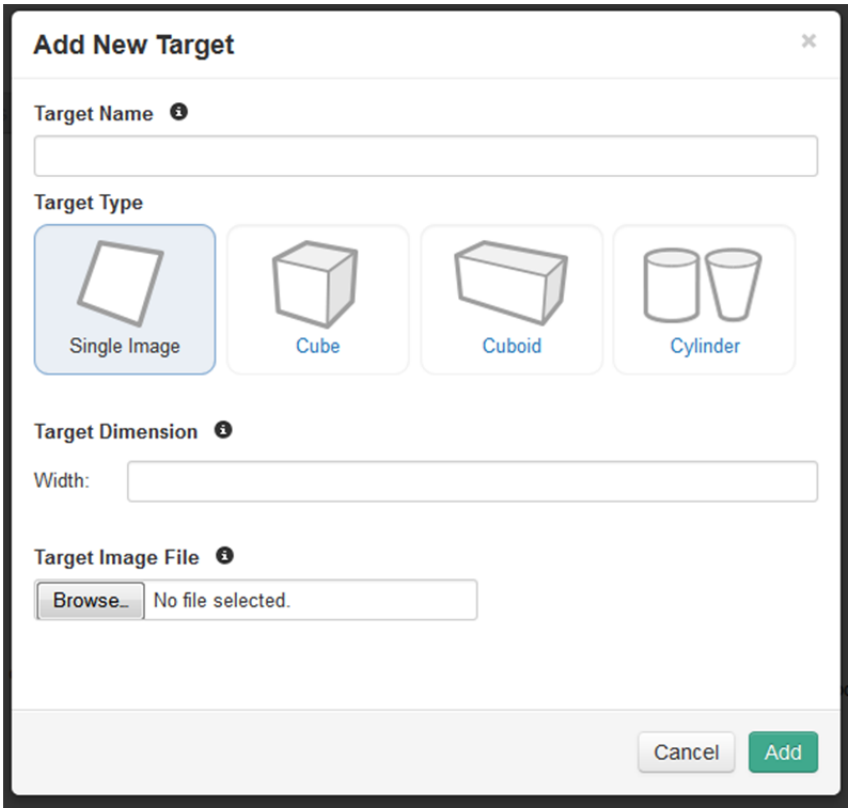
https://developer.vuforia.com/resources/sdk/unity

2. The Unity Extension plug-in will create an Asset package that will include the model Prefabs scripts needed to create an AR viewer.

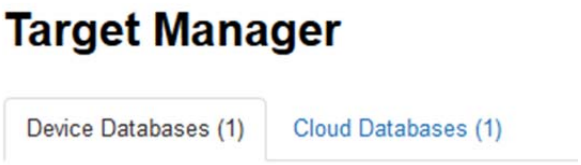
In the Package will also include an AR Camera and Marker needed to complete the process.



3. With your model placed in the Scene, now simply drag and drop the AR Camera, included in the AR package.
4. Then drag and drop the included ImageTarget Asset



There are a few new options to create an marker that were not available in the past. Now you have the ability to create 3D markers like a Cube, this is beneficial when you want to look at a building component that has been erected.



This ImageTarget is an Local Image marker: Meaning it will only be available at the individual device you have uploaded the content too. Vuforia has two options an Local Device Marker and a Cloud Based Marker, which other devices can recognize.

