Sheffield Hallam University

College of Business, Technology and Engineering

Department of Computing

Project (Technical Computing)

[55-604708]

2020/21

Author:	Callum Peter McLean
Student ID:	28015292
Year Submitted:	2021
Supervisor:	Penny Collier
Second Marker:	Martin Cooper
Degree Course:	Computer Science for Games
Title of Project:	A prototype AR app to enhance learning of A-Level Biology

Confidentiality Required?
No

I give permission to make my project report, video and deliverable accessible to staff and students on the Project (Technical Computing) module at Sheffield Hallam University.
YES

Synopsis

Augmented Reality has been gaining ground over the past decade; however, it has not been applied into mainstream education. Throughout this project, I researched and developed a mobile app which uses augmented reality that can be used to assist in the understanding of factors which impact on plant growth that are pertinent to the A-Level Biology Syllabus. Participants tested the app and gave feedback as to whether they thought it would be a viable option for education.

I concluded augmented reality could be effectively used in education.

Acknowledgement

I would like to thank all my friends and family, especially my parents, for supporting me throughout University. I also wish to thank Penny Collider for all her assistance and supervision throughout this dissertation.

1 Introduction

1.1 Project Background and Motivations

As technology advances, people are finding more ways to digitise education. Online teaching through the use of services such as Zoom is a prime example of this. Augmented Reality (AR) has not yet been adapted into any mainstream apps focused on the aiding of learning. The current project aims to determine whether AR could be used to help educate. The reason for choosing AR instead of other technologies, such as virtual reality, is accessibility: it can be run on the majority of smartphones, which most people have easy access to; and it is easy to use. Another reason AR could be a suitable tool for education is that it can easily portray 3-dimensional objects and ideas, allowing the user to manipulate them and view them from all angles.
Biology was chosen as the subject because it focuses on aspects of the physical, macroscopic world around us which can be immediately visualised. Subjects such as chemistry and mathematics are more theoretical and model-driven and are less easy to visualise. Some of the concepts that biology focuses on, such as the functioning of organs, growth of plants and cells, are ideal for AR as it may allow the user to see the development of organic structures at different spatial and temporal scales, which would otherwise be difficult to demonstrate.

1.2 Aims and Objectives

This project aims to create an AR mobile app which can be used as an aid in the demonstration of a topic in A-Level biology. Firstly, the different implementations of AR in mobile devices will be researched in order to see which apps of AR have been effective. The current A-Level Biology syllabus will be studied to find a suitable topic. Thirdly, the app will be created to a standard that a student could use and test. Fourthly, the effectiveness of the app will be determined by participants testing it and providing their feedback. Finally their feedback will be used to decide whether AR apps can be used effectively in education.

2 Research

The first section of the project involved deciding three things: which AR software to use for the app; which A-Level biology subject to dedicate the app to; and what tools and services to use throughout the project.

2.1 AR software

The most popular AR softwares have been found and listed (Table 2.1). The key requirements for a suitable software are:

- The software must be compatible with Android devices. These are available for testing in this project.
- Since the development of this project is going to take more than a month, a trial of any software would not suffice, so the software would have to be open-access and entirely free to use.
- Compatibility with the Unity engine is useful because it is a very easy engine to work with, and it allows rapid building onto mobile devices.
- Device and plane tracking are essential features for the development of the app. These are core mechanics in AR. While the availability of anchors, image tracking and light estimation would allow for more features to be implemented and so allow for a higher end-quality of the app, they are not necessary.

Toolkit	Platforms	Type	Support Unity	Free	Device tracking	Plane tracking	Anchors	2D image tracking	Light estimation
ARCore	And, IOS	FW	Yes	Yes	Yes	Yes	Yes	Yes	Yes
ARKit	IOS	FW	Yes	Yes	Yes	Yes	Yes	Yes	Yes
ARToolkitX	And, IOS, Wind	SDK	No	Yes	-	-	-	-	-
MARS	And, IOS, Wind	SDK	Yes	No	-	-	-	-	-
MaxST	And, IOS, Wind	SDK	Yes	No	-	-	-	-	-
Vuforia	And, IOS, Wind	SDK	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Wikitude	And, IOS, Wind	SDK	Yes	No	-	-	-	-	-

Table 2.1: Comparison of various features in different AR software.
And - Android. IOS - iPhone Operating System. Wind - Windows.
FW - Framework. SDK - Software Development Kit

Analysis of this data led to four potential softwares being considered: ARKit, ARCore,EasyAR and Vuforia. ARKit is a framework for iOS devices and so was considered incompatible. ARCore is a framework and not a software development kit (sdk) so additional work would be required in setting up a development environment and delivering to a mobile device. EasyAR seems suitable but does not include lighting estimation. Vuforia can be used to create AR apps for all mobile platforms and has all the needed and optional features. It can also be used within Unity, providing a good workspace which allows easy deployment apps to a mobile device. It also has a collection of very comprehensive guides for set up with Unity. Consequently Vuforia was selected as the most suitable toolkit.

2.2 Biology Subject

All of the subjects in A-Level biology were found in the AQA syllabus (Morris, 2016) and compiled in Table 2.1. Consideration of the subjects involved the following questions:

- Can it be displayed in 3D easily and effectively? This is essential because 2D visuals would render the use of AR irrelevant.
- Can the visuals be kept simplistic yet recognisable? Making complex models and textures would take extra time and effort. The graphical representation and artwork of the app is not the main focus of this project.

- Can the visuals be made interactive? This is important because it helps to engage the user. Interacting with an object can help someone to better understand it (Shabiralyani, 2015).

Subject	Visuals	Can the visuals be represented well in 3D?	Could the models be simplistic yet still recognisable?	Can the visuals be made interactive?
Biological Molecules	Elements, molecules	Yes	Yes	No
Cells	Cells, reactions	Yes	No	Yes
Organism exchange	Microscopic organisms	Yes	No	No
Genetic information	Molecules, bases, DNA	Yes	No	Yes
Energy transfer	Cells, Plants, Lungs, nutrient life cycle	Yes	Yes	Yes
Organism responds	Veins, skeleton, human body	Yes	No	No
Genetics and population	Population diagrams	No	No	Yes
Control of gene expression	DNA	No	No	Yes

Table 2.2.1: Biology subjects and their necessary factors

Only two subjects would have visuals which could be easily represented with 3D models: energy transfer (plants and the ecosystem); and organism response (veins, muscles and skeletons). The problem with the latter is that in order for the models to be recognisable, they would have to be very detailed and complicated. It would also be easier to make an app based around energy transfer more interactable, which is important for an educational app.

Energy Transfer has been chosen as the subject the app will be based around.

To learn about the subject of energy transfer, section 3.5 of the syllabus (Morris, J. 2016) was read through as well as the topic 5A - Photosynthesis and Respiration (AQA, 2016).

The subsections of energy transfer are: Photosynthesis, respiration, energy and ecosystems, and nutrient cycles. Respiration and Energy and ecosystems focuses on animals instead of plants so this section was not included in this project due to the artwork being too complex. Photosynthesis is the process by which plants generate energy, so this can be demonstrated by simulating plant growth and allowing the user to control the rate of photosynthesis. Nutrient cycles explains how plants pull nutrients from the soil to grow, then when they die, microorganisms decompose their mass back into the soil, creating a cycle. This could be shown using a small population of plants, and allowing the user to move a time slider back and forth to watch the plants die off and grow back, while having a display of the current nutrients levels.

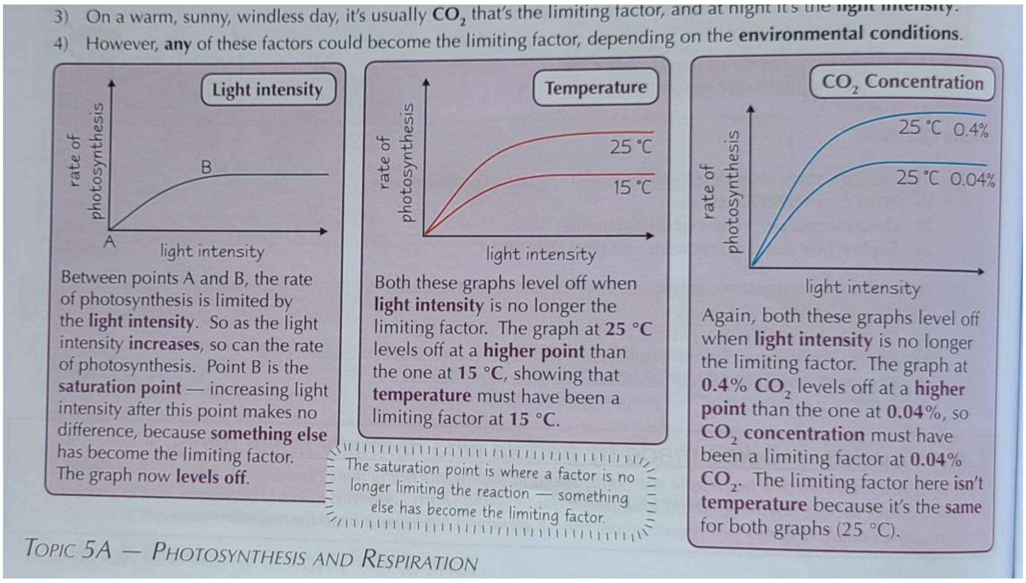

Figure 2.2.2: Graphics from the biology textbook showing limiting factors (Morris, 2016)

In the textbook (AQA, 2016) limiting factors of photosynthesis are taught with 2D graphs as seen in Figure 2.2.2. The proposed app could improve upon them by including similar graphs and also having an interactive element. The user can influence the plant growth by altering the light, water and co_2 levels.

2.3 Other AR Educational Apps

The next step was to look into other, existing educational apps which implement AR. After finding and installing some of the most popular apps, they were assessed for performance.

2.3.1 Civilisations AR

Civilisations AR (2020) is an app developed by the BBC which allows users to view various historical artifacts. It teaches users about these artifacts by playing recorded audio about them, also allowing the user to view them from different angles and see their actual scale. A special feature allows the user to see inside of the augmented objects, almost like an x-ray. All the user interfaces (UIs) are very minimalistic, and any text has a transparent background. This gives the user an extended view of the augmented surroundings and the artifacts.

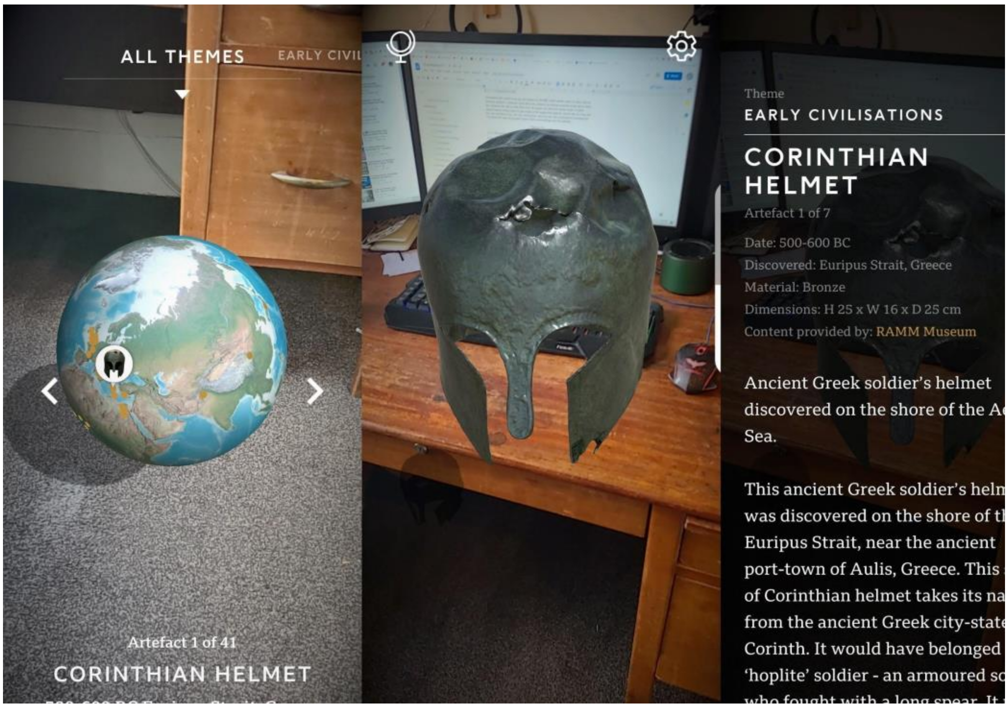

Figure 2.3.1: Screenshots of Civilisations AR

2.3.2 Edmentum AR Biology

Edmentum AR Biology (2020) is an app aimed towards teaching biology. It has three sections: organism classification; frog dissection; and respiration. In the first section the user picks from a selection of animals and can then see them in AR. There is a very enjoyable interactive feature which allows the user to control their chosen animal by moving them around with a joystick. Lots of the UI has solid backgrounds which reduces how much of the video stream the user can see. The worst example of this is the title bar along the top (see figure 2.3.2).

