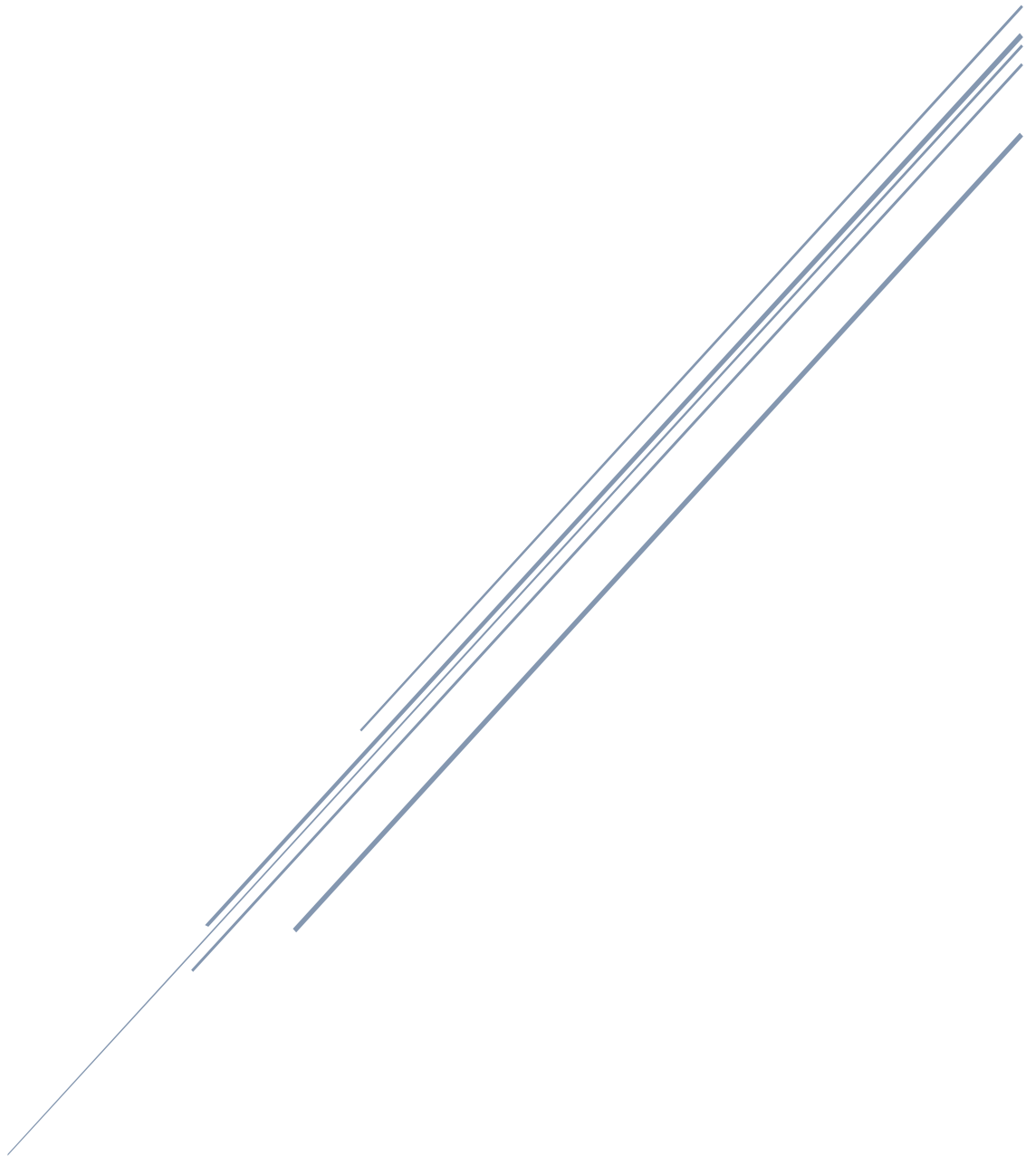


COMPUTER SCIENCE CHALLENGES DIARY

By Peter Davison

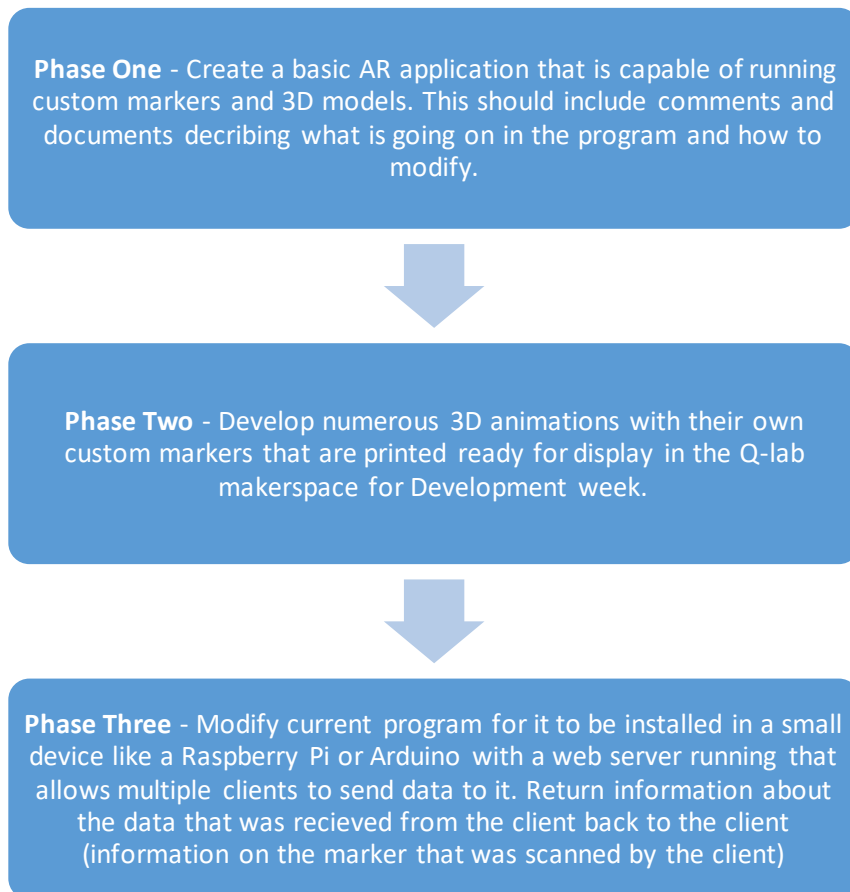


Queen's University Belfast

Contents

Planning.....	2
Week 6.....	Error! Bookmark not defined.
Daily Thoughts & Experience Diary	4
The Challenge	4
Week 1.....	4
Week 2.....	8
Week 3.....	10
Week 5 (Week 4 included).....	13
Week 6.....	17
Learning Log.....	22
Week 6.....	22

Planning



Phase One

The First phase of this project is to create a basic Augmented Reality application that will be able to have custom markers that people can scan on their mobile devices to display a 3D model. This phase is in place to create a foundation for the rest of the project. This phase also includes a lot of documentation development so when I submit my GitHub project then anybody can have a look at it and understand what is going on in a short space of time.

This can potentially help amateur programmers who are starting Augmented Reality for the first time, with the documentation clearly explaining what to do and what is happening at each line.

Phase Two

The Second phase of this project is to further develop the depth of the assets involved in the program, so when a custom marker is scanned then an animation, both 2 and 3D, will be displayed on the user's device. This can enable me to use my program in development week, where the visitors will be able to scan different markers around the QLab makerspace and be presented with 2D and 3D animations. This will be my back-up goal if I cannot successfully complete my final goal is. This is more realistic than the final goal I have set but I will come to that later on.

This phase can continue aiding amateur programmers who are starting Augmented Reality, but also people who already have an AR app and want to extend it by adding extra animations to possibly increase customer satisfaction.

