

Electronics and Computer Science
Faculty of Physical Sciences and Engineering
University of Southampton

Eamonn Trim
Date: 11/05/2020

Experiment In A Box (XB) Framework

Project supervisor: professor m.c. schraefel
Second examiner: Dr Nick Harris

A project progress report submitted for the award of
Electrical and Electronic Engineering

9840 Words

I Abstract

Experiments in a Box are five-day-long experiments that look at the effects of an activity on one of the In5 health heuristics. The current process requires the development of a new application each time a new box is created.

This report explains the work done to unify both users and researchers into one application (the Experiment in a Box Framework), that can run the Experiments in a Box. It also looks to enable researchers to quickly launch novel experiments. This removes the entry barrier for this research domain and allows for a larger volume of research projects to be produced in this field.

This Framework was created as an Android application. Looking at the SleepBetter Experiment in a Box as a basis, the Framework has implemented the same functionality by using free and available libraries for Android.

The Framework was successfully able to recreate the SleepBetter Experiment in a Box natively, however, some of the functionality is missing. It successfully implemented eight of nine specification points. This opens many avenues of future work, including a database restructure, an iOS deployment, and developing a physical device to work alongside the application.

II Statement of Originality

I have read and understood the ECS Academic Integrity information and the University's Academic Integrity Guidance for Students.

I am aware that failure to act following the Regulations Governing Academic Integrity may lead to the imposition of penalties which, for the most serious cases, may include termination of the program.

I consent to the University copying and distributing any or all of my work in any form and using third parties (who may be based outside the EU/EEA) to verify whether my work contains plagiarized material and for quality assurance purposes.

I have acknowledged all sources and identified any content taken from elsewhere.

The use of SleepBetter XB for comparison and images throughout this report was approved by George Muresan, the developer of the application.

I have used the following open-source, free-to-use libraries for my implementation of the project:

- Android Image Cropper by ArtherHub from GitHub [10]
- Graph View by jjoe64 from GitHub [11]
- Compact Calendar View by SundeepK from GitHub [12]
- Stitch-Android-SDK by mongodb from GitHub [13]

I did all the work myself and have not helped anyone else.

The material in the report is genuine, and I have included all my data and code.

I have not submitted any part of this work for another assessment.

My work did not involve human participants, their cells or data, or animals.

III Table of Contents

I ABSTRACT	I
II STATEMENT OF ORIGINALITY	III
III TABLE OF CONTENTS	V
IV TABLE OF FIGURES	IX
V ABBREVIATIONS	XI
VI ACKNOWLEDGEMENTS	XIII
1 INTRODUCTION	1
2 BACKGROUND RESEARCH	3
2.1 IN5 AND EXPERIMENTS IN A BOX.....	3
2.2 CURRENT “SLEEPBETTER” IMPLEMENTATION.....	3
2.3 CHALLENGES OF THE APPROACH.....	4
3 SPECIFICATION	5
3.1 INITIAL GOALS AND SPECIFICATIONS.....	5
3.1.1 <i>Goals</i>	5
3.1.2 <i>Application Specification</i>	6
4 PLANNING THE PROJECT	7
4.1 PLANNING FOR FIRST SPRINT.....	7
4.1.1 <i>Timeline</i>	7
4.1.2 <i>Budget</i>	7
4.1.3 <i>Risk Analysis</i>	7
4.1.4 <i>Tools</i>	7
4.2 REVISED PLANNING FOR SECOND SPRINT.....	8
5 IMPLEMENTATION	11
5.1 FIRST SPRINT.....	11
5.1.1 <i>Static Pages</i>	11
5.1.2 <i>Shop Page</i>	12
5.1.3 <i>Box and Experiment Page</i>	12
5.2 SECOND SPRINT.....	14
5.2.1 <i>Finishing the Box Creation Page</i>	14
5.2.2 <i>Experiment Information Generation</i>	15
5.2.3 <i>Local User Data</i>	16
5.2.4 <i>User Progress – Graphs</i>	17
5.2.5 <i>User Progress - Journal</i>	20
5.2.6 <i>Data Repository Using MongoDB</i>	23
5.2.7 <i>Overall GUI Theme</i>	24
6 TESTING	27
6.1 REPRODUCTION OF THE SLEEPBETTER XB.....	27
6.2 USER EXPERIENCE OF THE SLEEPBETTER XB.....	28
6.2.1 <i>Journal</i>	28
6.2.2 <i>Survey Response</i>	29
6.2.3 <i>Calendar</i>	30
6.2.4 <i>Graphs</i>	30
6.2.5 <i>Notifications</i>	31
6.2.6 <i>Login and Data Syncing</i>	31