Figure 2.3.2: Screenshots of Edmentum AR Biology

2.3.3 AR Skeleton

AR Skeleton shows a realistic skeleton using AR. It is an example of poor UI design, with solid backgrounds taking up around 30% of the screen (Figure 2.3.3). Even while taking up so much space, all of the text still does not fit, so that the user has to scroll in order to read it all. Aside from that the artwork is good with decent functionality. The user can open a menu and select a body part which is then highlighted and information about it displayed.

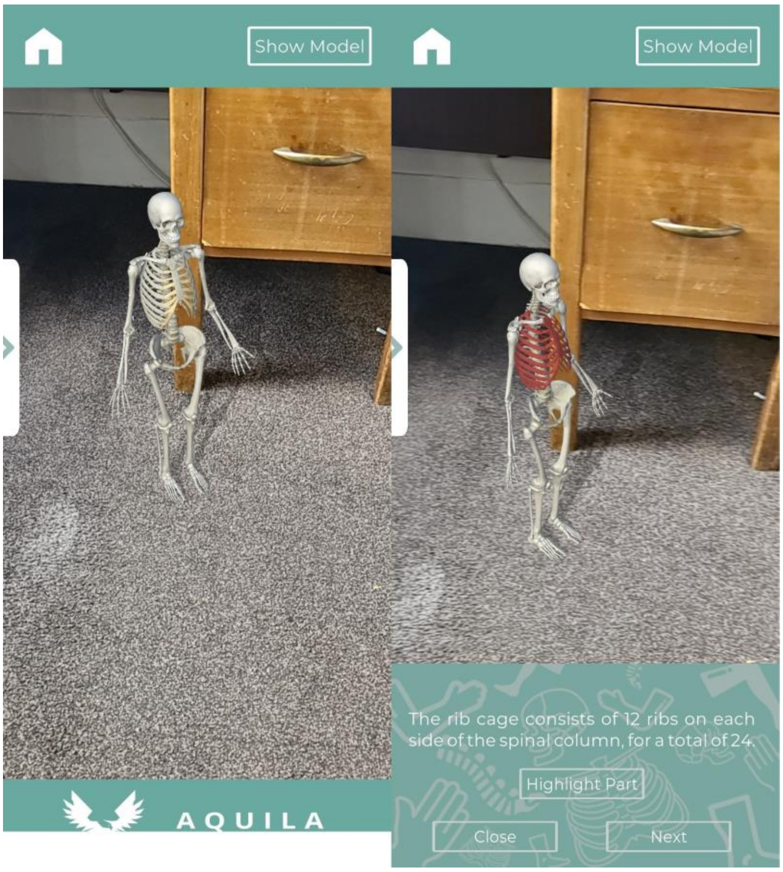

Figure 2.3.3: Screenshots of AR Skeleton

2.3.4 Plant A Tree

Plant A Tree is a very simple AR app which allows the user to choose from a selection of plants to be placed in the augmented space (Figure 2.3.4). The user can then take a photo of what is on their screen. Because of the limited functionality there is little UI, but the app is very intuitive.

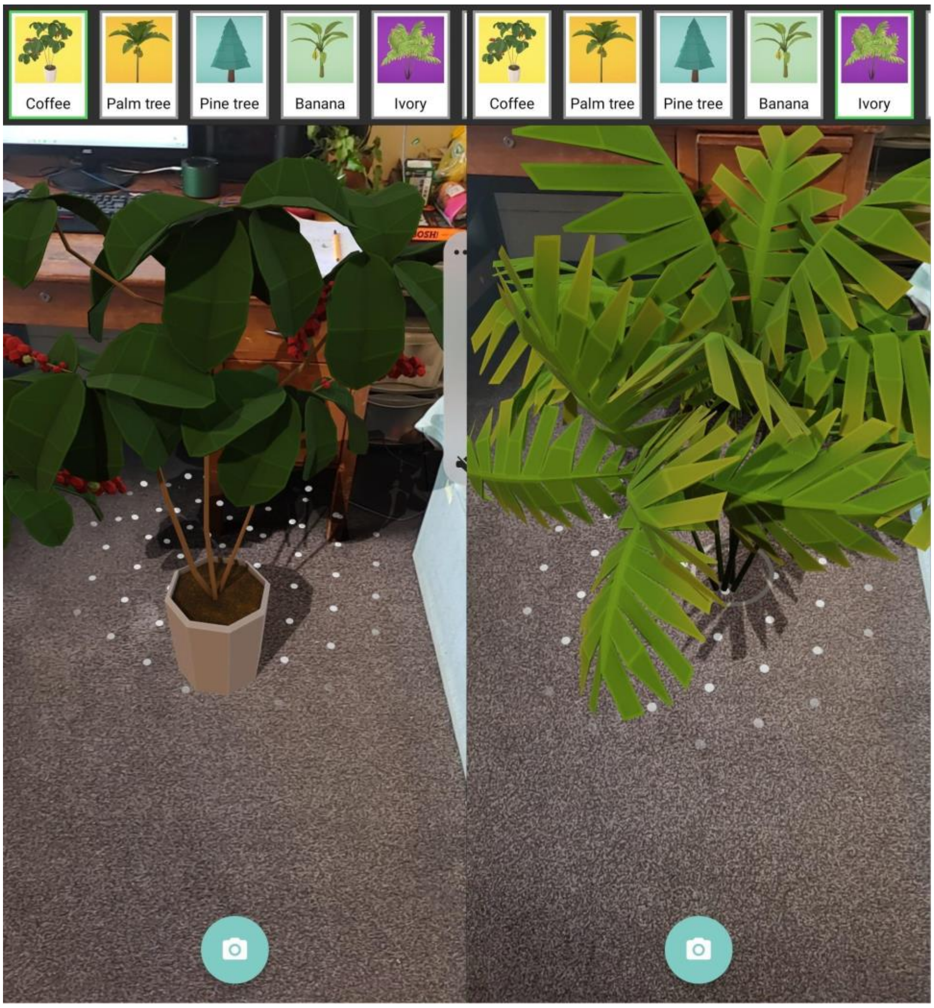

Figure 2.3.4: Screenshots of Plant A Tree

2.3.5 AR Solar System

AR Solar System has two features. The first shows our solar system, with each planet orbiting around the sun (Figure 2.3.5). The second feature allows the user to cycle through each planet and see information about it. The UI in the app is very good, with buttons that are simple, small icons, and text that has transparent backgrounds which fit in with the theme.

Figure 2.3.5: Screenshot of AR Solar System

2.3.6 Conclusions

This assessment shows that the nature of the user interface (UI) is central to the quality of all of the apps. The best UIs are minimalistic and are often slightly transparent where they could be, allowing the user to see as much of the camera stream as possible. However, a simplistic UI is only good if it is intuitive for the user to use, or if an accessible tutorial is included.

Most apps allow the user to shrink or expand models to a desired size. But some limit this size making it difficult to view certain models. If the models are made too big, they would clip out of view.

Some apps only function after the user has printed off a set of AR targets. This limits the audience as well adding more difficulty in using the app. Not everyone has access to printers, or wants to print out extra material. Ideally all of the material needed to run the app should be contained within the app.

Some apps would ask for permission to access the device’s files, which was not necessary for the app to function.

Civilisations AR (2020) and Edmentum AR Biology (2020) both have features which make them more interesting and enjoyable to use, which, for an educational app, is very important.

2.4 Mockups

Three different sections of the chosen subject of Energy Transfer have been decided upon:

- Reactions & Transport - the basic chemical reactions which occur in plants, and how materials are transported around the plant;
- Response - how a plant would grow in accordance with changes in its environment;
- Population - showing that the kind of plants that grow in an area are determined by their surrounding

The first stage in developing this was to sketch each stage of the app. This was to visualise how the UI would look and showcase the basic functionality of each section.

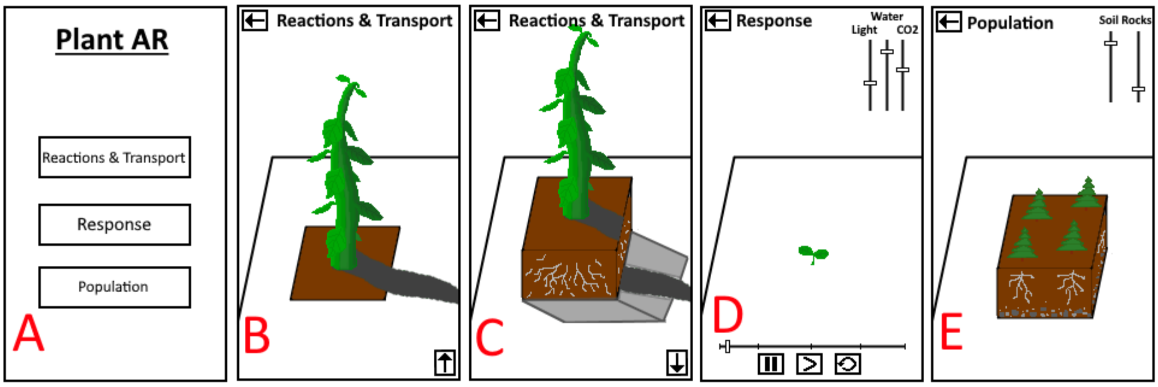

Figure 2.4: Initial sketch of the app

This is summarised in Figure 2.4. Sketch A shows a simple home screen with buttons to access each section of the app. The second sketch, B, shows the Reactions & Transport section. Here the user will be able to click on certain parts of the plant, such as the stem and leaves, to display information about them. This will include information such as what cells are there and what reactions occur within them. By clicking the button in the bottom right, a cube of earth will be exhumed to show the roots, as shown in sketch C.
The fourth sketch, D, shows the Response Section. Here the user will place a seedling, a water source and a light source. The user can alter the conditions in which the plant grows by altering the concentrations of CO_2, water and light. Upon hitting the play button, the user can watch the plant grow under the environmental conditions that they have set. This allows the user to see how the different variables affect plant growth. The variables

can be altered during the growth experiment to see the direct effect of changing each component.
In the final sketch, E, the Population section is shown. Here a cube of earth is visible with some sliders to the side. The sliders are used to change characteristics of the substrate in which the plant grows, e.g.soil type (clay, sand or silt) and soil depth. These factors will affect how plants will grow, demonstrating the importance of soil types and depth of soil.

It is intended that all of the UI will be minimalistic and will take up a small amount of the screen, as well as being positioned around the edge of the screen. Any text will have a slightly transparent background. This will give the user the biggest possible view of their surroundings and the augmented objects.

A common feature of AR apps is a requirement that the user prints off or buys a set of markers. These markers have an image which the app recognises and uses as part of the AR experience. The problem with using these markers is that it relies on the user having access to a printer, or for the app owner to send them a physical copy. The app is not self-contained on the user’s device. Markers will not be used in the development of this app.