Phase Three

The Third phase of this project is to modify the current program and assets for them to be installed onto smaller devices like a Raspberry Pi or an Arduino and also have a web server running on said device so that multiple users can connect to the web server. When they scan a marker, they will send a request for information about that marker to the server.

Then I need to enable the server to return information about the marker back to the client. This information will be in multiple different formats, including just a .png to an animation. For example, if there was a marker on a cupboard, the client can scan the marker and be shown the contents of the cupboard, or if the marker was on a computer then the client would be shown the specs of the computer. Because the main processing is done on the Raspberry Pi/Arduino, this decreases the amount of work the users mobile phone has to complete, potentially speeding up loading.

The final phase of the project will also be helpful to

This final goal is realistic; however, I will need to learn how to isolate the process of detecting the image as the mobile phone will still need to complete this. The web server on the server device (Raspberry Pi/Arduino) seems realistic as I have done quite a lot around web servers before.

Daily Thoughts & Experience Diary

The Challenge

To develop a Smart Contextual Application using Augmented Reality, for example, like a head-up display that provides people user-customisable relevant information as they work and live.

Week 1

- Through this week, I have focused on research and brainstorming ideas that my project could become. As this field is relatively new, it was difficult to find eligible products that I could research and learn off.
- Idea 1 – Smart Glasses with an interface that can use AR.js to help the user with their tasks e.g. Travel and Transport.

