

Creating a Space AR poster using Unity

Years 9-10



Summary

In this lesson, students explore how to design and implement a simple Augmented Reality (AR) poster experience using Unity 3D and Vuforia SDK for Unity 3D. This lesson is within the context of a Space-themed example, however, it could be used for other contexts such as Biology, Geography, Art or more.

Developer requirements

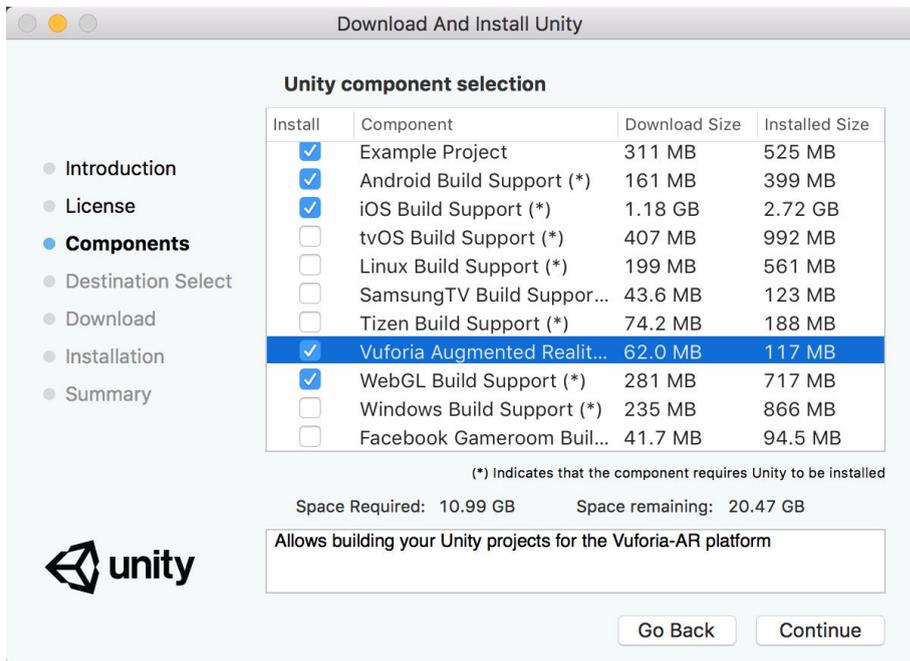
1. Unity 3D - This lesson is developed using Unity 3D (2018 4.9f1) although any new version would work
2. Free account with Unity 3d (<https://id.unity.com>)
3. Vuforia SDK (V8.5.8) for Unity 3D
4. Free account with Vuforia developer (<https://developer.vuforia.com>)

Resources

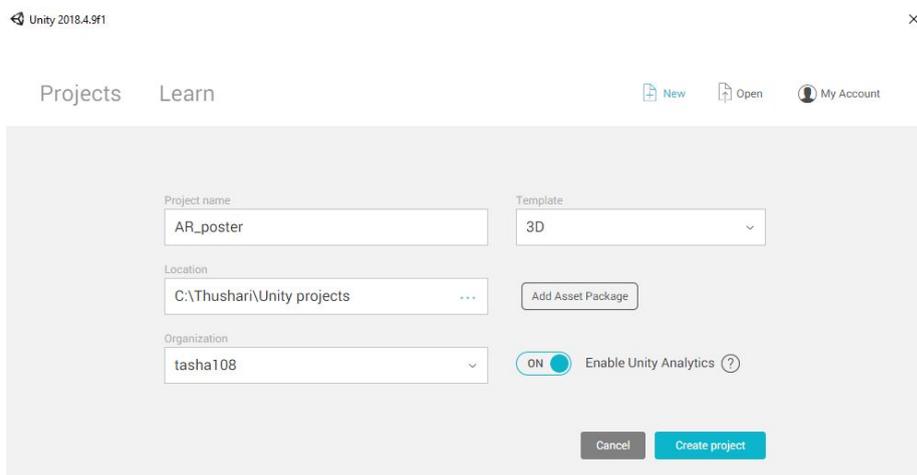
1. Printed copy of an **Image Target** (a sample is included in the 'Appendix' of this lesson). Download the image target used in this project from here - <http://bit.ly/ARSpaceAstronaut>
2. 3D object that you want to overlay image (Note: this lesson uses freely available 3D objects)
3. Computer to install Unity 3D (check 'system requirements' for development - <https://docs.unity3d.com/Manual/system-requirements.html>)
4. iPad or a computer with a webcam (in-built or wired) to test AR poster

Step 1: Unity 3D installation

1. If you haven't installed Unity in your computer, please follow the comprehensive steps listed in the Unity documentation (<https://docs.unity3d.com/Manual/GettingStartedInstallingUnity.html>)
2. You will need a free account with Unity 3D to access free Unity assets. Follow the instructions to create an account (<https://id.unity.com>)
3. You will need to add 'Vuforia AR' resources when you are installing Unity. 'Check' the **Vuforia Augmented Reality Support** during Unity installation as below. If you want to test your project using iOS or Android devices, check the corresponding components (Note: Be mindful about the **Install Size** of these components since you will need free disk space in your computer to install these). To develop this lesson, we used 'Vuforia Augmented Reality Support' and 'iOS Build Support'



- If you have completed the above steps, click **New** button to start developing this lesson



- Provide a name for the project (e.g. AR_poster), select **Template** as **3D** and choose a location from the computer to store the project files and click **Create project** (Note: it might take a while to import things needed for your project such as assets)
- Please complete steps 2-4 in order to set up the environment for your Unity-AR project
- Once these steps are completed, start 'Step 5' to start developing your AR project

Optional Steps: Familiar with Unity 3D environment

- If this is your first time working with Unity, you can try out step-by-step tutorials to get familiar with the Unity environment by downloading basic tutorials (e.g. Play & Edit Mode). These basic tutorials are recommended for beginners prior to try out our lesson since this lesson will not provide information about features in the Unity environment (such as Assets, Objects, Inspector).