2.5 Technical Plan

The Mockups show what mechanics are needed for the app. The next step is to create technical designs and a formal plan.
Creating UIs in Unity is very simple and so any UI implementation will not be discussed here. The Reactions & Transport section is mostly UI, and the plant will be obtained as a result of the Responses section. The Population section is also expected to be straightforward. There will be preset ‘populations’ and substrate consistencies, so as the user alters the sliders, different presets will be shown.
It is anticipated that the Responses section, which will simulate the growth of a plant, will be the most programming-intensive.

In the Responses section there will be two main mechanics: Input Handling and Plant Growth.
Input will be used to allow the user to place, delete, scale and rotate three augmented objects. This means that the app must be able to handle input from a mobile device, which is the first aspect to be addressed.

As seen in Figure 2.5.1, a single unmoved touch will trigger the app to select a position or object. This will work firstly by raycasting from the input position to the world position. If there is an augmented object there, that object is selected. If there is no object, Vuforia's ground plan script will take over to recognise the input, allowing an object to be placed in the relative augmented position.

The other two mechanics rely on objects having been placed. One moving input will rotate the object, or if there are two inputs, either of them moving, this will result in all the object being scaled. Scaling will be done by calculating the change in distance between the two inputs. This change will be applied to the objects' scale value in each axis.

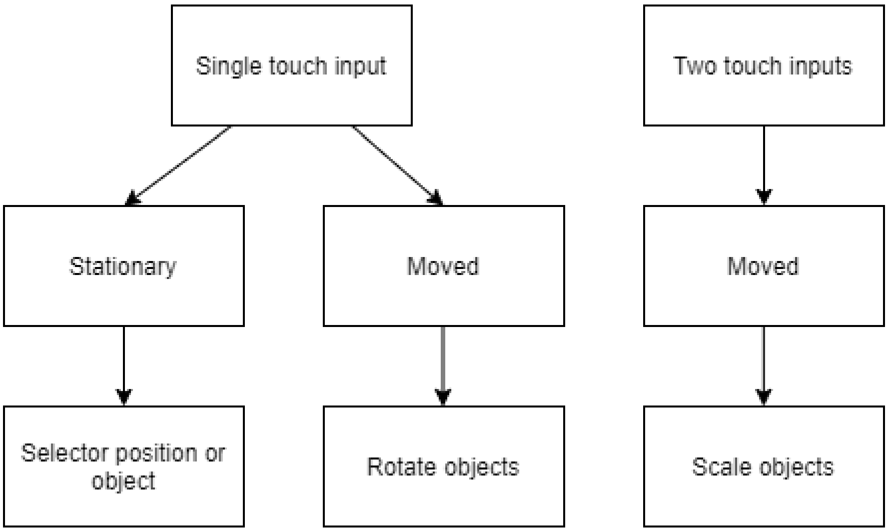

Figure 2.5.1: Input handling chart

Figure 2.5.2 shows a flowchart which would be implemented for plant growth in the Responses section of the app. At every update tick, each node of the plant will run through these steps to decide which actions to take.

This gives an idea of what the final function should look like, because initially, the plant will not respond to the environment and resources, and limiting factors will be ignored.

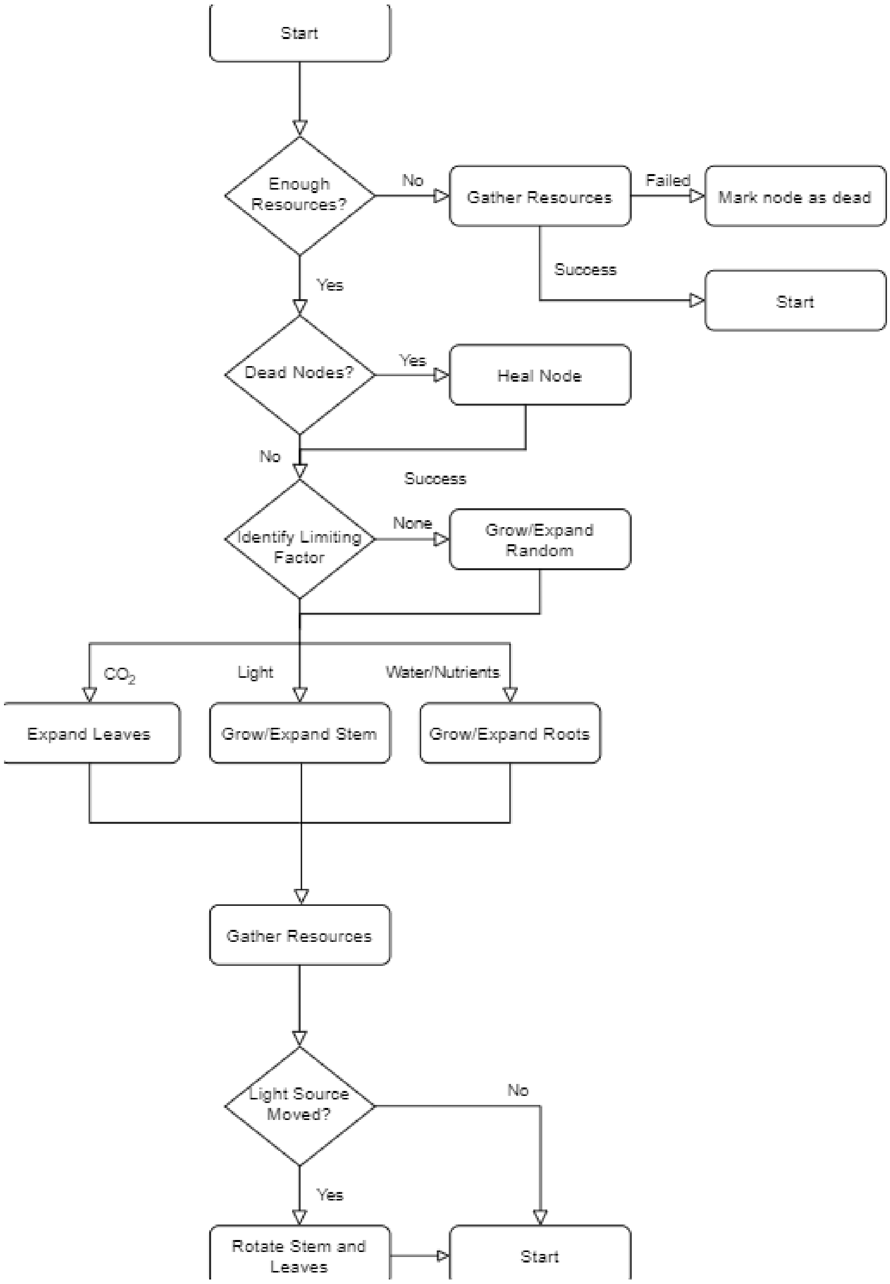

Figure 2.5.2: Basic flowchart for plant simulation

At this point the mechanics were understood, so a test plan could be created (Figure 2.5.3). During development of the app, this table could be used to ensure each mechanic was performing as intended.

Mechanic	**Expected Result**	**Actual Result**
UI is functional	Buttons and sliders work when pressed	
UI appears correctly	All the UI is displayed. It does not overlap	
Image targets are recognised	An AR object is displayed on a target	
Planes are recognised	An AR object is displayed on a plane	
The correct information is shown about plant parts	The information shown correlates to the selected plant part	
The plant is moved with the root cube	The plant does not clip the cube and is not floating above it	
The plant branches at realistic points	The plant branches at leaf nodes	
Stems grow towards the light source	Plants grow toward the light source or between two sources	
Leaves face the light source	Over time leaves tilt to face the light source	
Roots grow towards water	Roots primarily grow towards the water source but still branch in other directions	
A lack or resources kills the plant	The plant yellows/wilts if there are insufficient resources	
Lower leaves die-off	If leaves aren’t getting enough light or there are not sufficient enough resources, lower leaves die-off first	
Limiting factors have an optimum value	There is a best value for all limiting factors	

Table 2.5.3: Bug testing

2.6 Feedback

To receive feedback about this app, participants will be asked to answer some exam-style questions before and after using it. A control group will do the same but using the AQA textbook instead of the app. To implement this, some AQA past papers (2016) were researched to gain an understanding of how to set a question appropriately. After the exam style questions, participants will also answer a short survey asking about their experience of using the app. Hopefully this app will be more engaging, easier to use, and less stressful than using a book, or at least provide another way of learning. Participants' feedback will provide evidence on how AR could be used more in mainstream education.

2.7 Tools and Services

2.7.1 Task Management

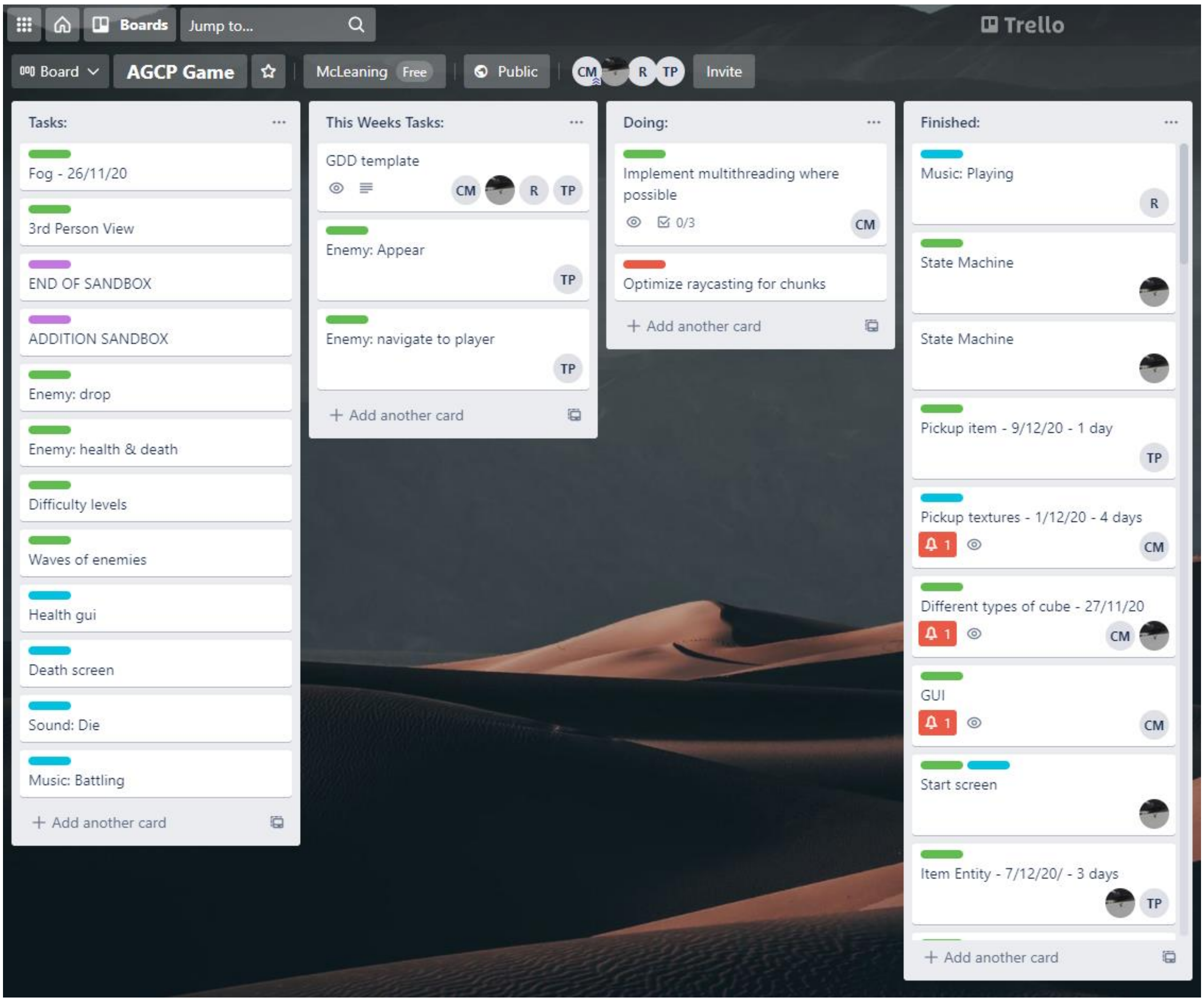

Figure 2.7.1: Trello example

Trello was used to keep track of the tasks. This allows users to create various boards (e.g. to do, work in progress, complete) then add cards to a board. Each card can be a task or part of a task, which can be moved between boards.