- Examples include:

- Vuzix

- Their product range spans from the M-series which are focused on the industry side of the market, and then the Vuzix Blade for the consumer market

- M-4000



- Helps the users perform tasks and increase productivity by showing tasks on the “See-Through” waveguide optics (A mini projector that shines onto a piece of glass)
- Contains very powerful processors for a 4k display
- M-400



- More robust product than the M-4000 due to its rugged design and is created to be ideal for users working in harsh environment
- Uses a small display on the inner side of the eyepiece
- Vuzix Blade



- Using a mini projector in the side of the glasses, the Vuzix Blade can project an AR interface onto a holographic lens for the user to see
- Similar to others, the Blade uses a haptic sensor on the side of the glasses for the user to control the interface with their fingers
- Although being for the consumer, the Blade did not fully take off partly due to its ugliness and "heavy duty" design

- Focals

- Made for the consumer



- Uses a joystick on a ring to control the AR interface



- Uses projector technology to show the AR interface

- HoloLens 2 by Microsoft



- Whilst the HoloLens 2 is a very advanced AR interface with many futuristic abilities, it is clearly made for the industry



- With many similarities to a Virtual Reality headset, the HoloLens will have been very expensive to research and produce, far beyond what I can solely achieve
- Overall, after looking at all of these examples, there is a slim chance I will be able to produce an AR interface with "holographic" capabilities, but therein lies the issue, the mini projector used to cast the interface onto a holographic lens is a piece of technology which I will not be able to get my hands on or produce a prototype, and therefore I created a new idea:
- Idea 2 - Smart glasses using a transparent OLED display to generate an AR interface for the user to assist with tasks
 - This would be very similar to the products above but a lot cheaper and lower quality for the AR interface
 - A YouTube called Jake Laser (Jlaservideo) did try to recreate E.D.I.T.H from the Marvel Universe doing exactly this, by adding a transparent OLED display to a pair of glasses and using text recognition software to pickup words shown to a webcam (also mounted onto the glasses) to show the definition of a word
 - To adapt to the display being so close to the eye, he added magnifying lenses in-front of the screen in order to adjust the focal length of his eye
 - I believe I can adjust this project and create a pair of Smart Glasses that enables the user to display directions on the display, translates a word or does mathematical equations, etc.
 - However, whilst looking for the transparent OLED displays needed, I struggled to find a satisfactory product, either they weren't available in the UK or I had to order them in bulk from a Chinese Retailer
 - The only business I could find that sold them at a reasonable price was SparkzFun (<https://www.sparkfun.com/products/15173>) which are based in America and priced the display with a breakout at \$39.95, but with the shipping included it put the price up to \$65.74 with UPS delivering in 2-4 business days
 - I would also need some sort of device to run the display, e.g. Arduino or Raspberry Pi, however these will create weight for the glasses frame to hold and also introduces more Health and Safety problems, e.g. What will happen if the device is

Wi-Fi enabled and is inches away from the head for extended amounts of time, or How much heat will the device produce etc.

- All-in-all this idea is a good idea however there are some issues with it, eg the price of components etc, so this may be a good backup idea
- Idea 3 - Augmented Reality Gesture-Based User Interface
 - This idea stems from the movie Minority Report (<https://www.imdb.com/title/tt0181689/>) in which there is a scene where Tom Cruise uses a gesture-based user interface to investigate a crime ([Minority Report 's gesture-based user interface](#))



- Going into more details for this concept, I would use a camera/multiple cameras to track anchor points that will be on the users hands, probably in the form of gloves as shown in the movie, to control and interact with an Augmented Reality desktop that is displayed onto a surface from a projector above/at an angle to it
- Idea 4 - Augmented Reality Graphical Display using Projection Mapping
 - Using a camera, the user will take a picture of a surface to project a display onto that will fit comfortably and sleekly into its surrounding environments, fitting uniformly on shelves or on the wall
 - The user will be able to customise the display and make their emails or messages be shown on the surface, other examples include a clock, a window to another place, or even just a piece of art
 - Although this may become very advanced, this may still be a good concept
- Idea 5 - Augmented Reality Maps Application for Mobile
 - So for this idea, the user will need a mobile phone with internet access and an inbuilt camera
 - Say the user is trying to reach a destination in a place they are not familiar with and use some sort of mapping service to receive directions to get to their destination
 - Using Augmented Reality, the user can point their camera where they think the direction is pointing and a graphic will come up on the screen, showing the user which road they will need to walk down
 - This has been created before ([Google Maps testing augmented reality](#)) by Google and other businesses



Week 2

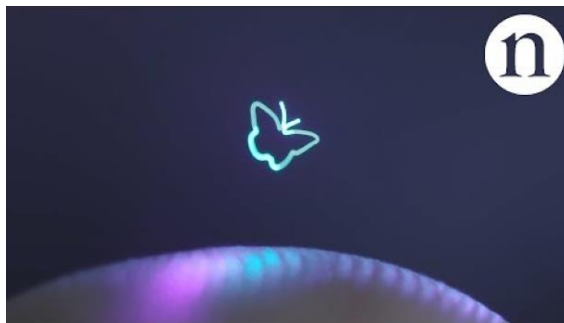
- Continuing my research and brainstorming this week, facing problems and thinking of solutions as I go
- Research
 - I want to focus on more of the fictional side of the AR field
 - Iron Man uses AR all the time, from the interface in his helmet to his interactive GUI in his lab



