Context

**Here you give details of the development or investigation of the new material proposed in 'New Ideas'. This must be done in a business-like manner. The development of any software must follow a suitable analysis and design methodology. There are CASE tools available to you for some methodologies, others will have to be a 'paper' design. An investigation must also follow a suitable methodology and use appropriate techniques and tools.**

**Software-based projects, requiring the production of a software solution for a set of requirements, should demonstrate that the software development has undergone appropriate analysis, design, project management, structured programming and testing. An investigation must produce a technical outcome from some development (software or hardware (e.g. networks, displays)) or testing (e.g. of system/network performance, system security, HCI/usability analysis).**

**Some projects aim to provide software for general use as their final product and these must include relevant aspects of HCI (Human Computer Interaction) and address such features of usability such as 'user friendliness' and most likely employ GUI (graphical user interface) standards such as Windows.**

**In any case, students often ask what should go in this chapter, how to describe what they have done, what is relevant, how much of existing work to include, what to include from what they have done, etc. The simplest and surest way is to refer to your diary of the work you have done and report on it in chronological order.**

**The complete requirements analysis, problem analysis & design of software must be done rigorously and included in full in an appendix. Avoid cross-referencing it too often, thus causing the reader to keep flicking pages back and forth, rather reproduce sections that you wish to draw the reader's attention to. That is, highlight the parts that you found particularly difficult to implement and feel rather proud of having solved. Do not include lengthy descriptions of standard techniques or methodologies, simply state that 'such-and-such was designed using such-and-such technique (give a reference, not just 'SSADM' but 'SSADM [James 1996]' where the reference is a standard text on the technique!)' and highlight where you found shortcomings in the technique that didn't quite cope with your particular problem. Highlight exceptions to the standard.**

* Talk about the tools used for development (android, node, unity)
* As per leanUX produce a paper prototype of the application, showcase the alterations that have occurred from user testing
* What are the tools I will use
* Development diary?
* Highlight key areas of development (the game, fitness integration), hosting on Android market place)
* Check what is meant by this-

“The complete requirements analysis, problem analysis & design of software must be done rigorously”

# Introduction

The following chapter is focused around the development of the idea proposed in the Chapter Three: New Ideas. Within; a summary of the steps taken whilst following the Lean UX project methodology, the tools used for development are discussed with a focus on the reasoning behind their choice, a summary of key elements of the system that were developed with the aim of meeting the projects goals and finally a discussion around some of the challenges faced during the development of this project.

# Lean UX project methodology

This section is a summary of the steps taken by the development team to accurately follow the process of managing a project using the Lean UX project methodology. For a full writeup of the process and results of the individual steps please refer to **Appendix A: Lean UX**.

The first stage of Lean UX is to identify a problem with the current status quo, this is to say some situation where the goals of a target audience are not being fully met. In this case, the problem identified in Chapter 3: New Ideas, was that there are no solutions that offer a more passive approach to user fitness and entertainment, and that the uses of such a solution have yet to be fully explored.

Once a problem had been identified the project team conducted research into the user demographic of the proposed application, this step is conducted as it aids in empathizing with the target demographic, and helps to identify any bias or false assumptions the project team may have. These steps are all in aid of creating a better targeted solution for the users. The result of this stage is a list of assumptions the project team had about their target audience.

Once the previous stage was complete any assumptions the development team have decided upon were ranked in order of priority, the justification for this is to generate a testing schedule that focuses upon the highest priority items first. Priority is determined by a combination of two factors; knowledge and risk. Those assumptions where the team had little knowledge of the validity of the assumption but presented a high risk to the projects failing, if the assumption was wrong were ranked as the highest priority and so would need to be tested first.

With a prioritised list of assumptions, the next task was to develop a series of hypotheses, these are summaries of the assumptions discussed in the previous sections with the key difference that these hypotheses also contained a marker for their success. This allows the team to be able to quantifiably tell if the hypothesis in question is in fact true and valid. Below in figure 1 is an example of one of the hypotheses created as part of this exercise\*

“I believe that NHS staff need a way to keep track of the location of wheelchairs around a trust and that creating a system to do this will both save time and improve NHS staff satisfaction, I will be able to see if this is true by creating a prototype of the system and using on a small test group.”