Projects Learn

New Open My Account

Basic Tutorials
Tutorial Projects
Resources
Links

101 01 : Play & Edit Mode
Take your first steps inside the Unity editor, as you learn the difference between the 2 main modes in Unity - Edit mode for working on your project, and Play mode for testing. This project is for Unity 2018.3 and above. [Download](#)

02 : Game Objects & Components - Rigidbodies to the rescue
We build games in Unity using Game Objects and components. Learn about how adding a Rigidbody component to your game objects adds Physics behaviour. This project is for Unity 2018.3 and above. [Download](#)

03 : Tweaking Components - Values vs Zombies
Develop your component knowledge further as you learn about values and how they affect gameplay. Can you slow down the polygon-eating zombie in time to escape? This project is for Unity 2018.3 and above. [Download](#)

04 : Prefab power - Ramping up
Prefabs are the secret sauce behind all Unity games, in this lesson you'll learn about building them as you ramp up your knowledge and escape another laser death trap! This project is for Unity 2018.3 and above. [Download](#)

Step 2: Vuforia SDK for Unity 3D

What is Vuforia?

*Vuforia is an Augmented Reality (AR) Software Development Kit (SDK) for mobile devices that facilitates the creation of AR applications. It uses **Computer Vision** technology to recognise and track 'Image Targets' in real-time. Vuforia is also used as a Mixed Reality (MR) application development platform.*

Vuforia uses Computer Vision technology to recognise image targets and overlays AR experience (e.g. 3D objects, videos) on image targets.

If you need to learn more about Vuforia, please refer to the documentation (<https://docs.unity3d.com/Manual/vuforia-sdk-overview.html>)

License key

1. You will need a free account with Vuforia developer (<https://developer.vuforia.com>). Click **Register** and complete the required information.
2. You will also need to get a Vuforia development key to build and test your applications with Unity using **License Manager** (<https://developer.vuforia.com/vui/develop/licenses>) and **Get Development Key** function.
3. Add a random licence key name (e.g. space_vuf), 'tick' the agreement box and select **Confirm** to get a free development key.
4. Now click the licence key and copy it to your application.

Step 3: Image Target

What is Image Target (or AR markers)?

AR markers are images or objects (e.g. QR codes, physical reflective markers, Image Targets and 2D tags) registered with the application which act as information triggers in your application. When your device's camera recognizes these markers, this triggers the display of virtual content over the world position of the marker in the camera view.

Vuforia provides free samples of **Image Targets** to reuse in your projects. You can download them from here - <https://assetstore.unity.com/packages/templates/packs/vuforia-core-samples-99026>

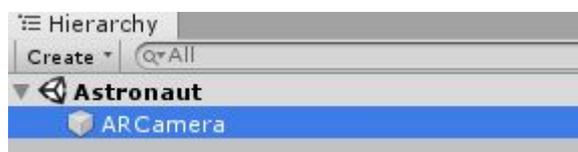
Alternatively, you can access them within the **Asset Store** tab in Unity. Use the **Search** textbox to find **Vuforia Core Samples**. You can add these samples to your asset store and **Import** them to the project. If you are unsure about which assets to import, select **All** and click **Import**.

Step 4: 3D objects

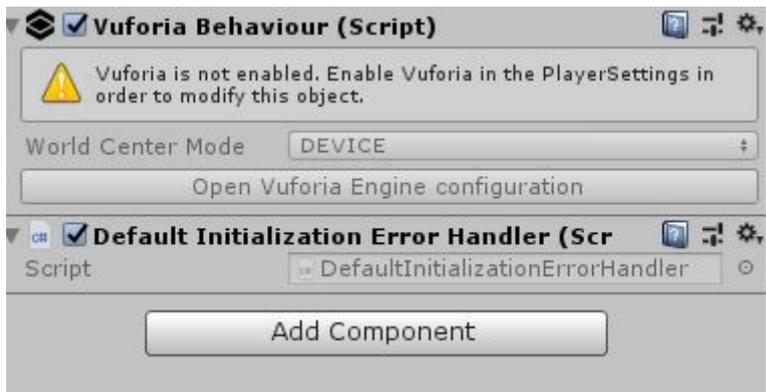
1. When our camera recognises the image target, we want to overlay a 3D object over it.
2. If you want to add your own assets, you can create 3D models on your own or download from the web (Note: 3D models that compatible with Unity can be found here - <https://docs.unity3d.com/Manual/3D-formats.html>)
3. Below are some suggested websites to find good 3D models. Feel free to share new suggestions with the CSER community (<https://teachingai.blog/>)
 - a. <https://www.assetstore.unity3d.com/>
 - b. <https://free3d.com/>
 - c. <https://grabcad.com/>