- However, obviously the technology that Stark is using is not available currently, although a few universities and businesses have been able to create 3D holograms that can be seen from all sides
 - [The world's first 3D Volumetric Display - Voxiebox VX1](#)



- Brigham Young University have developed a method to trap a particle and project light onto said particle from different angles to produce a "Volumetric image"
- This is majorly different to holographic screens and mirrors that have been used before because the particle can be seen from almost any angle
 - [Pictures in the air: 3D printing with light](#)



- Although it is obvious that I will not be able to use this with my project, it is however interesting to note that these fictional scenes won't be scenes for very long
- Spider-Man also gets an upgrade from Iron Man that uses AR
- I want to cut down potential projects to properly start on something, so I think through my ideas more logically and I have ended up with these
- Final idea 1 - Augmented Reality Graphical Display using Projection Mapping
 - Using a camera, the user will take a picture of a surface to project a display onto that will fit comfortably and sleekly into its surrounding environments, fitting uniformly on shelves or on the wall
 - The user will be able to customise the display and make their emails or messages be shown on the surface, other examples include a clock, a window to another place, or even just a piece of art
 - Although this may become very advanced, this may still be a good concept
 - Equipment
 - HD camera/Xbox Kinect for scanning surface
 - Projector to display
 - Libraries
 - PyQt for GUI
 - MATLAB Engine API

- https://uk.mathworks.com/help/matlab/matlab_external/call-matlab-functions-from-python.html
- Section that I would work on
 - Scanning part of the Projection mapping and discovering lines/edges for icons to "sit" on, make sure that they can be sleek
- Final Idea 2 - Augmented Reality Virtual Gesture-Based User Interface
 - Equipment
 - Either 360 or Xbox one Kinect with adapter to enable it to be plugged into a computer
 - Any VR Headset or HoloLens to display the interface
 - Potentially a RaspberryPi to run a server for controlling interface
 - Libraries
 - OpenKinect with Python for integration of the Kinect
 - https://openkinect.org/wiki/Main_Page
 - PyQt for GUI
 - <https://riverbankcomputing.com/software/pyqt/intro>
 - Section that I would work on
 - Making the headset and the Kinect compatible for interaction with the interface
- Final Idea 3 - Augmented Reality Physical Gesture-Based User Interface
 - I would use a camera/multiple cameras to track anchor points that will be on the users hands, probably in the form of gloves as shown in the movie, to control and interact with an Augmented Reality desktop that is displayed onto a surface from a projector above/at an angle to it
 - Equipment
 - Either 360 or Xbox one Kinect with adapter to enable it to be plugged into a computer/Multiple cameras with IR capabilities
 - Projector to display the interface
 - Potentially a RaspberryPi
 - Libraries
 - OpenKinect with Python for integration of the Kinect
 - PyQt for GUI
 - Section that I would work on
 - Creating a simple GUI that involves interaction with the Kinect/Cameras
- With more research, I realised that HoloLens has full eye and hand tracking already, so I won't be using HoloLens in my project unless it is for inspiration etc.

Week 3

- From the meeting I had with John Bustard (Course Leader), I have decided to use his feedback usefully and change my point of approach
This new approach will involve thinking of ideas that will be more useful for people more in the future e.g. there is a huge possibility that the big corporations will release a pair of smart glasses at the consumer level, excluding the corporations that already have release a pair
- Apple