Figure 1

*\*The format perspective of the hypotheses is in first person, this is in line with the guidelines of Lean UX (Add lean UX reference here)*

The last step in this process is the creation of minimum viable products that are created to test each individual hypothesis for validity. **Add information about the testing of hypotheses with the MVP’s**

# Development tools

The following section outlines the main tools used during the development of this project as well as the justifications for these choices.

## Android / Android studio

One of the objectives of this project is that the application should be designed to operate in on a mobile operating system, this is to allow for ease of access to the applications, as so long as the user has their mobile device with them they will be able to make use of the proposed application. There are several operating systems available that could be used to host the proposed application, however Android has been chosen as the most suitable for the following reasons;

* Android holds the largest market share of any mobile operating system, in Q2 2016 it saw a market share of 87.6% with Apple’s iOS holding 11.7%. Table 1 below shows the dominance of the android platform over other platforms between 2015 and 2016 (IDC: Smartphone OS market share, 2016)

Table 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Period** | **Android** | **iOS** | **Windows Phone** | **Others** |
| **2015Q3** | 84.3% | 13.4% | 1.8% | 0.5% |
| **2015Q4** | 79.6% | 18.6% | 1.2% | 0.5% |
| **2016Q1** | 83.4% | 15.4% | 0.8% | 0.4% |
| **2016Q2** | 87.6% | 11.7% | 0.4% | 0.3 |

* In 2015 Google announced Android had 1.4 billion active Android users (Callaham, 2015), with such a high user base the users of the proposed system will be more likely to know the android platform consequently making the user experience somewhat easier

### Justification for a single platform approach

With android set as the primary choice of operating system, the question did present itself as to what method of development would be used to produce the Android code, indeed there were several options available each offering different feature sets, some of the options present such as Cordova and HTML5 presented the opportunity to create an application that would work cross platform with Android, IOS and within a Web browser (Vensi, Inc., 2017) (Rajput, 2017). However, these options were passed up in favour of developing through Android Studio with Java for the following reasons;

* Google Fit is an open API that can be accessed using an android application, as the application proposed in the report intended to use fitness data to improve the users experience with the app the use of the Google Fit API was a crucial piece of functionality. The choice of Android Studio was in a large part influenced by this key functionality as access to the Google Fit API was not available through the presented other options as the trade-off of having multiplatform availability meant the lack of access to functionality native specifically to the Android platform
* Android Studio also provides the use of the Android Virtual Device manager (AVD), the AVD allows for the testing of an application on devices with multiple different configurations such as operating system version and screen size, as the proposed application is designed to appeal and work for a large audience, testing with these variations is another key reason Android Studio was chosen as the platform of development for this project.

## Node.JS/ Express/ couchDb

## Unity/ C#

# Key features

## The game

## The server

## Google Fit API

## Integration of all components

# Discussion

# Testing

# HCI considerations

# References

IDC: Smartphone OS market share (2016) Available at: http://www.idc.com/prodserv/smartphone-os-market-share.jsp (Accessed: 15 November 2016).

Callaham, J. (2015) Google says there are now 1.4 billion active Android devices worldwide. Available at: http://www.androidcentral.com/google-says-there-are-now-14-billion-active-android-devices-worldwide (Accessed: 14 November 2016).

Vensi, Inc. (2017). Benefits of Apache Cordova Cross Platform Mobile App Development. [online] Available at: http://www.vensi.com/benefits-of-phonegap-cross-platform-mobile-app-development/ [Accessed 23 Feb. 2017].

Rajput, M. (2017). Top 10 Reasons to Use HTML5 for Mobile App Development - Developer's Feed. [online] Developer's Feed. Available at: http://www.developersfeed.com/top-10-reasons-use-html5-mobile-app-development/ [Accessed 23 Feb. 2017].

# Bibliography