Step 5: AR development using Unity

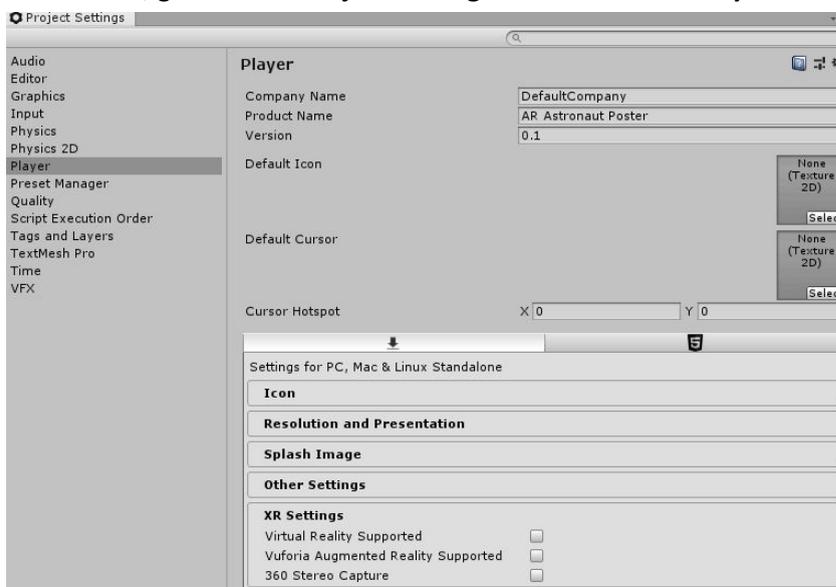
1. In order to start a new project, you will need to add a new scene by selecting **File -> New Scene** and provide a name for the scene (e.g. Astronaut). Alternatively, you can reuse the default **SampleScene** by deleting existing objects that loaded automatically.
2. In the left panel (**Hierarchy**), delete the default **Main Camera** and **Directional light** objects. To do this, right-click on each object and select **Delete**.
3. Once the existing objects are deleted, add **ARCamera** to the project by selecting **GameObject > Vuforia Engine > AR Camera** (if you are prompted for anything, you will have to authorise them all - e.g. importing additional assets).



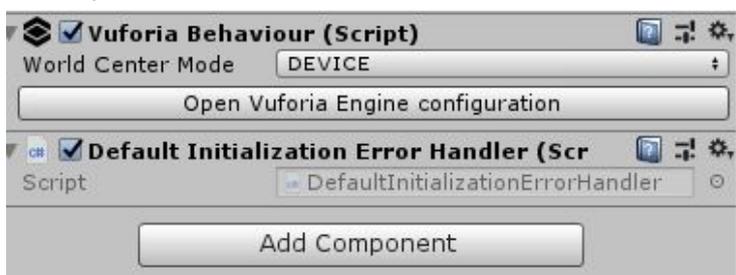
4. You should see the **AR Camera** object added to the **Hierarchy** panel. Once you select AR Camera, you should see the **Inspector** panel on the right hand side. This demonstrates the configurations of the object.
5. Towards the bottom of the 'Inspector' section, you should see a button **Open Vuforia Engine Configuration** that is disabled by default.



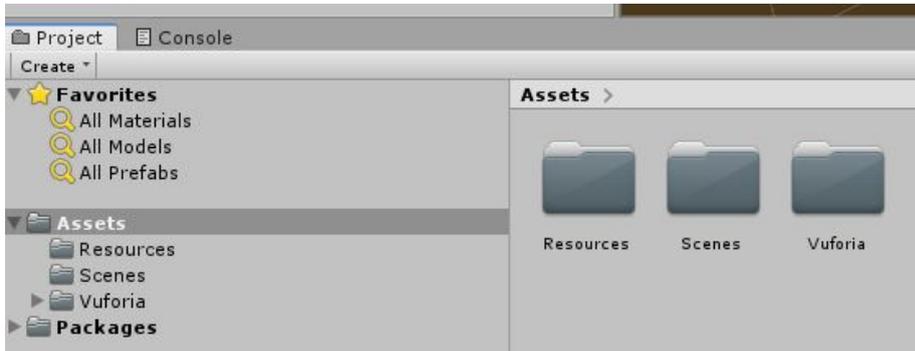
- To enable it, go to **Edit > Project Settings** and click on the **Player** button.



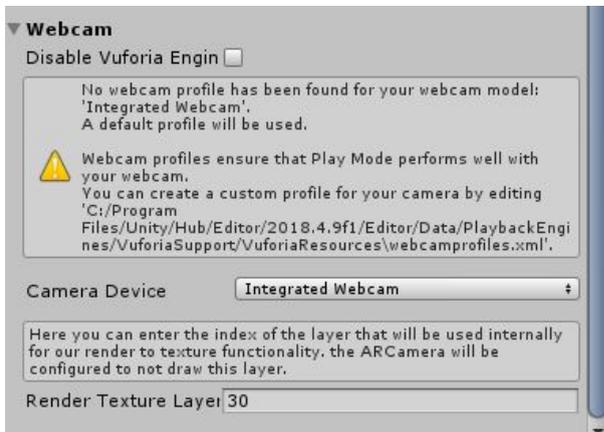
- In the **XR Settings** section (bottom), select the option **Vuforia Augmented Reality Supported** (Click 'accept' if you were prompted for Vuforia Software Licence).
- You can then close the **Project Settings** window and go back to the AR Camera's configurations window.
- The **Open Vuforia Configurations** button should be enabled by now (click 'AR camera' to see the 'Inspector' window).