- Apple AR glasses have been rumoured for the past two years and potentially been delayed for at least another year
- However, other sources state that Apple disbanded its AR team, which may mean they scrapped their plans for a pair of "iGlasses"
- Although the news is not confirmed, neither were the rumours in the first place
- After some further research however, a new patent was filed under Apples name, which may contain the rights to the future iGlasses, but production will not be started until much further down the line, looking at maybe a 2023 launch
- Android
 - Samsung
 - Samsung introduced their AR glasses at CES 2020 with an exercise demo using their GEMS (Gait Enhancing & Motivating System) exoskeleton device
 - In the demo, the actor showed a virtual personal trainer showing them what to do in their workout, whilst the Virtual PT altering the settings on the GEMS to increase intensity
 - However, these glasses were only part of a concept demo, so therefore Samsung don't have a name, price or release date yet
 - Lenovo
 - Although Lenovo showcased a monocular AR Smartglass at CES 2017, calling it the Lenovo New Glass C200, it seems that they are for businesses instead of the consumer as there is barely anything on them
 - Lenovo also released the Lenovo Mirage AR headset, which resembles the HoloLens a lot, but this seems to be focused on AR gaming as they partnered with Disney to create a Star Wars game solely for the Mirage
- Practical Research
 - As I will have to create a how to page on a selected portion of my area, I have decided to look for a guide to creating a quick and easy AR application for my phone
 - John recommended using a web framework that allowed me to connect using my phone, suggesting Flask with Python, and as I had already encountered Flask I knew how to create a simple web application just to test if it would work on the internet I now use in my new accommodation
 - This was the Python script that I created to run


```
from flask import Flask

app = Flask(__name__)

@app.route('/')
def index():
    return 'Hello world'

if __name__ == '__main__':
    app.run(debug=True, host='0.0.0.0')
```
 - However, my mobile phone wouldn't seem to connect
 - I expected as much as I did not have much control over the Firewall in the accommodation, and as my mobile phone was also connected to the internet, the Firewall would probably not allow the traffic to flow out then try to come back through
 - I then disabled debugging to try and let external devices access the web application

```

from flask import Flask

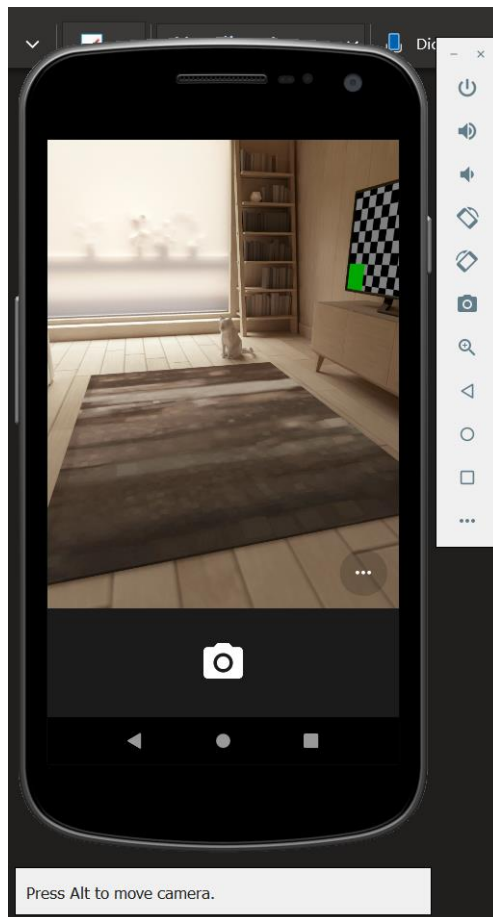
app = Flask(__name__)

@app.route('/')
def index():
    return 'Hello world'

if __name__ == '__main__':
    app.run(host='0.0.0.0')

```

- However this again did not work, probably due to the same problem
- I know this code works as I tried it both at the Computer Science Building and the Library (both the same network configurations probably) and could quickly connect using my phone
- So if I wanted to use this idea, using a web framework to host an AR web application, I would need to be on any network other than the one in which I live with, which would probably decrease productivity by a large amount, but it is still a possible idea for my project
- Another suggestion that John made was to use ARKit or ARCore to create an AR application, so I attempted to do so
- However, I quickly encountered a few problems
 - As I wanted to create an AR application for either the Apple glasses or the Google Glasses, I needed to use either one of their SDKs
 - I tried Apple first, as I have an iPhone so it would be easy to test the app, but to use their API, called AR-kit, I needed to code in swift
 - There were two problems with this
 1. I haven't programmed in Swift, but that could've been easily solved as it looked very similar to Java
 2. I needed a Mac or an iOS system to use Swift, which I didn't have as I have a Windows Laptop that I would be primarily using for my project
 - So my next option was to use ARCore with Android, as i couldn't use their iOS version because again it was in Swift
 - My first idea of testing the app with my phone couldn't be fulfilled because, again, I have an iPhone and couldn't program in Swift, which was my next problem
 - After hours of trying to get a Virtual Machine with iOS running, which I thought would be my solution to this problem, it didn't seem to work so I moved onto my next idea, using an Android Emulator
 - Installing the Android SDK was easy, taking less than 15 minutes, and it also came with the Android Emulator which was even better
 - The SDK came with a Hello World Application which when booted ran the app on a chosen virtual device, which I chose the Pixel
 - The build took quite a while (1 minute), however the second build only took 2 seconds but still slowed my laptop down a lot which will be a pain when using the emulator
 - The Emulator has a virtual environment that I will be able to use for AR testing



- Other than the emulator being very helpful, the actual Android Studio is very confusing so I will be using the following guide to learn Android development and the studio so that I can progress to creating an AR app
 - <https://developer.android.com/training/basics/firstapp/index.html>
- After learning about Android app development and what each file does in the project, I proceeded to learn how to create an AR app
- This consists of:
 1. Enabling ARCore
 2. Running AR apps in the Android Emulator
 3. Different Camera configurations
 4. Displaying 3D models in AR from Android apps or browsers
- There is an official guide on this from Google
 - <https://developers.google.com/ar/develop/java/quickstart>
 Which I will work through

Week 5 (Week 4 included)

