Computer Science Challenges - Augmented Reality with Smart Context

# Planning Diary

At the beginning of my project, I created a plan to guide me through my project and allow me to be the most efficient I could be. However, I have changed a lot of the project and its goals through the months of working on it, so this document is to summarise my project using the planning I did at the beginning, stating whether I completed an element of the project or whether I changed it and why.

# The Original Plan

In order to make my analysis of my original plan, I will be writing whether I succeeded or failed the specific objective in red. Below is the original plan I took from my Daily diary:

As you can see, I split my project up into 3 phases:

Phase One was 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. I did successfully develop a basic Augmented Reality application that was able to display a 3D model (found [here](https://peterdavison01.github.io/CSchallenges/)), however was not able to implement the custom markers feature. This still proves to be a big challenge for new AR.js programmers as the AR.js team haven’t specifically implemented a custom marker feature, so you have to do it yourself, which is difficult. This phase is in place to create a foundation for the rest of the project. It did also create a foundation to my project as I learnt how to use AR.js and also, because of the custom markers not working, it allowed me to use it as a test application that I could play around with to learn more about AR.js. 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. I did also include documentation for this application as I was trying to learn it so I was constantly writing notes on the code and commenting it saying what each part did. 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 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. Sadly, the development week was cancelled, however I did manage to use both 2D and 3D models but decided against animating them as they weren’t going to be useful for my final build, but only for the development week. 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. 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. Using both my final build and the tester application, amateur programmers could easily build an AR application using both 2D and 3D models, letting the user possibly choose what 3D object they would like to see by adapting my HTML forms to change the element of the a-entity geometry instead of the value so that the physical model would change and not just the text.

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. I was able to complete this objective as I used Flask on a Raspberry Pi to run AR.js on their local internet which then allowed the user to connect to with any device connected to their local network. 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. This objective was successfully completed, however in a slightly different way. Instead of the server sending a .png or an animation, the Raspberry Pi only sends String information about the cupboard or the PC. 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.

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. Before reaching this point, where I started implementation of isolating the process of the detecting the image, I changed my final goal to a Custom GUI. Implementing the work I had already done with returning information about a certain object, I created an application that allowed the user to custom the objects that the user was receiving. The user can now store a new object in their storage and customise it details. This meant that when the user scanned its corresponding marker, a 2D plane was displayed on the user’s device, showing the object’s information that the user customised. The final build of the project can be found [here](https://github.com/PeterDavison01/CSchallenges), where there is a full guide on the repository found in the README.md [here](https://github.com/PeterDavison01/CSchallenges/blob/master/README.md). The README.md is written in basic programming jargon, which allows any developer with any skill to pick up and use the project, creating a project of lasting value. Each program file is fully commented with basic comments to ensure that anybody could understand what was going on where and therefore how to use the code.