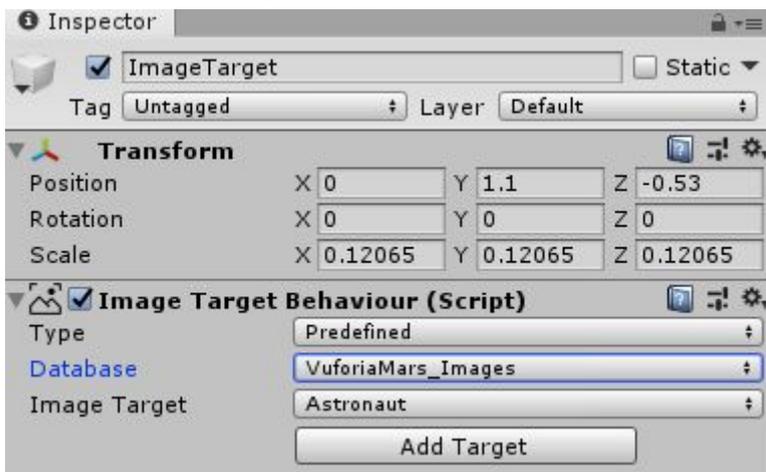
- You will need to have a webcam (either built-in or wired) configured for this project (Note: if you do not have a webcam, alternative steps for testing are listed at the bottom of this lesson). If you have only one camera associated with your PC or laptop, everything should be working fine with the default settings. To confirm this, click **Resources** in the **Assets** tab and select **VuforiaConfiguration.asset**.



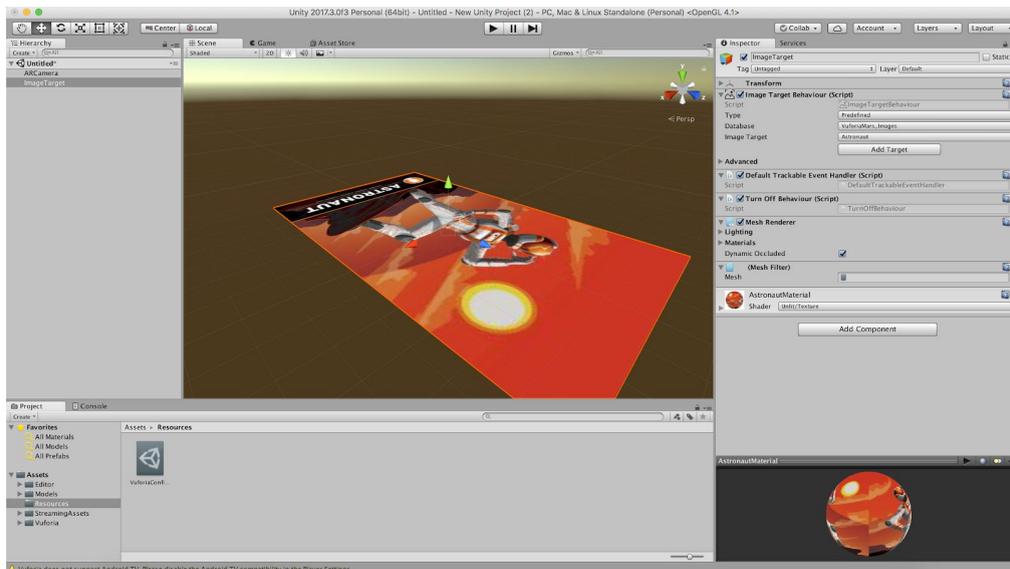
11. In the **Inspector** window, check **Webcam** settings towards bottom. In the **Camera Device**, select the webcam you want to use for this project. Otherwise, leave **Integrated Webcam** as below.



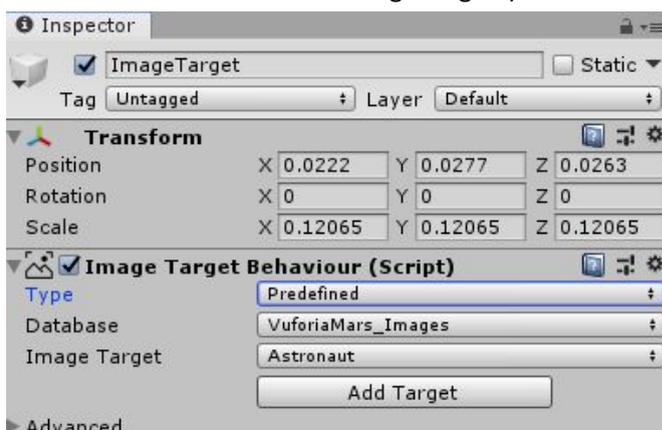
12. To add an image target to our project, go to **GameObject > Vuforia > Image**. This will add the image target to your scene
13. In the **Image Target Behaviour** component, click on the **Database** drop-down list and select your Target database (e.g. VuforiaMars_Images). In the **Image Target** drop-down list, select the name of your Image Target (e.g. Astronaut) from the database.