Daily

Sunday 9th February

- Due to me being at home when Storm Ciara, I feel I won't be able to return to Belfast for a couple days, which will likely impact my work
- On Friday 7th I submitted my how-to guide, however regrettably I have just realised I completely forgot to submit the diary that went with it

- I feel that this was because I was rushing to submit my website and did have problems with preparing the .zip file with all the necessary images and the .css files for the website to be displayed properly
- I have included it below:

Main Diary

- *This week, I will be focusing on creating a how-to guide for beginners to Augmented Reality*
- *First, I will need to choose some software to develop the HTML file for the website*
 - *My first instinct was to just write the HTML file myself, but I quickly shot that down because I want the website to be look sleek and professional*
 - *I then headed to Wix, a website creator that allows you to create them for free*
 - *I chose with a template and started filling in sections, then remembered I needed to have an actual HTML file so I checked if Wix let you export the website to a HTML file*
 - *They didn't allow this, only letting you to publish the website straight away, so I continued searching*
 - *Adobe has a free online website editor called Adobe Spark, however this had the same problem as Wix with not letting me export the HTML file for myself*
 - *Adobe also has a more professional piece of software included with the Creative Cloud called Adobe Dreamweaver, however the Creative Cloud is a subscription based service which is very expensive so I got the 7-day free trial, as I was only going to use Dreamweaver for less than 7 days*
 - *Using the design I made on Wix that I really liked, I tried to replicate the same format in Dreamweaver, but as Dreamweaver is more for professionals it was hard to copy the drag-and-drop styled website I created on Wix*
 - *Using the existing templates on Dreamweaver and online forums and guides, I got the initial layout created in HTML with CSS files to define the different styles I will be using on my design*
 - *I wanted to use a selection of greys and whites as this is what I saw in common with most of the other Augmented Reality guides I was using for inspiration*
 - *I also wanted to use a selection of images to help the user visualise what I was explaining in each of the sections*
 - *The first section was planned to explain Augmented Reality and show the reader some of the products that are currently using Augmented Reality*
 - *The second is to explain the basics and the different types of AR to the reader so when it came to them planning their own application, they could choose what type of AR they wanted to develop quickly*
 - *I plan on talking them through some of the most popular types of AR:*
 - *Projection based AR*
 - *Used for the AR glasses Focals and more*
 - *From the name, uses a projector for the display*
 - *Recognition based AR*
 - *Camera recognises an object and presents information about it on the AR display*
 - *Location based AR*
 - *Uses sensors in the AR device*
 - *Outlining AR*
 - *Superimposition based AR*
 - *The third section is to suggest some of guides to actually programming an Augmented Reality Application and add a few sentences about what they're about and who they're for*

- *I already knew that both ARCore and ARKit have official guides to creating an AR app, so I will say a few things about them, and then hopefully find a few more guides for the reader*
- *After finishing the content of my website, I still had time before the deadline so I made a few tweaks on the design, and made sure the images I provided are of the highest quality I could find with the right licencing (Free to use commercially etc), however I did have to compromise for the more official product like the HoloLens as I couldn't find an image that would've gone well with the website*

Learning Log

- *Before using Dreamweaver, I didn't know much HTML, just the basics, but now I feel a lot more confident with creating a website with HTML*
- *Also, I was not aware of CSS files so this helps a lot with my goal of making my website look sleeker and more professional*
- *Creating the website also extended my knowledge of Augmented Reality, as there are more types of AR than I was aware of previously, like I didn't know that the art industry are using recognition-based AR to extend their creativity*

Plans for the project

- *After my website is complete, I will continue to work on the practical side of my project: programming a smart contextual application with AR*
- *From my research that I showed in my previous diary, big tech companies are rumoured to be releasing AR glasses to the consumers*
- *If I can create an application that is compatible for a pair of these glasses, then this would be very useful to other developers*
 - *I can help future developers with this project because I will combine multiple libraries or even make my own library to allow a developer to create a HUD for the chosen glasses*
 - *This will include creating assets and formatting an environment for the user to be able to customise and see all the content they want in front of their eyes*

Monday 10th February

- Due to me forgetting to submit my how-to guide diary, the Course instructor set me an email stating I will need to attend work sessions in the learning lab 2 times a week for 3 hours each to make sure I get a lot of work done
- I think this will be very helpful to me as sometimes I struggle to get started unless I'm in a helpful work environment
- I can also utilise the Makerspace equipment which will help me become more productive

Thursday 13th February

- Today, I want to use the time I have spare to try and get the Android SDK working with a basic AR app using Android Studio
- There is a tutorial with example files I can use on the ARCore website, so I will be following that guide

- It took way longer than expected because my laptop isn't the quickest, so each boot of the emulator took about 3-4 minutes, which soon adds up
- I finally got the app working after updating all the required libraries and I was able to place the Android Robot that they frequently use for showcases