2.7.2 File Storage

There are two types of file which will need to be stored: documents and project files. Documents are lightweight and don't take up much space. The options for this are Microsoft Office, Google Drive and storing them on a local hard drive. The first two share the advantage of being stored on well-maintained and protected servers, so the chance of corruption is very low. However, because they are online services, they require the internet in order to access them. Storing on a local hard drive means backups of each file would need to be made frequently in order to prevent any loss of work though corruption or power shortages. Google Docs have been chosen over Microsoft Office because of its integration into Google Chrome.

Unity project files can take up lots of space, particularly as having a backup of each file version can be very useful. Therefore, a git repository was used, allowing for lots of free, online storage. Each time a file is updated the user provides a reason or name for the update and this means that it is easy to backtrack to a particular, different version, allowing old code to be viewed or reverted to if something breaks.

2.7.3 Text Editors and Compilers

Visual Studios is an obvious choice for use as a text-editor and compiler. This text editor provides useful features such as colouring different types of code for ease of reading, and 'IntelliSense' which is an extremely useful feature similar to auto-correct, but for code. There is also an in built compiler which provides useful error messages when necessary. Finally, the debug mode allows the user to step through each line of code as it is being executed, and allows the variables and the raw memory to be viewed.

2.7.4 3D Modelling

Autodesk Maya will be used for modelling because it is relatively easy to use while giving the user lots of control over the model.

3 Development

3.1 Mobile Input and AR Implementation

Setting up the work space was the first requirement. Vuforia was downloaded and added to a Unity project. The Vuforia developers website (2021) has useful tutorials on setup and basic usage.
To test that Vuforia worked, a webcam was pointed at a printed hard copy ‘Ground Plane’ card. This is an A4 sheet of paper with a specific image on it as seen below in the middle window (Figure 3.1.1). In Unity a cube was added to the ground plane and then the cube was shown on the ground plane card.

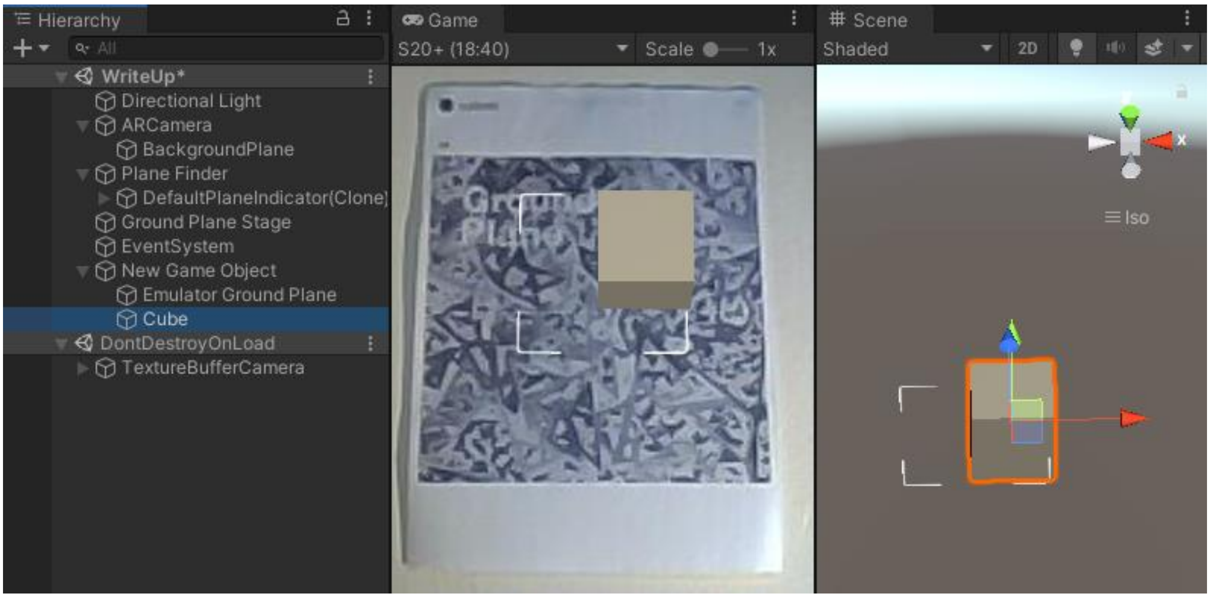

Figure 3.1.1: Basic augmented object placement in Unity. Vuforia Ground Plane with the addition of Unity Cube.

Being able to run the app on a computer using a webcam proved to be very useful. It removed the need to build to a mobile device each time a feature needed testing.
Unity Remote 5 is a mobile app which allows unity projects to be executed on a computer, while being displayed on a mobile. This allows the project to be tested using a mobile screen and input during development. The only problem encountered was the mobile’s camera could not be accessed, so the video stream was provided by the webcam plugged into the computer. Despite these limitations, Unity Remote 5 allowed the user interface (UI) and touch inputs to be tested.
Features were implemented to allow the objects on the ground plane to be rotated (by dragging one finger) and scaled (by pinching with two fingers). Scaling proved a problem

since the intention was to have three placeable objects. If objects are scaled large enough then they may overlap one another. This problem was solved by scaling objects around the Seedling, so that the distance between each object is kept relative. Similarly, when the objects are rotated, they are rotated around the Seedling.

The script shown in figure 3.1.2, shows the implementation of handling two touch inputs.

```
private void HandleTwoTouches(Touch t1, Touch t2)
{
  //Check both touches have just begun, then calculate the initial distance between
    them
  if (t1.phase == TouchPhase.Began || t2.phase == TouchPhase.Began)
  {
    startFingerPosDifferece = Vector2.Distance(t1.position, t2.position);
  }
  //Otherwise if either of the touches have moved, scale the plant
  else if (t1.phase == TouchPhase.Moved || t2.phase == TouchPhase.Moved)
  {
    //Current position minus the start to get the difference between the two
    float currFingerPosDifferece = Vector2.Distance(t1.position, t2.position);
    float scaleFactor = currFingerPosDifferece - startFingerPosDifferece;

    //Check the plant is within the scale limits after the scale. Only scale if
       it's within the limits
    float xScale = gameobjects[(int)GOs.Seedling].transform.localScale.x +
        (scale.x * scaleFactor);
    if (scaleMin < xScale && xScale < scaleMax)
    {
      //Set the scale values and scale the collision checking gameobject
      currentScale += scale * scaleFactor;
      collisionGO.transform.localScale = currentScale;

      //If the seedling has been placed, scale it
      if (placed[(int)GOs.Seedling])
        gameobjects[(int)GOs.Seedling].transform.localScale = currentScale;

      //Attempt to scale the other two objects, while keeping a relative
        distance from the seedling
      ScaleAroundSeedling((int)GOs.Water, scaleFactor);
      ScaleAroundSeedling((int)GOs.Light, scaleFactor);
    }

    //Reset the start position to the current position
    startFingerPosDifferece = currFingerPosDifferece;
```

```
    }
  }
```

Figure 3.1.2: Code snippet for handling two touch inputs

Finally the project was built onto the mobile device to test that everything worked while running solely from the phone and using the phone camera. The only problem which was found here was that the initial size of all of the objects was too small. The prefab of each object was increased in order to overcome this issue.

3.2 Graphical User Interface

The position and size of UI's are determined by the size of the screen they are displayed on. The first step to creating a UI was to add a custom aspect ratio to Unity's Game window which would mimic that of a mobile phone. A ratio of 18:40 was used (Figure 3.2) as this matched the device used for testing.

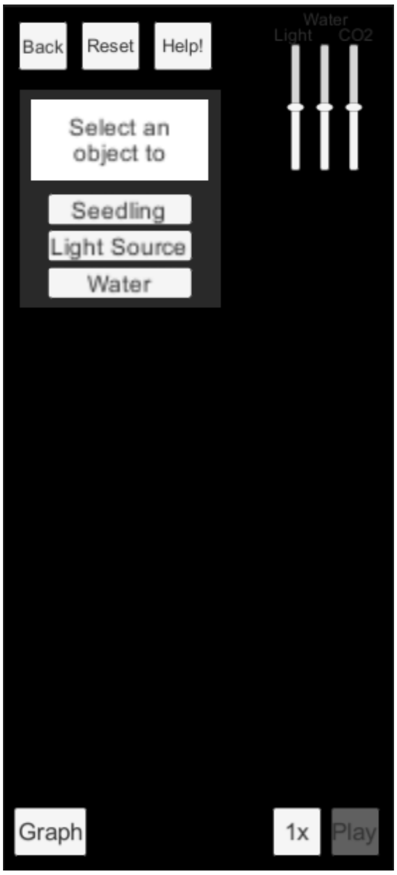

Figure 3.2: UI layout

The ‘Selection’ menu is only activated when the user taps the screen to place one of the three objects. Each button on this menu is active or inactive depending on whether the corresponding object has been placed. There are two additional menus like this which are used to delete objects or alter the height of the light source.
The three sliders in the top right corner are used to alter the level of the three variables: light, water and carbon dioxide (CO_2), which will eventually affect the rate of plant growth.
The Graph button toggles a graph which charts the plant’s growth rate against the variables. This allows the user to clearly see the effect of changing each variable in a different visual format.
The Back button takes the user to the main menu. The Reset button resets the plant’s growth, and, if the plant has not grown, it deletes the placed objects and allows the user to change their positions.
The button with text ‘1x’ changes the playback speed. Each time the user clicks it, the speed doubles, until the maximum speed of 12 times is reached. At this point clicking again, the speed goes to half the original speed (0.5x).

The Play button is only active once all three objects have been placed. Once active, it toggles the growth of the plant. Objects cannot be deleted once the plant has started growing unless reset.

3.3 Unresponsive Plant Growth

The first stage of creating a growing plant in Unity is to make a nonrandomized structure which grows straight upwards, has leaves, branches out at intervals and also enlarges over time. It does not grow differently depending on the resources available or the position of the light sources.
The plan for growing a plant is to have a ‘stem section’ which grows. Eventually another stem section will be produced on top of the first, or on its side. This is repeated until several stem sections have been produced, all of them being connected to at least one other as shown in Figure 3.3.

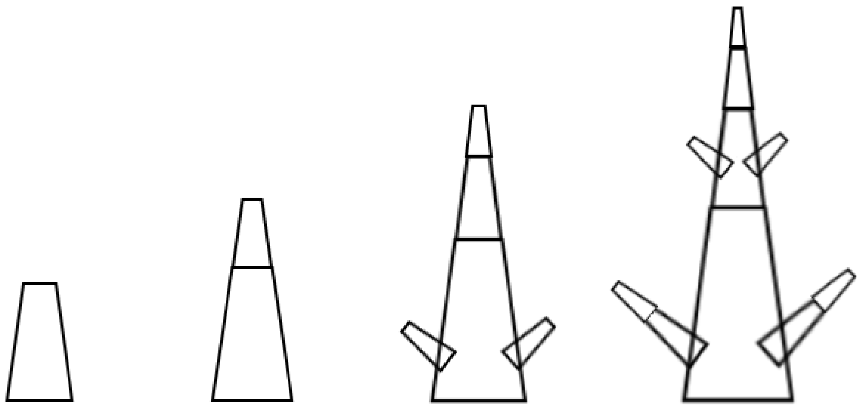

Figure 3.3: 2D diagram of stem sections

A prefabrication (prefab) is a collection of models or game objects which can be spawned into the game world in multiple, separate instances. They can also have other components attached to any of the models within, such as scripts, colliders and rigid bodies.