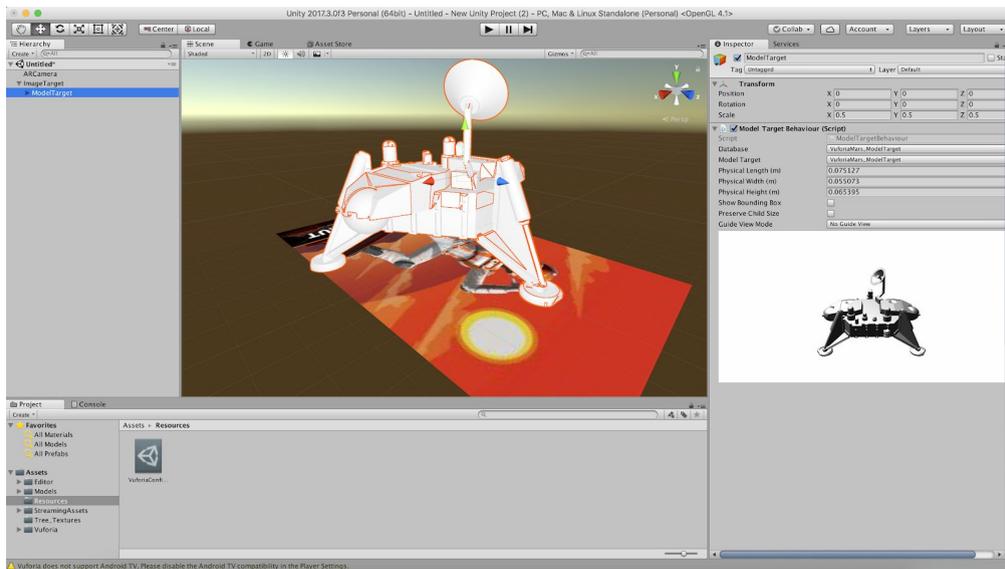
Note: You may have to zoom in a little bit inside your scene to see the image.



14. The default image comes from the Vuforia core samples we imported earlier. It can be replaced later with your custom image, but for now, we want to check that our setup is working.
15. Therefore, we use **predefined** as the **Type** of **Image Target Behaviour**. You can also define custom image targets by changing the **Type** to **User Defined** (See optional steps at the end of this lesson to add custom image targets).



16. Now we will need to add a 3D object to the Image Target. For this, right-click on the **ImageTarget** in the left panel and select **Vuforia > ModelTarget**.
17. We have selected **VuforiaMars_ModelTarget** as the **Database** and the **Model Target** in this project.
18. By default, the 3D model is quite big. Therefore, we need to scale it down. Click on the **ModelTarget** object in the left panel and update its scale in the right panel. A scale X, Y and Z of 0.5 should work well. After scaling it down, it should look something like below.



- At this stage, we should test that our setup is working properly before starting to add our own assets. In the **Hierarchy** window, make sure the **Model Target** is placed under the Image Target (Note: You might want to change the position of the model, change the values of the Scale / Rotation (inside the inspector) to get it into the right place)

Step 6: Testing the project

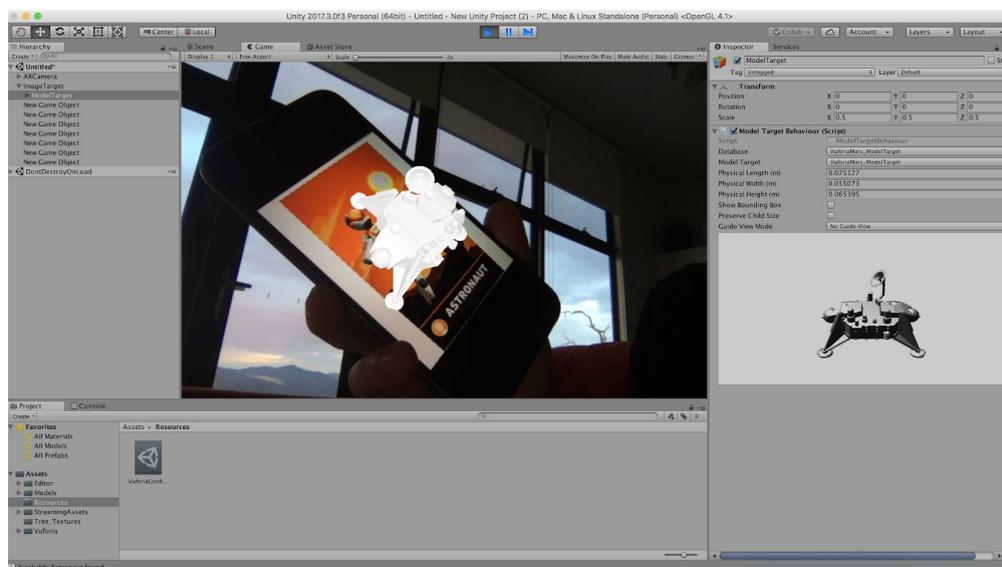
Option 1: Testing using the computer

- To test your project with a computer, you will need a built-in or wired web camera. If you do not have access to these, please try out 'option 2: Testing using the iPad' below.
- Now you can try out the AR experience by clicking the **Play** button of Unity and pointing the printed poster (or a mobile device loaded with the image target) to the camera.