To do list tasks

- ~~Get a simple AR application running~~

Weekly

Review prior to supervision meeting *(not all points are needed every week)*

- I feel like this week went only okay due to the fact that I was stranded away from University until Tuesday really impacted my productivity, but I dedicated a large chunk of time on Thursday to work on this and I successfully completed my task of being able to get a simple AR application running on the emulator
- Although I could've done more, I feel like the work I did do was very helpful and lays solid working ground for me to continue off

Summary of feedback and actions post supervision meeting

- I need to include a lot more of my planning, and plan more for the future instead of the present
- I have also been suggested to be able to extract an image from the AR app which includes the AR elements that can then be used for post image processing
 - This may be useful for security reasons, so if a camera picks up motion then can highlight the motion using outlining based AR
 - If the user wants to have more information on the motion that was detected, then the app can give them a selection of frames including the AR elements that can then potentially be used with other algorithms

Week 6

To-Do

- ~~Set up a web server with Flask~~
- ~~Create a barcode that will be scanned by a mobile phone~~
- ~~Create a HTML page for Flask with AR.js~~
 - Didn't end up working so I am now using github.io which is a lot easier as it is dedicated
- ~~Create a simple .obj 3D model in blender~~
- ~~Use github.io to run a web AR application that can scan a 'hiro' marker and show a prebuilt object~~
- Use github.io to run a web AR application that can scan a 'hiro' marker and show a customer object

Daily

Tuesday 18th February

- Today is the first day of my extra sessions in the makerspace lab.
- Even though it was busy, I felt very productive and got a lot done towards laying the foundations of the practical side of my project.
- I was able to use the iPhone X to test out AR.js application that had been built by others and also use the WebXR viewer which gave me an insight into what professional Augmented Reality apps had included in their GUI and also performance wise etc.
- At the beginning I thought I would be using Flask to host a web server for AR.js, which started off strong as I knew how to use Flask and how to add HTML code which needed to be run, so it was going smoothly.
- I created a QR code which when scanned by a users phone, took them to the Flask web server, which was running on 10.107.59.77:5000, and then they could point their camera at a 'hiro' marker which created an animation or 3D model on their phone, showing a simple AR application.
- However, as I started to modify the HTML code, an error came up on the phone saying that WebRTC was not installed on the browser.
- I was then suggested an idea which was to use github.io to host the website for AR.js, so I created a simple repository and used the same HTML code from the python file that was being used by Flask and was able to create a new QR code to take the user to my new github.io website, <https://peterdavison01.github.io/CSchallenges/>.
- After playing around with pre-made code, I started to customise it for my own AR app, but to do this, I needed a 3D model, so I downloaded Blender to create one.
- However, I have never used Blender before, so I will need to learn how to create a professional 3D model later on, but for now I will use a simple 3D cube

Wednesday 19th February

- So now I have my object, I need it to be display when the users phone recognises the marker.
- I have searched on the internet for a long time now and can't figure out where to import or reference the marker so that the phone can recognise it.
- I have resorted to using a AR.js developers code and deleting certain lines to see if I can find where the marker is referenced, even though it is highly inefficient, I need to work this out.
- So, AR.js have a few pre-set markers, but I will be using the "Hiro" marker, which looks like this:



Thursday 20th February

- Continuing my work from yesterday, I will be trying to display a basic 3D object I made on Blender. I was able to display a prebuilt 3D object when the phone scanned the 'hiro' marker, but the web application does not want to load.
- This may be because the app may not be able to load my raw github file of the basic 3D object, which is just a blank cube for now.
- So, the problem was because I had two <a-assets> tags which meant that the html file wouldn't work when it tried to load my object, but now my simple cube is displayed in the colour red.
- Now I have the 3D model working, I will move on to the advice I was given in the last meeting, to make the application capable of extracting an image which includes the AR elements.
- A-Frame has a built-in screenshot taker, but it assumes you're in Virtual Reality, not AR, on a computer which means the user would have access to a keyboard, therefore they have created keyboard shortcuts to take a screenshot.
- However, as the user will most likely be using a mobile phone, I need to stimulate the keyboard shortcut when the onscreen button is pressed. Therefore, I will need to integrate C# code into the HTML.
- I will also have to sort out the layers because the AR scene will be on top of the button, meaning you won't be able to see or interact with the button

Weekly

Review prior to supervision meeting

- I have done a lot of work this week around my project which is very good, and the sessions in the makerspace are proving to be very useful.
- As I am working in GitHub, I want to dedicate some of next weeks' time to creating a README.md and commenting my code to let a beginner understand what is going on.
- It also mentions in the spec that I will need to produce a social media post about my project, so I will have to think about how I am going to approach that and make a few notes in the upcoming diaries.