To Create a Cloud Based Marker

1. **Create a Log-in on the developer website for the Vuforia Asset.**

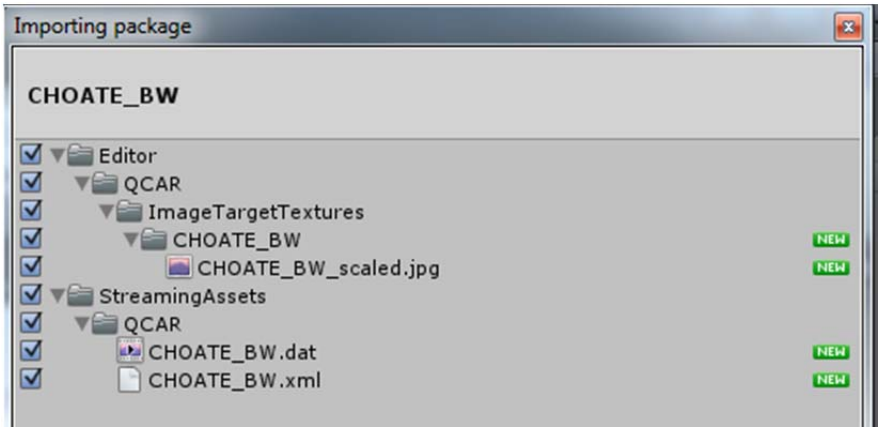
https://developer.vuforia.com/

2. Click on the **Target Manager** section
3. Listed will be two options **Device Database** and **Cloud Database**
4. **Choose Device Database for now**
5. In the Target Manager section choose **Create Database**

6. Make a name for the marker, (ie. Company name)
7. Once the database has been created now add the Target

8. The Target needs to be created from a High Contrast Image.

9. Vuforia will upload this image and a tracker id associated with image you've just uploaded. Then an Asset package will be created that then you will import into Unity Scene
10. In **Unity** Choose to Import a New Asset, Navigate to the downloaded marker package just created off **Vuforia**

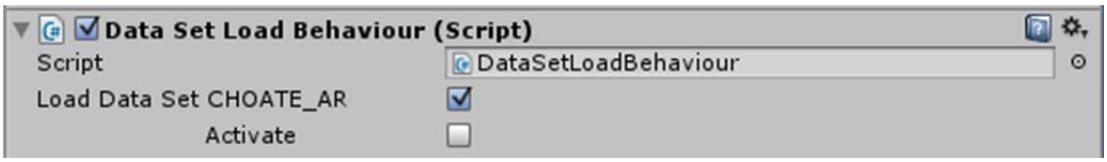


2.2 Applying the Local Marker to your Scene in Unity

At this point in the process you've created the first three steps in the overall process.(Model, Scene, and Marker)

Now there is one last step; to simply tie together your scene with a mobile device. With the marker just created we are now going to upload everything together onto your device using Unity.

1. In **UNITY** click on the **ARCAMERA** in your scene directory
2. Under the Inspector Plane for the Camera properties is a section to Load Dataset



1. Choose the Load Data Set, (your *named* database)

Now the Image has been attached to your camera, with any mobile device the scene will be overlaid to the object marker

2.3 Test the marker

Unity has an option to view your model via, a webcam.

In UNITY navigate to Edit > Project Settings > Editor

In the Editor Settings > Choose your device option>

WebCam

Any Android Device

Any ios Device

2.4 Viewing Model on an Android Device

In Unity Choose File> Build Settings

Select the Device System you would like to view your model in.

Example> Android Device

Plug-in Your Mobile Device to your computer through USB.

Note: Android Devices need to have developer options chosen to upload content directly to your phone.

The Unity App will then install on your mobile device with your Icon and now you'll be able to view your Model through AR.

3. Workflow Advanced:

This final process type in developing your own Augmented Realty viewer is a combination of all the types listed above with one new added part. This step will include some very basic scripting. Don't worry as the code is included in this session. Just like anything in technology you'll never get what you exactly need but you'll be able to customize your own with some very basic scripts.

The Unity AR viewer allows you to zoom in and rotate around the object. But the end user will want to either rotate the model or zoom in on their own.

Included below is a few options that can be included in the AR viewer.