We have included the **Image Target** used in this lesson at the end of this lesson (Appendix). Alternatively, you can download and print them on your own using this link (https://library.vuforia.com/content/dam/vuforia-library/docs/target_samples/unity/mars_target_images.pdf)

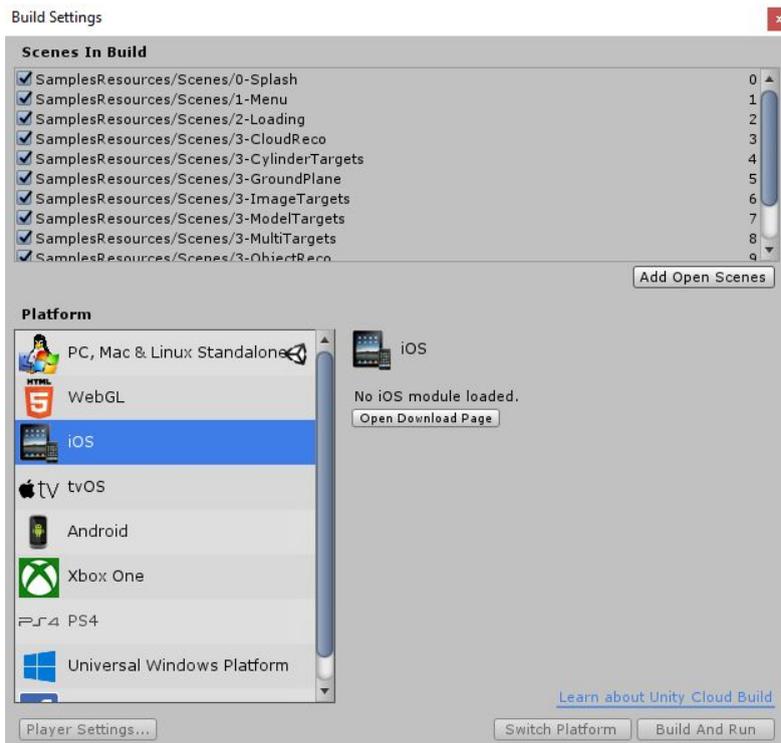


3. If everything is working properly, a 3D model should appear when you point the image target to the computer's camera. To stop running the project, hold and press the **Play** button again.

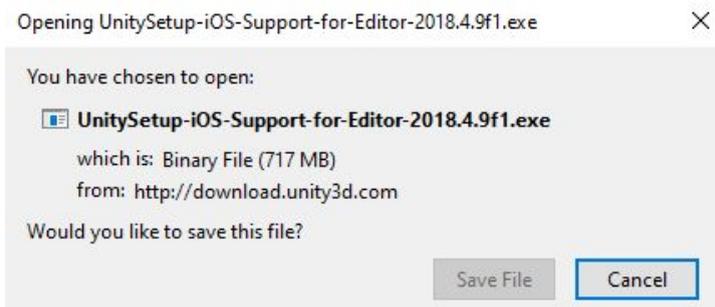


Option 2: Testing using the iPad

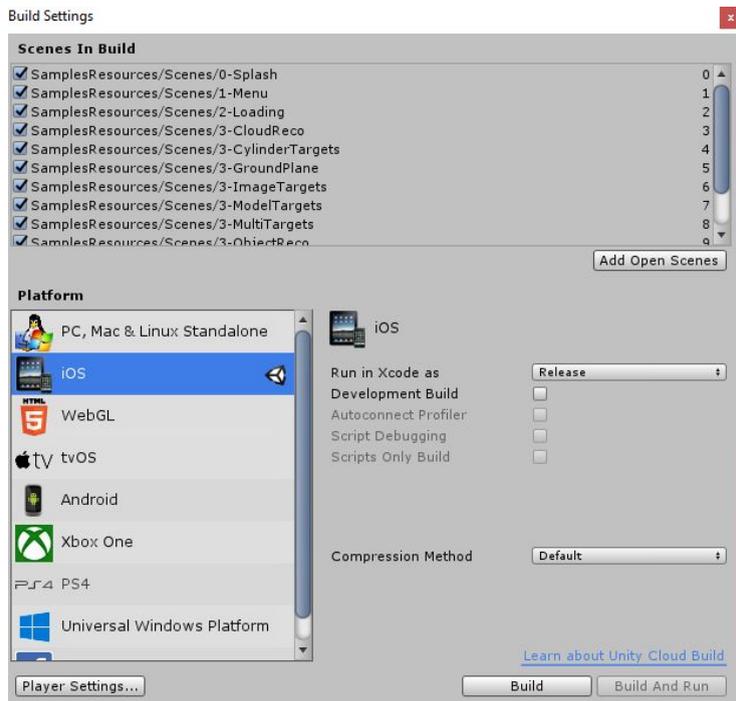
1. To test your project with an iPad, you will need an iPad with 'Unity Remote 5' app loaded. If you do not have access to these, please try out 'option 1: Testing using the computer' above.
2. To test the project with iPad, we will have to switch platform to iOS. Select **File -> Build settings** and then select **iOS** as the platform.