A game object (object) is an instance of a prefab, or primitive piece of geometry which has been spawned into the game world.

The word 'spawn' or 'spawned' will be used for its computer science meaning, not the biological meaning.

3.3.1 Main Stem

3.3.1.1 Models and Prefabrications

Three models were made in Autodesk Maya: bud, leaf and stem(figure 3.3.1.1.1) . They were all made in the same file so they would all hold the same relative scale. For example the diameter of the stem matches the diameter of the bottom of the bud. They were then imported into Unity as FBX files.

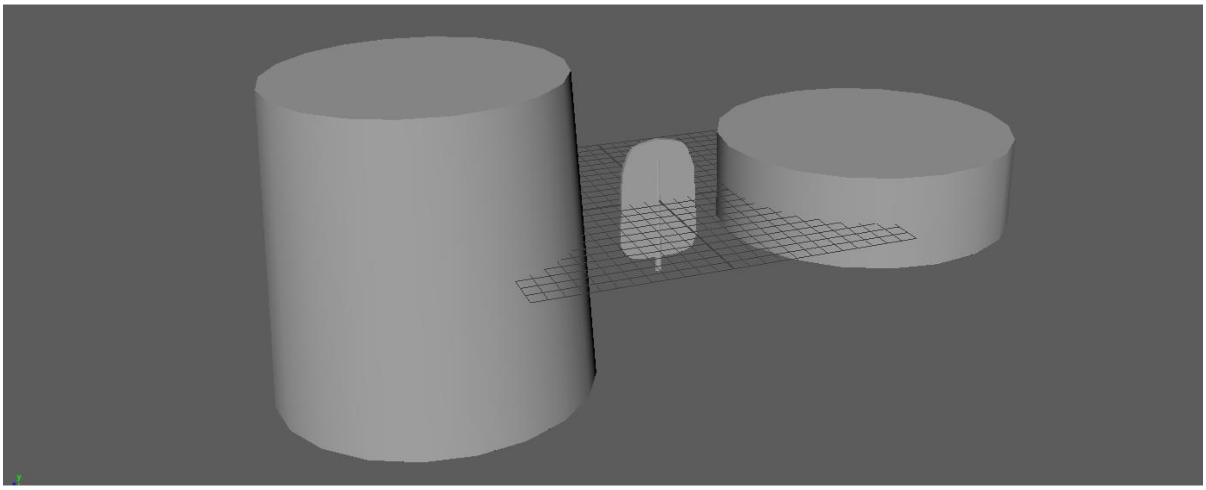

Figure 3.3.1.1.1: Basic models viewed in Autodesk Maya showing a bud model (left hand side), leaf model (middle) and stem model (right hand side)

Once they were imported into Unity, two prefabs were made (Figure 3.3.1.1.2).

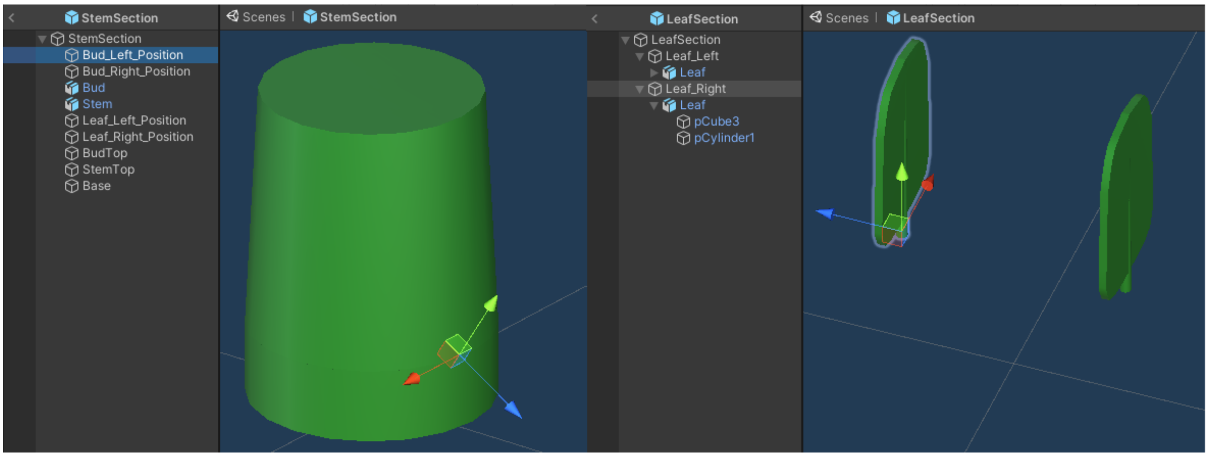

Figure 3.3.1.1.2: First prefabs in Unity

The prefab on the left is the ‘Stem Section’, which had the bud model placed directly on top of the stem. Having two models allows the stem to be expanded upwards, making it taller, while maintaining the same top diameter (which will be useful later on). The other objects visible in the hierarchy are used as locations to spawn other prefabs. This is useful because they maintain a consistent relative position when the entire prefab is transformed. For example, the first child ‘Bud_Left_Position’ is on the positive Z side of the bud with a Y value of 0.062, placing it just above the base of the bud. The first child also has a rotation, so when another Stem Section is spawned as a side stem of this, the new Stem Section can use the position and rotation of its first child. Each subsequent child has a similar use.

The other prefab (right hand side of Figure 3.3.1.1.2) has two of the leaf models. Both are children of empty game objects which are used to control their pivot point during rotation. The ‘Leaf_Right’ object has an additional rotation of 180° in its Y axis. This means that a positive rotation applied to either object will result in it rotating away from the other, and also away from the Stem Section it is attached to. Rotating away from the Stem Section is important as it gives the effect the leaves are opening outwards from the stem.

The positioning of the leaves is made so that if both prefabs are spawned at the same position, then the leaves appear in the correct position on the stem. This also means that when they are transformed, the outcome of the transformation is that the two objects keep their positions relative to one another.

The reason for having two prefabs instead of one, is that the leaves and stem need to be scaled independently.

3.3.1.2 Iteration

With the prefabs ready to be used, the ‘PlantGrowth’ script was created. This script will control the plant's growth.
The first task to deal with was iteration. Each Update of the script would have to check every stem and leaf section, and grow them accordingly. There were three possible ways to deal with this:

- Make each object a child of its parent transform. However this would mean that the leaves could not be scaled independently;
- Have two lists, one for stems and one for leaves. The issue with this would be keeping track of which leaves belonged to which stem, and which stem had grown off of which parent stem;
- A structure could be made to hold information about a stem. The information could include its parent, its leaves and which stems come from it. This seemed to be the most efficient solution.

```
class StemStruct
{
  public int id = -1;

  public Transform container;
  public Transform thisBud;
  public Transform leftLeaf;
  public Transform rightLeaf;
  public Transform stem;

  public StemStruct parent;
  public StemStruct topBud;
  public StemStruct leftBud;
  public StemStruct rightBud;

  public Vector3 emergeDir = Vector3.up;

  public bool emerged = false;
  public bool leftLeafDead = false;
  public bool rightLeafDead = false;
}
```

Figure 3.3.1.2.1: Code snippet of the Stem Struct

The snippet in Figure 3.3.1.2.1. shows the StemStruct class, which holds all of the information on one stem section. The transforms are intended to be self-explanatory. The ‘container’ is the transform of the Stem Section, ‘thisBud’ is for the ‘Bud’ model within the Stem Section. Similarly ‘stem’ is for the ‘Stem’ model within the Stem Section. Each Stem Struct has references to their parent Stem Struct and all of their children Stem Structs.

The ‘emergeDir’ vector is used during the first stage of a bud's growth, telling it which way to emerge from its parent. Top buds emerge out of the top of their parent, i.e. parent’s local Y axis, while side buds emerge out of the side of their parents, i.e. their parent’s local X or Z axis.

Because all Stem Sections have access to their children, the script only needed direct access to the base stem. By iterating the base stem, all other Stem Sections are iterated. When the Play button is pressed, if the first stem has not already been spawned it is created. Then each time the Update function is run the Bud Tick function is called, with the first stem passed in as the parameter. Since the first stem has no parent, it has already emerged once it is spawned. If it tried to emerge it would encounter an error because it has no parent to emerge from.

The Bud Tick function is recursive and it takes a StemStruct as a parameter. Firstly it grows the stem and bud. Then, providing it has emerged, it goes on to try to spawn a top bud. Next, it checks if the left bud exists. If it does exist it iterates through that stem. Otherwise, it grows the left leaf. This also happens for the right bud and leaf. Then it checks whether the top bud exists and tries to spawn or emerge the side buds.

```
private void BudTick(StemStruct bud)
{
  //Grow the stem and keep the bud on top of it
  GrowStem(bud);
  //Grow the bud and keep the leaves and side buds in their position
  GrowBud(bud);

  //If the bud has emerged, it can grow in other aspects
  if (bud.emerged)
  {
    //Spawn a new top bud. If a bud has not been grown above this one and this bud
      is big enough
    if (bud.topBud == null && bud.thisBud.lossyScale.x > budMinScaleBeforeNewSpawn)
```

```
    {
      SpawnNewTopBud(bud);
    }

    //Grow the side bud or the leaf.
    if (bud.leftBud != null)
    {
      BudTick(bud.leftBud);
    }
    else
    {
      GrowLeaf(bud.leftLeaf);
    }
    if (bud.rightBud != null)
    {
      BudTick(bud.rightBud);
    }
    else
    {
      GrowLeaf(bud.rightLeaf);
    }

    //Grow a side bud once the top bud has emerged
    if (bud.topBud != null)
    {
      if (bud.topBud.emerged)
      {
        if (bud.leftBud == null)
          SpawnNewSideBud(bud, true);
        if (bud.rightBud == null)
          SpawnNewSideBud(bud, false);
      }
      else
      {
        EmergeBud(bud.topBud);
      }

      //Update the top bud
      BudTick(bud.topBud);

      //Update the side buds if they exist
      if (bud.leftBud != null)
      {
```

```
            if (!bud.leftBud.emerged)
            {
              EmergeBud(bud.leftBud);
            }
            BudTick(bud.leftBud);
          }
          if (bud.rightBud != null)
          {
            if (!bud.rightBud.emerged)
            {
              EmergeBud(bud.rightBud);
            }
            BudTick(bud.rightBud);
          }
        }
      }
    }
```

Figure 3.3.1.2.2: Code snippet of the iteration function

3.3.2 Basic Growth Functionality

3.3.2.1 Top Buds

When a new Top Bud is spawned, it is given the position of the top of its parents stem (the bottom of the bud) so it spawns within its parent. It also has its parents rotation, with a random value between 25° and 75° added to the Y axis. This prevents the leaves from overlapping each other and adds a level of randomness to make the plant appear more natural. It's 'emerge direction' is the same as its parent.

A Stem Struct is created for the new bud, and parent and child are linked through it.

Once the new Top Bud has been spawned it slowly emerges from its parent. This is done by adding its 'emerge direction' to its current position. Initially buds were set to emerge for a set amount of time, but this led to having to scale a timer to match the scale of the parent. In the end a different method was used.

A collision capsule component was added to the bud of the Stem Section prefab. This allows the stem to know when it has finished emerging by checking if its next position is outside of its parents collision capsule.

Once emerged, a bud is iterated as a normal stem section with the 'Bud Tick' function.

3.3.2.2 Side Buds

Creating a Side Bud is similar to creating a Top Bud. The differences are the rotation and emerge directions.
The rotation is found in one of the parent's children objects, either 'Bud_Left_Position' or 'Bud_Right_Position', see Figure 3.3.1.1.2. This gives the side bud a rotation so that it grows outward of its parent stem, and so that the side buds leaves will grow perpendicular to the leaves of its parent.
The emerge direction is found by subtracting the position at the top of the side stems bud from the bottom of the side stems stem. This produces a vector pointing in the side stems local positive Y direction. The vector is normalized for later use.
The stem is spawned within its parent. Using its emerge direction it emerges from the side of its parent instead of out the top.