Virtual Buttons: A Simple GUI

Touch Navigation: An added Asset available on Unity Asset store

Virtual Project Browser:

*Creating a **Virtual Button** that interacts with your environment. This is can be used as a way to change colors of an object or a stop play button in an animation.*

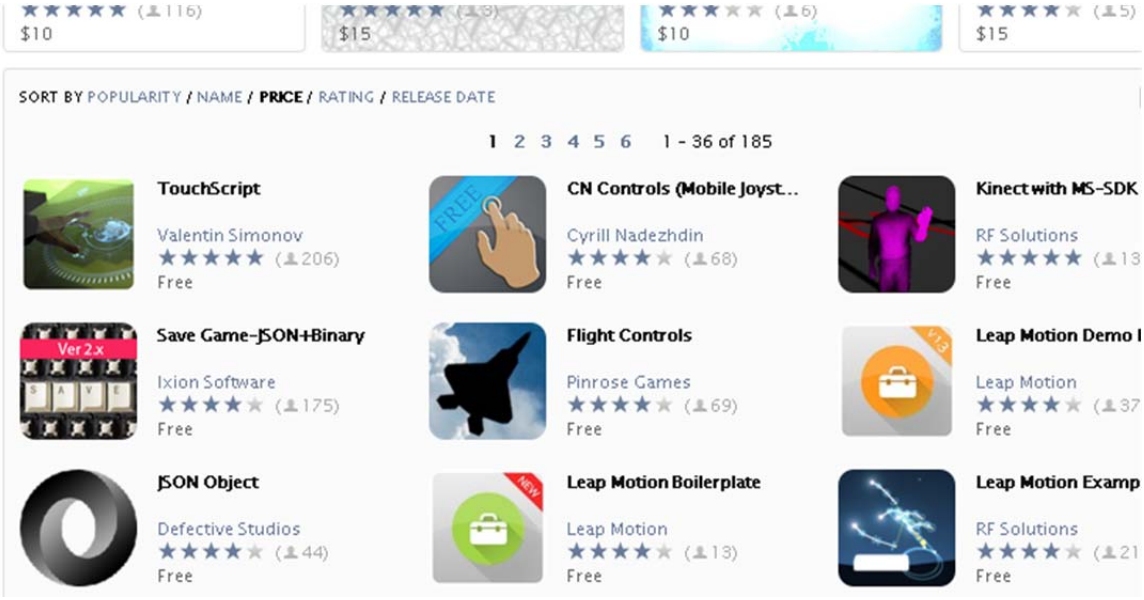
1. Open up MonoDeveloper, provided by Unity
2. Below is the Source code example, this code is provided by the Vuforia Plug in.


```
/*==============================================================================
Copyright (c) 2010-2014 Qualcomm Connected Experiences, Inc.
All Rights Reserved.
Confidential and Proprietary - Qualcomm Connected Experiences, Inc.
==============================================================================*/

using System;
using System.Collections.Generic;
using System.Runtime.InteropServices;
using UnityEngine;

/// <summary>
/// This behaviour associates a Virtual Button with a game object. Use the
/// functionality in ImageTargetBehaviour to create and destroy Virtual Buttons
/// at run-time.
/// </summary>
public class VirtualButtonBehaviour : VirtualButtonAbstractBehaviour
{
}
```

3. Place a plane in Unity; then In the Objects Inspector you'll now see the option to Run Virtual Button Behavior script. Essentially now that object has a behavior attached to it.

Adding a **Touch Navigation** to your AR viewer is simply downloading an Asset online and loading it into your Unity Scene.

Below are a few listed in the Asset Store:

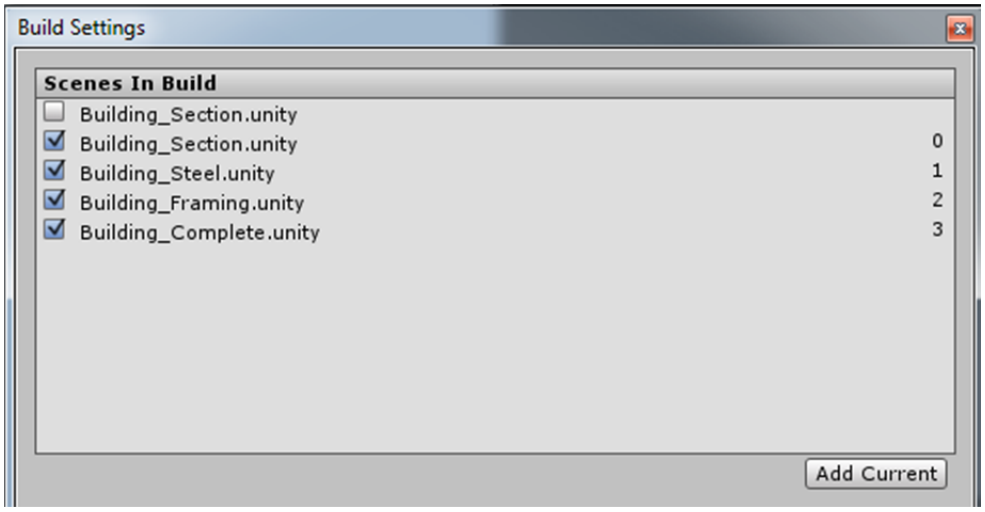


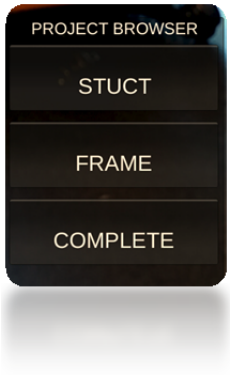
Adding a **Virtual Project Browser** which is a simple GUI menu loaded into Unity. Unity creates scenes, which are just like levels in an game. The appropriate workflow to take an construction model and show the underling layers is to separate out the model layer by layer. Creating individual scenes labeled as such; Framing, Structure, and Finish product.

Setting up a multiple Scene Project in Unity

1. Create the appropriate AR Camera and load all the materials and objects needed to showcase
2. Save the scene once all materials have been applied to the right objects
3. Save the overall model as **Building Complete**
4. Then delete out the exterior of the building and save as **Building Framing**
5. Then just like listed above remove the framing objects in scene; now your left with just the structure. Save that out as **Building Steel**

Once you've saved out the individual pieces of the project, you'll now have essentially 3 different scenes. Open one at a time and choose to Add current to Build Settings. Once each scene has been added you'll notice that Unity is adding a number to each scene, example 0,1,2,3, etc.





Creating the GUI menu(is an interactive buttons on your Mobile Device) is as simple as copying the source code listed below, obviously you'll change the naming of each button to your liking. Essentially now each button will load the scene number referenced above.

Listed Below is the source code to create the interface shown above.
For those who are not so familiar with Java, just a quick guide. GUI Box refers to the rectangle in the Viewer, the Overall length of that square is defined by 50,50,400, (<color=white> is the Text color of the buttons and JS uses standard color names but, you can also list by rgb values if needed. The next statement is the Size of Text "30" and then the label of that text.

```
/* Example level loader */

function OnGUI () {

  // Creation of Background Box
GUI.Box (Rect (50,50,400,1000), "<color=white><size=30>PROJECT
BROWSER</size></color>");

// Make the first button.
        if (GUI.Button (Rect (60,100,380,120),
"<color=white><size=40>STUCT</size></color>")) {
Application.LoadLevel (1);
  }
```

```
    // Make the second button.
            if (GUI.Button (Rect (60,240,380,120),
  "<color=white><size=40>FRAME</size></color>")) {
Application.LoadLevel (2);
     }

// Make the third button.
    if (GUI.Button (Rect (60,380,380,120),
  "<color=white><size=40>COMPLETE</size></color>")) {
Application.LoadLevel (3);
    }
  }
```

CLASS SUMMARY

A brief summary of lessons learned today are as follows. When creating an object to view in AR, be careful of how complex the object is constructed. Remember to check your polygon count in either Unity or 3Ds Max. When creating materials keep them very simple. Once you have gone step by step and figured out how to connect the dots using Revit, Unity and Vuforia you will be able to quickly create an Augmented Realty viewer on your own.

Listed Below are the links to the Screencasts associated with this learning session:

Revit to Augment

https://screencast.autodesk.com/main/details/153d69cd-06a5-4e37-a4e0-44c417639440

Revit to Unity to Mobile Device (Create AR Viewer)

https://screencast.autodesk.com/main/details/a2318796-a541-42bc-8503-e7dc05b7066d