Summary of feedback and actions post supervision meeting

- Post the weekly meeting with John, I feel a lot more confident about the project because I have a "final destination" that I want to work towards
- The idea is that I install AR.js onto a small device like a Raspberry Pi, which also has a web server running that can have multiple devices connect to it simultaneously, that is fed an AR scene/screenshot from a device (like a mobile phone) when it is shown a custom marker and

returns information about what the scene/screenshot involves. I'm still a bit patchy about what the return information will be, but I can work that out.

Week 7

To-do

- ~~• Use github.io to run a web AR application that can scan a 'hiro' marker and show a custom object~~
- Comment code to help future devs understand
- Create a README.md file for GitHub page to explain what the project is all about
- Complete Planning section in this document
- ~~• Allow the AR app to be able to handle customer markers~~
- Create onscreen button for saving AR scene

Daily

Monday 24th February

- Something that was mentioned in the meeting was that I need to improve my Planning section of this document, so I will start the week by working on that
- I've been told that I need to talk about what the final goal is, how realistic it is, what are the steps I need to take to get there and if I cannot make my final goal, what is a backup goal that I can achieve.

Tuesday 25th February

- Today I have another lab session in the makerspace, which means I will get a lot of progress done on my project
- I will be putting the screenshot feature on hold for now because I want to focus on trying to be able to get the app to work with customer markers, so instead of using the basic Hiro marker, I can link any pattern file for the app to open and use.
- I will be doing this so that when Development Week comes around, I will be able to show the app with multiple custom markers around the lab that visitors can walk around and play with.
- However, when I add the custom marker into the code, every time the application recognises a marker whether it be the Hiro marker or my custom marker, the app freezes and doesn't work until I refresh the page leading the same problem to happen again.
- Due to there not being a console on my phone, I cannot get an error to work off, so I will try to work this issue out.

Thursday 27th February

- Continuing on with the problem I had yesterday, I have now got an error by opening my application on my laptop and using Chrome Inspector to get an error.
- It seems that the app cannot load the .patt file properly, which stops the app from fully recognising the marker.
- I finally fixed the error. It seems that there is a problem with AR.js' source code when it comes to custom markers therefore instead of using patternURL to link where the pattern file is, you have to use just url to link it.
- However, I encountered another problem whilst testing the problem above. Now the app reads the custom marker it creates an AR object, but when the camera doesn't see the

marker, the object should disappear, but it doesn't until the marker is seen again when it moves to the marker.

- The problem was again with confusing AR.js documentation and source code, the solution was to change a-marker-camera to a-marker and use an individual a-frame entity for the camera. Both these methods apparently create the same result, but the method is different.
- I will now be continuing my task of creating a screenshot button for the AR environment.
- This involves figuring out a way to place a button on top of the AR scene and giving it functionality. This will involve both HTML and JavaScript.
- However, currently it seems like GitHub are having some server-side issues which means I cannot edit my code currently. I will use this time to finish my Planning section in this document.
- The information I will return will probably be things like if the marker is on a cupboard then the client will see the contents of the cupboard or if the marker was on a machine then the client will see the specs of the machine.

Weekly

Review prior to supervision meeting

- Although I felt I wasn't as productive as last week, my mental state surrounding the project has massively improved because I now have a goal that I want to work towards.
- I was able to finish my section on custom markers and can now use custom markers that I want to use for different assets, may that include 2D or 3D animations and models.

Summary of feedback and actions post supervision meeting

- The supervision meeting was only to check up on how I was doing and give some advice on the planning section of my document, so therefore was quite short.
- I explained how I understood what I was doing and where I wanted to go with the project with my planning section. I was told that I could potentially merge Phase 3 & 4, as they are currently split, which was my initial idea, but I didn't want to make it too ambitious at the start.
- I was also told that I need to expand on who would be using my project on each phase.

Week 8

To-do

- Comment code to help future devs understand
- Create a README.md file for GitHub page to explain what the project is all about (same as the blog post)
- ~~Complete Planning section in this document~~
- Create onscreen button for saving AR scene
- Create blog post explaining my project

Daily

Tuesday 3rd March

- This week I will need to focus on my blog post and documenting my code as that is what I need to hand in by the end of the week.

- I should've gone into the labs today however I felt very ill and with the recent coronavirus outbreak I did not want to risk it, so I decided to stay at home and work on my laptop.
- To start off my blog post in html format I will need to use another html editor however the editor I used for my how-to guide was a free trial of Adobe Dreamweaver which has run out so I will have to find another one.
- I quickly found BlueGriffon which is an opensource alternative to Dreamweaver and is powered by Mozilla Firefox's rendering engine Gecko.
- I was able to polish up the blog post using the same format as I did for my How-to guide so a lot of the features are the same.

Learning Log

Week 6

- How to use to github.io
 - I feel this will be very useful for future projects as I can use github.io for a free dedicated web server, making it very easy to test my programs on devices other than the one I'm working on
- Basics of Blender
 - This will be very useful in future when dealing with 3D objects again