3.3.2.3 Leaves

The leaves grow in two ways: scale and rotation. These are both handled in the same way. They are incremented each iteration by a certain value, until they reach a preset maximum value.
Leaves die once a side bud has emerged from their location. While a leaf is dying it undergoes the opposite of growth. It is scaled down and keeps rotating, eventually it is deleted when it is either too small to be seen or has rotated back inside its parent stem. This mimics a leaf wilting.

3.3.2.4 Relative Positions

At each iteration every stem is scaled. When a stem has top or side stems or leaves, they slowly become engulfed by their parent. As seen in figure 3.3.1.1.2, a Stem Section has a number of empty game objects which are used to hold a position for the children. They are always in the same place on the parent relative to the Stem Sections scale, rotation and position.
Whenever a Stem Section is transformed a function is called to set the position of each of its children to their corresponding, relative position.

3.3.3 Modifications

Two additional classes were created in place of enumerations. They hold constant integers which can be used to locate children within the hierarchy of Stem or Leaf Sections. This is useful when the prefab of the Stem Section is altered to allow for 'initial growth'.

3.3.3.1 Initial Growth

Initial growth occurs immediately after the Stem Section has emerged. During this phase, the stem grows in its local Y axis, but the bud and leaves do not grow. This was implemented so that the bud and leaves of a stem only start growing when they are at a distance from their parents, to prevent their overlapping.

It was important that the Collision Capsule component was on the bud child here, and that children were spawned at the base of the bud, not the stem.

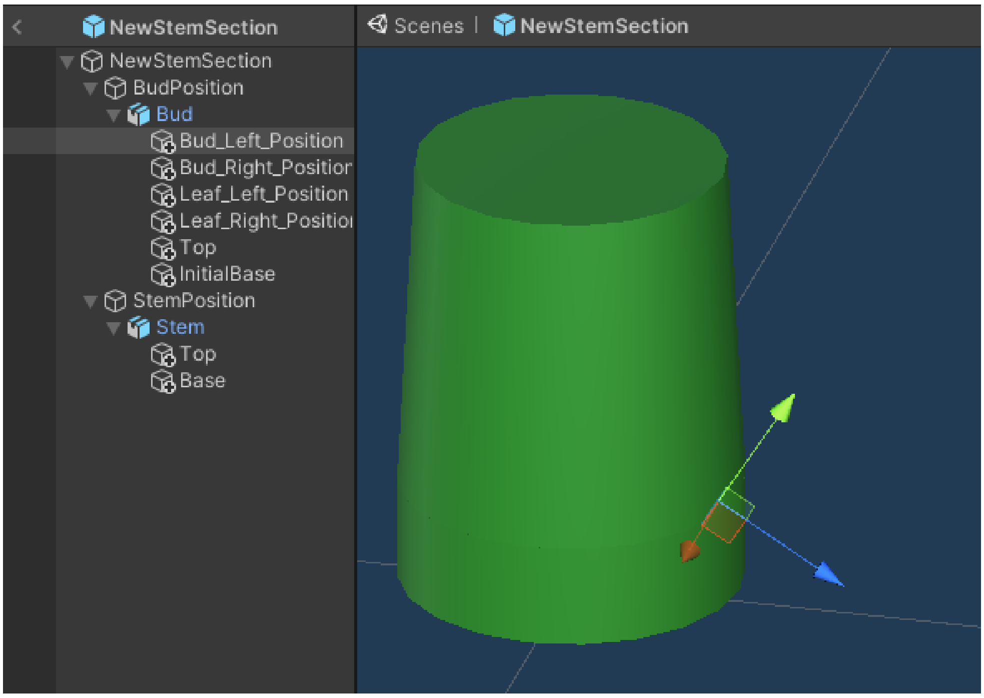

Figure 3.3.3.1: New Stem Section prefab

3.3.3.2 Leaf Array

Throughout the Plant Growth script there was lots of duplication of code because of the presence of two leaves. To solve this a list of Leaf Structs was made in the Stem Struct class. This allowed both leaves to be accessed with an index. The index matched up with the child index of the left and right side stems (see figure 3.3.3.1), which allowed for some easy optimizations.

```
class LeafStruct
{
  public LeafStruct(Transform t)
  {
    leafTransform = t;
  }
  public Transform leafTransform;
```

```
    public bool leafFullyGrown = false;
    public float leafDead = 0;
}
```

Figure 3.3.3.2: Code snippet of the Leaf Struct

The side stems of each Stem Section were also put into an array, meaning that the index used to access a leaf could access the corresponding side stem.

3.4 Responsive Plant Growth

At this point the plant could grow straight upwards, with leaves growing and then dying to allow side stems to branch off. The next things the plant needed to do were to grow towards a light source, to generate energy and to only grow when it had energy.

3.4.1 Energy

3.4.1.1 Resources

In order to generate energy a plant photosynthesises. This is the process of turning water and CO_2 into chemical energy using light. As long as a plant has enough of these three resources it can grow, and the more of them it has, the more energy it can produce. However there comes a point where having more of a resource does not allow a plant to produce more energy, so the resource is then in excess.

Figure 3.4.1.1 shows the variables defined for use in resource gathering and photosynthesis:

```
//Current and max values
public float lightLevel = 0.5f;
public float waterLevel = 0.5f;
public float co2Level = 0.5f;
private const float maxLightLevel = 0.9f;
private const float maxWaterLevel = 0.6f;
private const float maxCO2Level = 0.7f;

//Values to photosynthesise
private const float waterNeededPerEnergy = 0.2f;
private const float lightNeededPerEnergy = 0.1f;
private const float co2NeededPerEnergy = 0.1f;
```

Figure 3.4.1.1: Code snippet of energy generation variables

The first three are variables that the user can change throughout the growth of the plant. The second set of three are the limiting factors, and the final three are how much of each resource is needed to produce one unit of energy.

Light and CO_2 are implemented simply, because each leaf - where photosynthesis occurs - has access to the same amount of these resources. The amount of these that a leaf has is directly related to the values that the user has provided.

Water on the other hand is passed up from the base of the plant. So if the first stem section has excess water after photosynthesis, that water is passed onto its children's Stem Sections equally. This process repeats until the last Stem Section.

3.4.1.2 Photosynthesis

Now that leaves have access to resources they can generate energy. At Each Update a function is run to do this on each Stem Section. If a Stem Section has only one leaf, the amount of energy it can generate is halved. As long as the leaves have enough resources to generate energy they do so, resulting in an energy unit, but depleting the used resources. This keeps on happening until their resources are depleted.

```
//Generate as much energy as possible with the given resources
while(str.water > waterNeededPerEnergy && str.light > lightNeededPerEnergy && str.co2
> co2NeededPerEnergy)
{
  newEnergy += energyBaseValue;
  str.water -= waterNeededPerEnergy;
  str.light -= lightNeededPerEnergy;
  str.co2 -= co2NeededPerEnergy;
}
```

Figure 3.4.1.2: Code snippet of the Photosynthesis function

3.4.1.3 Energy Usage

Plants require energy to do anything, so a set of variables were defined to determine how much energy is needed for any particular task. Then whenever a function was entered, such as ‘SpawnNewSideBud’ or ‘GrowLeaves’ an if statement would test whether the Stem Section had enough energy to perform the task. If there was not sufficient energy the function returns, but if there is enough, the function is entered and the energy deducted.

```
//Energy needed for certain tasks
private const float energyNeededForNewBud = 20f;
private const float energyNeededToEmergeBud = 1f;
```

```
private const float energyNeededToGrowLeaves = 0.1f;
private const float energyNeededScaleBud = 0.5f;
```

Figure 3.4.1.3.1: Code snippet of more energy generation variables

These costs needed lots of balancing in order to make the plant grow properly and at a good speed. The plant cannot spend all of its energy on growing leaves before it grows more stems.

The problem which arose from this was that, if a Stem Section generated less than 0.2 energy, then only one leaf could be grown. Because of the layout of the code, this meant that one leaf would grow while the other wouldn't change. A flag was implemented, which was an integer value which alternated between 0 and 1 each Update. This was used in the 'GrowLeaves' function seen below (Figure 3.4.1.3.2). Since the leaves were stored in a list they could be accessed with the flag value, thus alternating which leaf is updated every frame.

```
private void GrowLeaves(StemStruct str)
{
  if (str.leavesAlive[0] || str.leavesAlive[1])
  {
    Photosynthesise(str, str.normal * str.scale);

    if(str.leavesAlive[flag1])
      UpdateLeaf(str, flag1);

    if(str.leavesAlive[flag2])
      UpdateLeaf(str, flag2);
  }
}
```

Figure 3.4.1.3.2: Code snippet of the function which grows a set of leaves each tick

3.4.2 Growth Towards Light

To make the plant more realistic, the user can place a light source while placing the seedling, and the plant will grow in relation to its position.

3.4.2.1 Leaves

Leaves naturally turn so that as much of their surface is facing the light. To do this the normal of a leaf needs to be roughly parallel with the direction from the leaf to the light source.

Every time the leaf was updated, it would check if its normal was within +/- 0.1 of the direction to the light source. Unless it was within that range, the leaf would rotate until it faced the light source.

3.4.2.2 Stem

Whenever a top bud is spawned it is rotated slightly so that its local Y axis is closer to the direction towards the light source. This rotation was done using the ‘FromToRotation’ function in the Quaternion class. This provides the whole rotation but the plant needs to rotate by only a small amount each section. The Lerp function in the same class was used to achieve this. This requires a value between 0 and 1 telling it how far to lerp the result. Such a value was added to each Stem Section. The ‘first stems’ value was set to 0.025, which is the same as the ‘lerp incremental value’. Then each time a top stem is spawned, its lerp value is set to its parents plus the incremental value. When a side stem is spawned, it’s lerp value is set to 0.

```
//Rotate the stem towards the light source
Quaternion toLight = Quaternion.FromToRotation(newStem.transform.position,
lightSource.transform.position);
toLight = Quaternion.Lerp(newStem.transform.rotation, toLight, str.lightLerpValue);
newStem.transform.rotation = toLight;
newLeaves.transform.rotation = toLight;
```

Figure 3.4.2.2: Code snippet of how the lerped angle to the light is found for a Top Bud

3.4.2.3 Energy Update

Now that the plant grew towards the light and generated energy, it seemed unrealistic that the rate of photosynthesis did not depend on the size of each leaf or its orientation in terms of the light source. These issues were fixed by giving Stem Section two variables used to hold the combined normal and the combined scale of its leaves. The values were updated after both leaves are updated as seen in Figure 3.4.2.3.

```
str.normal = (1f - (str.leaves[0].normal * str.leaves[1].normal)) *
energyGeneratedNormalMult;
str.scale = str.leaves[0].scale * str.leaves[1].scale * energyGeneratedScaleMult;
```

Figure 3.4.2.3: Code snippet of how the photosynthesis scale value is calculated

When that Stem Section next undergoes photosynthesis, the normal and scale values are multiplied and passed into the function, and the amount of energy generated is multiplied by that. This gave the desired result of the scale and rotation of leaves affecting the rate of energy generation.
However this also caused an issue because upon spawning the normal of a leaf is 0. This meant that a new Stem Section could never generate any new energy. To solve this whenever a Stem Section generated energy, some of that energy was moved into 'energy stores'. This is energy that is passed through to its children upon spawning, allowing the children to start increasing the size of their leaves as soon as they have emerged.

