Group Assignment 0

Deadline: **February 11th, 2021**

In this assignment you will set up the basis for all future group assignments. You will create a basic Vuforia project and with a simple AR application.
Your work will be assessed during the next office hours. However, **this assignment will not count towards your grade.** The purpose of this trial-assessment is to give you an idea of how points are awarded in the group assessments.

PREPARATION

- Get an external webcam: we will provide each of you with a webcam, which you can pick up at AU. Please refer to further instructions on Blackboard.
- Download and install Unity. We are using version 2019.4 (LTS) Personal. If you have another version installed, you must install this specific version in addition. Managing multiple versions is well supported with the Unity Hub.
- Add Vuforia to your project, register as a Vuforia developer, and setup your project with the generated license.
- If you are using Unity or Vuforia for the first time, we recommend you to do our basic AR tutorial available on blackboard, before starting to work on this assignment. There you will learn how to set up a basic AR scene using the built-in image targets (e.g., Astronaut and Drone).
- Use a version control system to collaboratively work on the assignments. We recommend using Git with a dedicated gitignore for Unity projects, such as this one.

Basic augmented reality in Unity with Vuforia

Tasks

a) **1pt** Create a planetary system with Sun, Earth, and Moon objects with Spheres. Use the Unity GameObject hierarchy to make Earth a child of Sun, and Moon a child of Earth.

b) **1pt** Add one image target for positioning the planetary system and attach your hierarchy of planets to it. Do not use the example image targets from Vuforia. Instead, we will provide you with some image targets that you must upload to the database in your Vuforia Developer account.
Make the image target the center of your scene's coordinate system (see *Vuforia Behaviour* in your `ARCamera`).

> Explain: What do the different options for *World Center Mode* mean?

c) **1pt** Make the earth rotate around the sun using `transform.Rotate` in a script. Then make the moon rotate around the earth with the same technique.

d) **2pt** Add another image target to position a meteor, and create a the corresponding Sphere Game Object. (Remember to increase the number of simultaneously tracked image targets in settings.) Print to the console whenever the earth and meteor collide (i.e., the spheres touch). Use two different approaches for collision detection: 1) use Unity Colliders (i.e., Sphere Collider) to check for collisions and 2) simple distance calculations. For the latter, you will need to use the `position` and `lossyScale` of each object's `transform`.

> Explain: Why is it important to have correct measures for your image targets when uploading them to Vuforia?

e) **1pt** Make the scene look more realistic: Apply textures to each of the planetary objects by creating and assigning dedicated materials. Turn off the default directional light and create a point light at the position of the sun, to realistically illuminate the earth and moon. Adjust the emission and albedo of the sun's material so that it is not influenced by the point light itself.

> Explain: What is the difference between *albedo* and *emission*?

f) **BONUS 1pt** Use a particle system to create an explosion at the collision point, whenever the earth and the meteor collide. In addition, render a trail behind the meteor (e.g., using a second particule system) when it moves (i.e., you move its image target).

Min / Total = **3.0pt / 7.0pt**