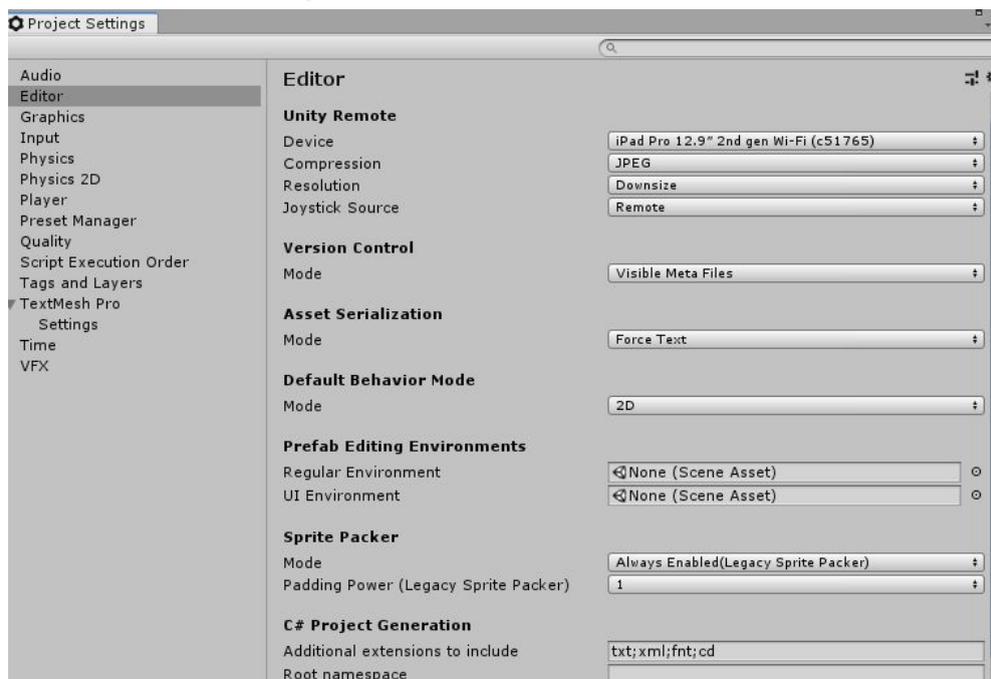
3. If you do not have iOS modules loaded (see pic above), click **Open Download Page** and you will be prompted to **Save** iOS support file. Download this file to your local computer and double click to install it (Note: this installation required ~2 GB of your disk space).



4. Once you have installed **iOS support**, make sure you restart Unity. Now go back to **File -> Build setting -> iOS** (step 2 above) and click **switch platform** at the bottom and wait until switching platform done.
5. If switching of platform is successful, you will see a window like below in **Build settings**.



6. Connect your iPad to the computer using a USB cable (provided for iPad charging) and open **Unity Remote 5** app. Once it is opened, click the **Play** button in the Unity window.
7. Click **Edit -> Project Settings -> Editor** and select the **Device** that is connected to Unity. You will be prompted to select the iPad you are using (Note: iPad name should differ from what is shown in the image below)

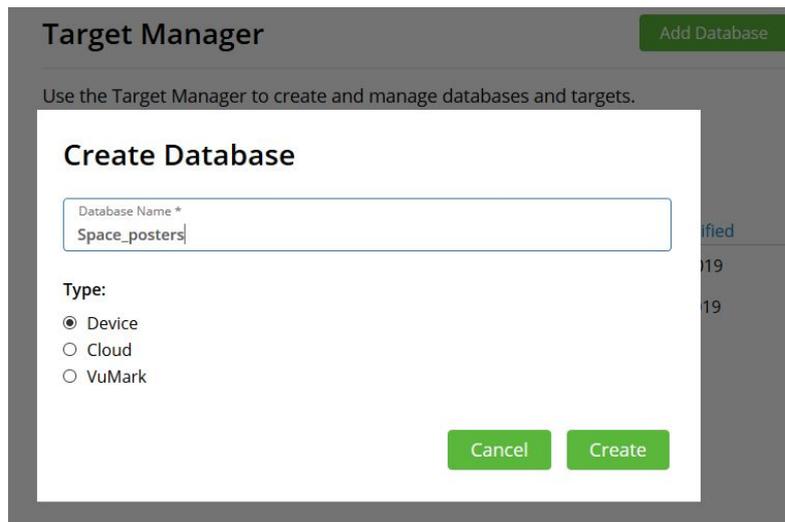


8. You will see the Unity play screen duplicated in your iPad. Now point the printed poster in front of your iPad camera and enjoy the AR experience with 3D models. Enjoy!

Optional Steps: Adding a custom Image Target to your project

1. To learn more about Image Targets, go to this page - <https://library.vuforia.com/articles/Training/Image-Target-Guide>

- In the Vuforia Developer Portal (<https://developer.vuforia.com/vui/develop/databases>), select **Target manager**.
- Click **Add database**, provide a name, select type as **Device** and click **Create**.



- Now select the database we have created (e.g. Space_posters) and click **Add Target**.

Add Target

Type:



Single Image



Cuboid



Cylinder



3D Object

File:

Choose File

.jpg or .png (max file 2mb)

Width:

Enter the width of your target in scene units. The size of the target should be on the same scale as your augmented virtual content. Vuforia uses meters as the default unit scale. The target's height will be calculated when you upload your image.

Name:

Name must be unique to a database. When a target is detected in your application, this will be reported in the API.

There should be four different target types. Select **Single Image** and browse your disk to locate the image you want to use as an **Image Target**

- The **Width** value (e.g. 5.5 in cm) is a scale value that you need to set to the size you want the image to appear in your Unity **Scene** (Note: Unity measures everything in your **Scene** in relation to the size of your **Image Target**).
- Give a name to the target and upload it
- Now the uploaded image appears in the list of targets with a rating. If the image you uploaded has low ratings, it may be difficult for the camera to track it (see more information [here](https://library.vuforia.com/articles/Solution/Optimizing-Target-Detection-and-Tracking-Stability) - <https://library.vuforia.com/articles/Solution/Optimizing-Target-Detection-and-Tracking-Stability>). In such cases, try to upload a different image.
- Once you are satisfied with your image's rating, check the box next to **Target Name** and click **Download Database** button.