3.5 Finishing Touches

3.5.1 Graphical Representation

A graph was implemented in order for the user to easily tell how altering each variable affected the plant's growth. Similar static graphs are depicted in the AQA Biology textbook. On the graph, each individual resource value was plotted along with a growth rate variable which was calculated.

```
float tempWater = waterLevel;
if (waterLevel >= waterNeededPerEnergy)
{
  tempWater -= waterNeededPerEnergy;
  tempWater *= waterMult;
  tempWater /= 3;
}
growthRate = tempCO2 + tempLight + tempWater;
```

Figure 3.5.1.1: Code snippet of the part of how growth rate value is found

As seen in figure 3.5.1.1, where the 'waterMult' was found from the below equation:

```
private const float waterMult = 1f / (maxWaterLevel - waterNeededPerEnergy);
```

Figure 3.5.1.2: Code snippet of how the 'waterMult' variable is calculated

Both calculations were done with each resource, then they were all added together to give a value between 0 and 1, which could be plotted on the graph.

3.5.2 Time Scale

To allow the user to quickly view the effects of certain resource levels, the speed button can be used to change the overall time scale of the plant's growth. This was simply implemented by creating a local ‘scaled delta time’ variable, which is updated each frame and found by multiplying the time scale by the current delta time.

3.5.3 Tutorial

Two tutorials were set up to help first-time users. The first tutorial tells the user how to place and delete objects. The second explains how to grow the plant, control its resources and view the graph.

3.6 Bug Testing

The testing plan which was designed in section 2.5 was used to see how many of the planned features had been successfully implemented, and of those implemented, which worked as planned.

Mechanic	**Expected Result**	**Actual Result**
UI is functional	Buttons and sliders work when pressed	As expected
UI appears correctly	All the UI is displayed, none is overlapping	As expected
Image targets are recognised	An AR object is displayed on a target	This feature was not used
Planes are recognised	An AR object is displayed on a plane	As expected
The correct information is shown about plant parts	The information shown correlates to the selected plant part	This feature was not implemented
The plant is moved with the root cube	The plant does not clip the cube and is not floating above it	This feature was not implemented

The plant branches at realistic points	The plant branches at leaf nodes	As expected
Stems grow towards the light source	Plants grow toward the light source	The plant can grow towards the light source
Leaves face the light source	Over time leaves tilt to face the light source	Leaves can rotate along their local Z axis towards the light
Roots grow towards water	Roots primarily grow towards the water source but still branch in other directions	Roots were not implemented
A lack or resources kills the plant	The plant yellows/wilts if there aren't enough resources	A lack of resource only halts plant growth
Lower leaves die off	If leaves aren't getting enough light or there aren't enough resources lower leaves die off first	Leaves only die when a new stem is made in its place
Limiting factors have an optimum value	There is a best value for all limiting factors	As expected

Table 3.6: The completed in bug testing table

Two people were asked to bug test the app. These are the notes taken from their sessions:

Participant 1:

- Some menus overlapped the tutorial
- Explain the collision cube when placing objects
- Ability to change the light source height in the delete menu
- Scale down the height variability of the light source
- Some spelling errors

Participant 2:

- Automatically show the tutorial instead of telling the user to press the button

- Make the sliders bigger, for ease of use
- The help button failed to work for the second tutorial
- Explain that the plant stops growing when the graph is viewed
- The side buds displayed a bug where they didn't grow towards the light

Once these were all considered and fixed, the app was complete.

4 Participant Feedback

Due to the timing of this project overlapping that of the Covid-19 pandemic, the intended participants could not be tested. Instead of allowing people who had just finished their A-Levels to participate, anybody within the Researchers bubble was tested.

Participants ran the app and answered preset questions after they had used it. These are the questions they answered:

1. What do you think this app is trying to teach you?
2. Which aspect of the app did you find most engaging?
3. Can you picture this app being used in a learning environment, such as a school or college?
4. (Presuming this app was further developed) would you rather use the app or a book to learn from, or a mixture of both?
5. Do you think other subjects could be taught effectively using apps such as this?

The participants' responses can be seen in Appendix D.
All of the participants understood that the app was demonstrating the relationship that water, light and carbon dioxide have to the growth of a plant. Therefore it can be considered that the app could be used to effectively help teach about plant growth and limiting factors in Biology.
In question 2 the answers were either: using the sliders and graph, or watching the plant grow. The former indicates that some participants would have been just as engaged viewing the plant in an entirely virtual environment, while the latter shows that others enjoyed being able to watch it grow in an augmented space. One participant said they found it engaging "Being able to ... move in and out without having to use the interface to do that". This is a prime way in which AR can be useful: the user can view the augmented object from any position without having to directly interact with the app.

All participants agreed that they could see the app being used in a learning environment. However one participant went on to say they could only see it in use in a lower level school due to its simplicity. A published product would be far more in-depth and attuned to the demands of the A-Level syllabus.
The majority of participants said they would rather learn with a mixture of books and apps such as this one. This was the expected outcome as the aim of this app is to aid the teaching, not to replace other media such as books or teachers.
Finally, all participants agreed that they could see other subjects being taught with similar apps.

This feedback shows that similar apps could be used to help teach subjects in schools and colleges with the use of AR.

5 Evaluation

5.1 Project Evaluation

This project has shown that AR can be used as a teaching aid. It helps to visualise the direct impact of limiting light, water and CO2 in plant growth. This app does not contain enough information or explanation of what is occurring and why. This is why it needs to be accompanied by a book or a teacher.
Enough research was done on this project to create the final deliverable, but very little was done in the field of teaching. More research in this area could have led to different features being implemented, and features such as the graph could have been developed differently.
Many of the tasks within the app development were misjudged for their complexity. This meant that the amount of time dedicated to each task was generally greater than anticipated. However the planned finish date for the deliverable was set 2 months before the deadline giving plenty of room for error.
Due to the time constraints, only one out of the three sections of the app were created. In figure 2.4 all three sections are shown, however they provided less interactivity with the augmentation aspect, so they were not prioritized. They could be developed at a later stage. The section which was developed demonstrates plant growth to a satisfactory level where it is clear what is being simulated.
Throughout most of the project Trello was used effectively. Reliance of Trello reduced towards the end of the project as there were fewer tasks to manage.

5.2 Further Development

As previously mentioned, only one of three sections were developed. Developing the other two sections would make the app much better suited to teaching its intended subject in biology. So this would be the first logical step in further development.
In the Responses section there are some improvements and advances which could be made:

- Leaf model could be improved;
- Germination stage could be included, so the plant grows from a seed;
- The plant could undergo its reproductive cycle, so producing flowers,fruit seeds;
- Make the plant die if too few resources are present.

5.3 Individual Development

Creating an app for education is something I have not done before. During research and development, different aspects (such as how to best convey a message without explicitly explaining it) needed to be thought through . This learning experience could have been aided by reading more papers about education and the teaching and learning process.
If I'd known how long it would take to create a reactive plant simulation, the scope of the project would have been much smaller. This was a lesson on how long programming tasks can take. It is also important to learn how much time is required for bug fixing.

Now that the project is finished, I'm happy to have gained an understanding of how to create apps for educational purposes, and to have learned how to implement basic AR. These are the two things I set out to achieve in this project.

Previously I had not created a simulation of anything like this. Now I feel more confident and capable in doing so. I can plan further ideas and projects to pursue.

6 Conclusion

This project was designed to create an app which could help teach a small but important part of the A-Level biology syllabus using augmented reality. To some extent this was achieved, the deliverable showcases limiting factors, which is a small section of the Energy Transfer subject. However because it only helps with such a small part, it cannot be said that the app helps to teach all of the subject of Energy Transfer. It has, however, proved that AR apps could be used to aid teaching, so this project works more as a proof-of-concept. This was demonstrated to a limited extent by participant feedback. Due to Covid-19 restrictions in place at the time of developing this app the available participant sample size proved too small to deliver any definitive results.

Bibliography

Biology:

AQA. (2016). AQA biology past papers & mark schemes. Retrieved from https://www.aqa.org.uk/subjects/science/gcse/biology-8461/assessment-resources?f.Sub-category%7CF=Sample+papers+and+mark+schemes&start_rank=21&cache=1

Morris, J. (2016). *Biology* Specification(Second edition, first printing ed.). New York: W.H. Freeman & Company, a Macmillan Education Imprint. Retrieved from https://www.aqa.org.uk/subjects/science/as-and-a-level/biology-7401-7402/subject-content

Coordination Group Publications Ltd, (CGP). (2015). *A-level biology: AQA year 1 & 2 complete revision & practice with online edition*

Ghulam Shabiralyani. (2015). Impact of Visual Aids in Enhancing the Learning Process Case Research. Retrieved from https://files.eric.ed.gov/fulltext/EJ1079541.pdf

Other AR Apps:

Aquila Education Apps (2020). AR Skeleton (Version 0.5) [Mobile app software]. Retrieved from https://play.google.com/store/apps/details?id=com.Aquila.ARSkeleton

Aquila Education Apps (2020). AR Curriculum(Version 0.4) [Mobile app software]. Retrieved from https://play.google.com/store/apps/details?id=com.aquila.arcurriculum

Arthur Arzumanyan (2020). AR Solar System (Version 1.5.10) [Mobile app software]. Retrieved from https://play.google.com/store/apps/details?id=com.ar.solar

Edmentum, Inc. (2020). Edmentum AR Biology (Version 1.2) [Mobile app software]. Retrieved from https://play.google.com/store/apps/details?id=com.edmentum.edmentumsciencear

Media apps Technologies for the BBC (2020). Civilisations AR (Version 2.0) [Mobile app software]. Retrieved from https://play.google.com/store/apps/details?id=uk.co.bbc.civilisations

Kayan Tech (2020). Plant A Tree AR (Version 1.2) [Mobile app software]. Retrieved from https://play.google.com/store/apps/details?id=com.kayan.nurav.artree

AR Toolkits:

ARCore. Retrieved from https://developers.google.com/ar/discover

Apple. ARKit. Retrieved from https://developer.apple.com/augmented-reality/arkit/

ARKit vs ARCore. (2017). Retrieved from https://www.newgenapps.com/blog/arkit-vs-arcore-the-key-differences/

ARToolkitX. Retrieved from http://www.artoolkitx.org/docs/about/

Best AR SDK for development for iOS and android. Retrieved from https://thinkmobiles.com/blog/best-ar-sdk-review/

EasyAR. Retrieved from https://www.easyar.com/

Maxst. Retrieved from http://maxst.com/#/

PTC, Inc. Vuforia. Retrieved from https://developer.vuforia.com/

Technologies, U.Unity MARS. Retrieved from https://unity.com/products/unity-mars

Wikitude. Retrieved from https://www.wikitude.com/

Vuforia developers portal. Retrieved from library.vuforia.com

Appendices

Appendix A - Project Specification

PROJECT SPECIFICATION - Project (Technical Computing) 2020/21

Student:	Callum Peter McLean, 28015292
Date:	07/10/2020
Supervisor	Penny Collier
Degree Course:	BSc Computer Science for Games
Title of Project:	A prototype AR app to enhance learning of A-Level Biology

Elaboration

Biology is a complicated area of science. As its content is based in the tangible, real world, lots of the learning of that content could benefit from a 3D visualisation teaching aid. The aim of this project is first to identify a subject in which an AR mobile app could effectively help teach, then to develop the app for a mobile device. In order to create an app which would have potential for real-world use, this project is focused on A-Level students and the syllabus they learn from. People within this group usually have access to smartphones powerful enough to run AR software, meaning most people in this audience would have access to my app. During this project I will learn about the implementation of AR technologies into educational apps.

Project Aims

- Research into A-Level biology and AR mobile app development
- Create and test the app
- Critically evaluate my progress, the final app, the feedback and my response to the feedback

Project deliverable(s)

The deliverable will be an app for mobile devices. It's purpose will be to aid the teaching and learning of a specific subject of biology within the A-Level syllabus by using AR.