7 ANALYSIS	33
7.1 REPRODUCTION OF THE SLEEPBETTER XB.....	33
7.2 USER EXPERIENCE OF THE SLEEPBETTER XB.....	33
7.2.1 <i>Journal</i>	33
7.2.2 <i>Survey Response</i>	33
7.2.3 <i>Calendar</i>	34
7.2.4 <i>Graphs</i>	34
7.2.5 <i>Notifications</i>	34
7.2.6 <i>Login and Data Sync</i>	34
8 EVALUATIONS AND FUTURE WORK.....	35
8.1 EVALUATION OF PLANNING.....	35
8.1.1 <i>Time Management</i>	35
8.1.2 <i>Risk Management</i>	35
8.1.3 <i>Tool Usage</i>	36
8.2 PROJECT EVALUATION AND FUTURE WORK.....	36
9 CONCLUSION.....	39
10 BIBLIOGRAPHY	41
11 APPENDIX	A
A SALTSHAKER DEVICE DESIGN.....	A
A.I ARTEFACT SPECIFICATION.....	A
A.II SALTSHAKER SCHEMATIC	B
B PERT CHART FROM THE PLANNING PHASE	C
C GANTT CHART	D
C.I GANTT CHART LEGEND	D
C.II GANTT CHART FOR THE FIRST HALF OF THE PROJECT (WEEK 1 – 15)	E
C.III GANTT CHART FOR THE SECOND HALF OF THE PROJECT (WEEK 16 – 33).....	F
D BILL OF MATERIALS.....	G
E RISK MATRIX FROM THE PLANNING PHASE.....	H
F GIT COMMIT LOG	I
G JSON DATA FILE (CONTAINING BOX AND EXPERIMENT DATA FOR FIGURE 7).....	J
G.I FIRST OF DATA SHOWN IN FIGURE 7.....	J
G.II SECOND SECTION OF THE DATA SHOWN IN FIGURE 7.....	K
H PROJECT ARCHIVE FILES TABLE OF CONTENTS	L
H.I FIRST SECTION OF ARCHIVE	L
H.II SECOND SECTION OF ARCHIVE	M
H.III THIRD SECTION OF ARCHIVE	N
I DATABASE SYNCING SCENARIOS.....	O

J CREATION OF SLEEPBETTER XB IN THE XB FRAMEWORK	P
J.I SLEEPBETTER BOX CREATION	P
J.II LIGHT – “INCREASE BRIGHT LIGHT EXPOSURE” EXPERIMENT CREATION	Q
J.III LIGHT – “BLUE LIGHT BLOCKER” EXPERIMENT CREATION	R
J.IV LIGHT – “TURN OFF ANY BRIGHT LIGHTS TWO HOURS BEFORE GOING TO BED” EXPERIMENT CREATION	S
J.V CAFFEINE – “DO NOT DRINK CAFFEINE WITHIN SIX HOURS OF GOING TO SLEEP” EXPERIMENT CREATION	T
J.VI CAFFEINE – “LIMIT CAFFEINE INTAKE” EXPERIMENT CREATION.....	U
J.VII CAFFEINE – “DO NOT DRINK CAFFEINE ON AN EMPTY STOMACH” EXPERIMENT CREATION	V
J.VIII SLEEP SCHEDULE – “GO TO SLEEP AND WAKE UP AT THE SAME TIME EVERY DAY” EXPERIMENT CREATION	W
J.IX SLEEP SCHEDULE – “SLEEP NO LESS THAN 7 HOURS PER NIGHT” EXPERIMENT CREATION.....	X
J.X SLEEP SCHEDULE – “DO NOT GO TO BED UNLESS YOU ARE RELAXED” EXPERIMENT CREATION PROCESS.....	Y
J.XI SLEEP SCHEDULE – “GO TO SLEEP AT 22:30 AT THE LATEST” EXPERIMENT CREATION.....	Z
J.XII DATA STRUCTURE PRODUCED (FIRST SECTION)	AA
J.XIII DATA STRUCTURE PRODUCED (SECOND SECTION)	BB
J.XIV DATA STRUCTURE PRODUCED (THIRD SECTION).....	CC
K PROJECT BRIEF	DD