9. Under the development platform, select **Unity Editor** and click **Download** and save this file (.unitypackage) in your disk.
10. Switch back to your Unity project to use the Unity package in your AR application.
11. Now, we need to import **Image Target** that we have saved. In the **Project** window, right click **Assets** and select **Import Package > Custom Package** and select the package saved in the disk before.
12. Add you **App Licence Key** saved from **Vuforia Licence Manager**.
13. Once you have set the image target, next step is to add image target to the scene by selecting **GameObject > Vuforia > Image** and follow the instructions listed in **Step 5: AR development using Unity**.

Discussion

The following are some suggested discussion points to create the augmented world with astronaut poster. You can add answers to following questions as audios, new 3D models, videos, or new Image Targets.

- How many astronauts can live on the International Space Station at once?
- What was the first mission to successfully land on the Moon? Who involved? When was it?
- Which NASA space shuttle exploded 73 seconds into its flight? What was the reason? And which astronauts on board sacrificed their lives?
- Why 2019 October is important for female astronauts? What it means by 'All-Female SpaceWalk'? Who involved? (See more information here - <https://www.space.com/first-all-female-spacewalk-date-moved-up.html>)
- How does the computer/device recognise the Target image?

Why is this relevant?

An astronaut is a person trained by a human spaceflight program to command, pilot, or serve as a crew member of a spacecraft. Before learning about the life of an astronaut it is important to know about the environment where astronauts work. Space is located approximately 76 miles above the earth's surface. Because it is above the atmosphere, there is no air - more specifically there is no oxygen. Space is a dangerous place to travel, filled with unique challenges for working and living there. And yet, it somehow draws astronauts to go there and explore this vast unknown.

In this activity, students are engaged in evaluating the AR posters and considering how they could build their own AR/VR experience about astronauts. In doing so they are thinking about the digital content and ways of presenting information in a 3D space.

Assessment

[We have suggested Assessments, however, you may choose to use your own or for further advice, examples and support around assessment please visit the Digital Technologies Hub at digitaltechnologieshub.edu.au/teachers/assessment]

A group activity to add more information to the AR space poster as audio and/or 3D text about astronauts and spacecrafts would be an excellent chance for students to present their AR project to the class, along with addressing:

- Project title - introducing their favorite astronaut and spacecraft
- Project description (Science) - what Science concepts (e.g. gravity of planets, what materials used to build space crafts) are coupled with this project
- Project description (Digital Technologies) - what Digital Technologies concepts (e.g. digital systems, information systems, Internet) are coupled with this project

- Project description (Artificial Intelligence) - what Artificial Intelligence concepts (e.g. computer vision to track target image) are coupled with this project. This is an optional activity for students familiar with AI technology. For more information, please visit our free CSER MOOCS on teaching AI in Secondary years (https://csermoocs.appspot.com/ai_secondary)
- Project description (Augmented Reality) - what AR concepts (e.g. image target, 3D objects) are coupled with this project
- Process of design and development of the project
- What is the experience of developing this AR project, what are the identified challenges and how they were resolved

Curriculum links

Links with the Digital Technologies curriculum area

Year band	Strand	Content description
Years 9-10	Processes and Production Skills	Create interactive solutions for sharing ideas and information online, taking into account safety, social contexts and legal responsibilities (ACTDIP043) Plan and manage projects using an iterative and collaborative approach, identifying risks and considering safety and sustainability (ACTDIP044)

ADD Links with other curriculum areas

Year band	Learning area	Content description
Year 9	Science - Science Inquiry Skills	Communicate scientific ideas and information for a particular purpose, including constructing evidence-based arguments and using appropriate scientific language, conventions and representations (ACSIS174)
Year 10	Science - Science Inquiry Skills	Communicate scientific ideas and information for a particular purpose, including constructing evidence-based arguments and using appropriate scientific language, conventions and representations (ACSIS208)

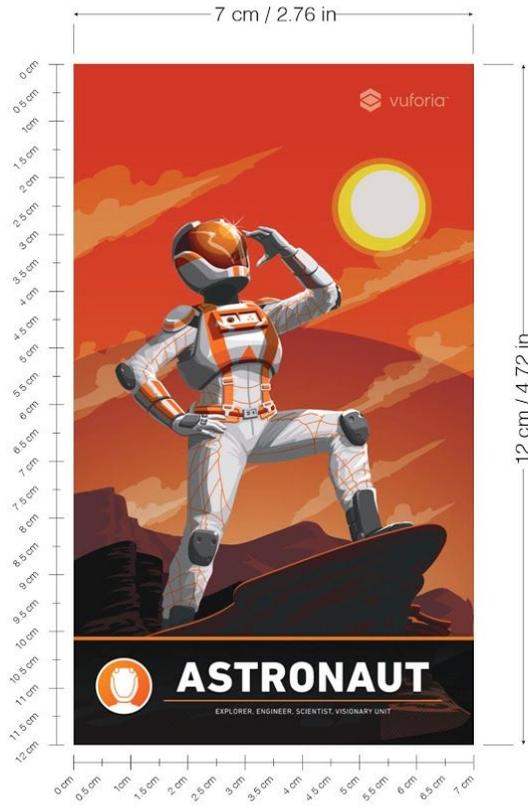


Author: Thushari Atapattu

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/). Computer Science Education Research (CSER) Group, The University of Adelaide.

Appendix

For best results, please print without scaling.



©2017 - 2018 PTC Inc. All rights reserved.