Action plan

Task	Date
Create specification and ethics forms	14/10/20
Set up Trello and a project diary for time management	15/10/20
Gather a list of research resources for AR mobile development and A-Level biology for an information review	16/10/20
Identify a topic within the syllabus of A-Level biology to focus on	19/10/20
Create a contents page	19/10/20
Make sure I understand the biology topic to a good extent	2/11/20
Decide which tool to use to develop the AR app	20/10/20
Learn how to use said tool for mobile development	20/11/20
Design UML diagrams and a plan for the app	26/11/20
Create a test plan	30/11/20
Draft evaluation of my research	12/12/20
Set up Git for version control	15/12/20

Create the app and test it: ● Create the main mechanics, with block graphics, in a basic 3D space ● Make the app work on a mobile using AR ● Import final models, UI and other graphics and make adjustments ● Run through the test plan ● Fix any found bugs	15/02/21 11/01/21 28/01/21 04/02/21 08/02/21 15/02/21
Draft evaluation of the app development and testing	01/03/21
Finalise the information review	02/03/21
Find feedback candidates and finalise tests, surveys, questionnaires and testing materials	05/03/21
Gather feedback on the app	12/03/21
Write up my final evaluation	15/03/21

BCS Code of Conduct
I confirm that I have successfully completed the BCS code of conduct on-line test with a mark of 70% or above. This is a condition of completing the Project (Technical Computing) module.
Signature:
CMcLean

Publication of Work
I confirm that I understand the "Guidance on Publication Procedures" as described on the Bb site for the module.
Signature:
CMcLean

GDPR
I confirm that I will use the "Participant Information Sheet" as a basis for any survey, questionnaire or participant testing materials. This form is available on the Bb site for the module and as an appendix in the handbook.
Signature:
CMcLean

Appendix B - Ethics Form UREC2

UREC2 RESEARCH ETHICS PROFORMA FOR STUDENTS UNDERTAKING LOW RISK PROJECTS WITH HUMAN PARTICIPANTS

This form is designed to help students and their supervisors to complete an ethical scrutiny of proposed research. The University R HYPERLINK "https://www.shu.ac.uk/research/ethics-integrity-and-practice" HYPERLINK "https://www.shu.ac.uk/research/ethics-integrity-and-practice" HYPERLINK "https://www.shu.ac.uk/research/ethics-integrity-and-practice"eshould be consulted before completing the form. The initial questions are there to check that completion of the UREC 2 is appropriate for this study. The final responsibility for ensuring that ethical research practices are followed rests with the supervisor for student research.

Note that students and staff are responsible for making suitable arrangements to ensure compliance with the General Data Protection Act (GDPR). This involves informing participants about the legal basis for the research, including a link to the University research data privacy statement and providing details of who to complain to if participants have issues about how their data was handled or how they were treated (full details in module handbooks). In addition the act requires data to be kept securely and the identity of participants to be anonymized. They are also responsible for following SHU guidelines about data encryption and research data management. Information on the Ethics Website

The form also enables the University and College to keep a record confirming that research conducted has been subjected to ethical scrutiny.

The form may be completed by the student and the supervisor and/or module leader (as applicable). In all cases, it should be counter-signed by the supervisor and/or module leader, and kept as a record showing that ethical scrutiny has occurred. Some courses may require additional scrutiny. Students should retain a copy for inclusion in their research projects, and a copy should be uploaded to the relevant module Blackboard site.

Please note that it may be necessary to conduct a health and safety risk assessment for the proposed research. Further information can be obtained from the College Health and Safety Service.

Checklist Questions to ensure that this is the correct form

1. Health Related Research with the NHS or Her Majesty's Prison and Probation Service (HMPPS)or with participants unable to provide informed consent

Question	Yes/No
• Does the research involve?	
• Patients recruited because of their past or present use of the NHS	No
• Relatives/carers of patients recruited because of their past or present use of the NHS	No
• Access to data, organs or other bodily material of past or present NHS patients	No
• Foetal material and IVF involving NHS patients	No
• The recently dead in NHS premises	No
• Prisoners or others within the criminal justice system recruited for health-related research*	No
• Police, court officials, prisoners or others within the criminal justice system*	No
• Participants who are unable to provide informed consent due to their incapacity even if the project is not health related	No
• Is this a research project as opposed to service evaluation or audit? *For NHS definitions of research etc. please see the following website*	No

http://www.hra.nhs.uk/documents/2013/09/defining-research.pdf	

If you have answered **YES** to questions **1 & 2** then you **MUST** seek the appropriate external approvals from the NHS, Her Majesty's Prison and Probation Service (HMPPS) under their independent Research Governance schemes. Further information is provided below.
h HYPERLINK "https://www.myresearchproject.org.uk/Signin.aspx" HYPERLINK
NB College Teaching Programme Research Ethics Committees (CTPRECS) provide Independent Scientific Review for NHS or HMPPS research and initial scrutiny for ethics apps as required for university sponsorship of the research. Applicants can use the IRAS proforma and submit this initially to their CTPREC.

- **Checks for Research with Human Participants**

Question	Yes/No
• Will any of the participants be vulnerable? *Note: Vulnerable' people include children and young people, people with learning disabilities, people who may be limited by age or sickness, people researched because of a condition they have, etc. See full definition on ethics website*	No
• Are drugs, placebos or other substances (e.g. food substances, vitamins) to be administered to the study participants or will the study involve invasive, intrusive or potentially harmful procedures of any kind?	No
• Will tissue samples (including blood) be obtained from participants?	No
• Is pain or more than mild discomfort likely to result from the study?	No
• Will the study involve prolonged or repetitive testing?	No
• Is there any reasonable and foreseeable risk of physical or emotional harm to any of the participants? *Note: Harm may be caused by distressing or intrusive interview questions, uncomfortable procedures involving the participant, invasion of privacy, topics relating to highly personal information, topics relating to illegal activity, or topics that are anxiety provoking, etc.*	No
• Will anyone be taking part without giving their informed consent?	No
• Is it covert research? *Note: 'Covert research' refers to research that is conducted without the knowledge of participants.*	No
• Will the research output allow identification of any individual who has not given their express consent to be identified?	No

If you have answered **YES** to any of these questions you are **REQUIRED** to complete and submit a UREC 3 or UREC4). Your supervisor will advise. If you have answered **NO** to all these questions then proceed with this form (UREC 2).

General Details

Name of student	Callum Peter McLean

SHU email address	b8015292@my.shu.ac.uk
Course or qualification (student)	Bsc Computer Science for Games
Name of supervisor	Penny Collier
email address	cmspc2@exchange.shu.ac.uk
Title of proposed research	A prototype AR app to enhance learning of A-Level Biology
Proposed start date	14/10/20
Proposed end date	15/03/21
Background to the study and scientific rationale for undertaking it.	To gain an understanding of augmented reality in mobile devices and learn about educational apps.
Aims & research question(s)	• Research into A-Level biology and AR mobile app development • Create and test the app • Critically evaluate my progress, the final app, the feedback and my response to the feedback
Methods to be used for: 1.recruitment of participants, 2.data collection, 3. data analysis.	1. Contacting by email 2. The participants will fill in a survey 3. Questions with numerical answers will be plotted to a graph, others will be evaluated and split into their key points
Outline the nature of the data held, details of anonymisation, storage and disposal procedures as required.	The data collected will be the participants personal opinions on how useful my app is. I don't need any person information (like name, age address) of the participants unless they wish to know the outcome of the research. The collected data will all be held in one document on a computer while being analysed. Once the project is complete only the analysed data will be kept.

3. Research in Organisations

Question	Yes/No
• Will the research involve working with/within an organisation (e.g. school, business, charity, museum, government department, international agency, etc.)?	No

• If you answered YES to question 1, do you have granted access to conduct the research? *If YES, students please show evidence to your supervisor. PI should retain safely.*	
• If you answered NO to question 2, is it because: • you have not yet asked • you have asked and not yet received an answer • you have asked and been refused access. *Note: You will only be able to start the research when you have been granted access.*	No

4. Research with Products and Artefacts

Question	Yes/No
1. Will the research involve working with copyrighted documents, films, broadcasts, photographs, artworks, designs, products, programmes, databases, networks, processes, existing datasets or secure data?	No
2. If you answered YES to question 1, are the materials you intend to use in the public domain? *Notes: 'In the public domain' does not mean the same thing as 'publicly accessible'.* • *Information which is 'in the public domain' is no longer protected by copyright (i.e. copyright has either expired or been waived) and can be used without permission.* • *Information which is 'publicly accessible' (e.g. TV broadcasts, websites, artworks, newspapers) is available for anyone to consult/view. It is still protected by copyright even if there is no copyright notice. In UK law, copyright protection is automatic and does not require a copyright statement, although it is always good practice to provide one. It is necessary to check the terms and conditions of use to find out exactly how the material may be reused etc.* *If you answered YES to question 1, be aware that you may need to consider other ethics codes. For example, when conducting Internet research, consult the code of the Association of Internet Researchers; for educational research, consult the Code of Ethics of the British Educational Research Association.*	
3. If you answered NO to question 2, do you have explicit permission to use these materials as data? *If YES, please show evidence to your supervisor.*	
4. If you answered NO to question 3, is it because: A. you have not yet asked permission B. you have asked and not yet received and answer C. you have asked and been refused access. *Note You will only be able to start the research when you have been granted permission to use the specified material.*	**A/B/C**

Adherence to SHU policy and procedures

Personal statement	
I can confirm that: • I have read the Sheffield Hallam University Research Ethics Policy and Procedures • I agree to abide by its principles.	
Student	
Name: Callum Peter McLean	Date: 19/10/2020
Signature: CMcLean	
Supervisor or other person giving ethical sign-off	
I can confirm that completion of this form has not identified the need for ethical approval by the FREC or an NHS, Social Care or other external REC. The research will not commence until any approvals required under Sections 3 & 4 have been received and any necessary health and safety measures are in place.	
Name:	Date:
Signature:	
Additional Signature if required by course:	
Name:	Date:
Signature:	

Please ensure the following are included with this form if applicable, tick box to indicate:

	Yes	No	N/A
Research proposal if prepared previously	X		
Any recruitment materials (e.g. posters, letters, etc.)	X		
Participant information sheet	X		
Participant consent form	X		
Details of measures to be used (e.g. questionnaires, etc.)	X		
Outline interview schedule / focus group schedule			X
Debriefing materials			X
Health and Safety Project Safety Plan for Procedures			X

Appendix C - Figure List

Table 3.6: The completed in bug testing table

Appendix D - Participant Responses

Participant 1:

1. I think the app is teaching me the important components in the growth of plants and how their values cooperate with each other, like light levels, water levels, co2 levels.
2. I liked watching it grow.
3. Yeah this is the sort of thing in school which I would find pretty engaging.
4. Probably a mixture of both but things like me when I was in school
5. Yeah I think sciences can be taught very effectively using AR maybe above other subjects because you can do demonstrations like this. I'm not sure how you'd use it for maths and stuff like that.

Participant 2:

1. How the different levels of light, water and CO2 affect the growth rate of the plant, like so how fast it grows and how well it grows.
2. Moving the sliders to see how it affected the growth rate. Looking at the graph and being able to see what difference it makes.
3. I could see it being used at a lower school level just because it is quite simple
4. I mean I would prefer to use this because I learn through engaging with things more than just reading.
5. Yeah definitely. Maths might be a bit more difficult, but physics and chemistry would be really good with apps like this.

Participant 3:

1. I think it's trying to teach me about the factors which impact plant growth and how changing them affects the growth of the plant.
2. I think the AR app is very engaging, being able to move around and see it grow, move in and out without having to use the interface to do that.
3. Yes.
4. A mixture of the two, definitely. I think it would require development into a different direction in order for it to replace the book entirely.
5. Definitely yeah.

Participant 4:

1. How water, light and co2 levels affect plant growth
2. Sliding the sliders to change the variables
3. Yeah I think so
4. Both, but I'm old so I'm used to books
5. Yeah

Participant 5:

1. The relationship between the plants need for co2, light and water
2. I liked watching the plant grow
3. I can
4. A mixture of both
5. Certainly