IV Table of Figures

Figure 1: SleepBetter XB app features	4
Figure 2: Current XB process versus proposed <i>XB Framework</i>	5
Figure 3: A scan of the project checklist in the logbook as an example of its primary use	8
Figure 4: (Left) The home page, (Middle) About page, (Right) Login page	11
Figure 5: Mapping of <i>data structure</i> to GUI elements in the shop page.....	12
Figure 6: The <i>data structure</i> is converted to a Boxes GUI Object.....	13
Figure 7: An example of the mapping from the <i>data structure</i> to GUI elements for the Experiment Selection Page (raw data is shown in appendix E)	13
Figure 8: The mapping of the New Box GUI to the “Box” <i>data structure</i>	14
Figure 9: The mapping of the New Experiment Page GUI to the experiment <i>data structures</i>	15
Figure 10: (Left) The page flow for creating a new XB (Right) The expansion of “D” (the decision and variable change) in the page flow	16
Figure 11: The overview of the “Local User Data” <i>data structure</i>	16
Figure 12: The transaction of data when selecting an experiment (Changes in bold).....	17
Figure 13: Survey Response Page inputting new data into the “CurrentExperimentData” <i>data structure</i> (changes in bold).....	18
Figure 14: <i>Data structure</i> change to User Data on release from an experiment (changes in bold)	19
Figure 15: The three scenarios for data display in the Progress page (Left) No data available (Middle) No current experiment data available (Right) Current and archived experiments data available	20
Figure 16: The different scenarios that the edit journal entry pop-up can be displayed from.....	21
Figure 17: A demonstration of the journal selection feature (note that the 04/15/2020 entry is repeated) ..	22
Figure 18: The collection of user's data in the MongoDB Atlas computer client	23
Figure 19: An example of the changes between the old GUI style and the new GUI style (using New Box Creation page as the example)	24
Figure 20: Sleep Better XB recreation process into the <i>XB Framework</i>	27
Figure 21: Direct comparison between (Left) <i>XB Framework</i> and (Right) SleepBetter XB application.....	28
Figure 22: The comparison between (Left) <i>XB Framework</i> questionnaire and (Right) SleepBetter XB questionnaire.....	29
Figure 23: Comparison between (Left) The two calendars in the <i>XB Framework</i> and (Right) the SleepBetter XB calendar	30
Figure 24: Comparison between (Left) <i>XB Framework</i> Graph page, (Middle) SleepBetter XB Graph page (Right) SleepBetter XB Data analysis page.....	30
Figure 25: Notification for <i>XB Framework</i> (Left) and SleepBetter XB (Right) [Note: the colours are inverted to improve readability].....	31
Figure 26: (Left) Samsung Galaxy S8 <i>XB Framework</i> application (Right) the same user data on the Google Pixel 2 emulated application.....	31

V Abbreviations

Box	A collection of experiments
XB	Experiment in a Box (sometimes just experiment)
In5	InBodied 5
MEECS	Move, Eat, Engage, Cogitate and Sleep
GUI	Graphical User Interface
JSON	Java Script Object Notation
Data Structure	JSON Object (sometimes just structure)
Data Array	An array of data structures
IDE	Integrated Development Environment
Framework	Referring to the Experiment in a Box Framework application developed for this project

VI Acknowledgements

I would like to thank my supervisor, professor m.c. schraefel, for helping me focus my project and Dr Nick Harris for advice on how to compose the report.

I would also like to acknowledge George Muresan for his help in providing information about, and approval of use of, the current SleepBetter XB and its implementation. Also, thanks to Yue Shuen for access to her EatBetter XB Git repository.

1 Introduction

In America, health tracking is a large phenomenon, with roughly 70% of Americans tracking one or more health indicators. Of those, only 20% use a device/application to track this data. Also, 46% of people who tracked an aspect of their health had tracking change “their overall approach to maintaining their health” [1]. These statistics give a strong indication that the current climate in the western world is to try and be proactive about health to improve quality of life.

Beyond tracking, physical health indicators have a very real and direct link to the mental and overall wellbeing of a person. Over-sleeping, or under-sleeping, have both been linked to mental health issues and health risk behaviours amongst army veterans [2]. Physical activity has been shown to have a small benefit on sleep, showing the intrinsic link between different aspects of health and each other [3].

Social interaction has been linked to higher self-esteem [4]. A greater sense of community also helps to build mental fortitude. Also, meeting a minimum amount of time for physical activities has been shown to improve the quality of life of the population [5].

This information provides a strong rationale for research on personal health practices. Using the InBodied Interaction model, the body is described as 11 always-on, non-volitional systems [6]. These systems (such as the respiratory, digestive, or epidermal systems) are always adapting to the surroundings, and each has an impact on one another. The use of this model is to enable researchers to base activities to target one, or several, systems to improve overall wellbeing. By impacting these systems, the participant's overall health and wellbeing will also be affected.

On top of the InBodied Interaction model, the In5 model was introduced. This is a model that allows a methodology for interacting with the 11 non-volitional systems. The five points of interaction are Moving, Eating, Engaging, Cogitating and Sleeping; or MEECS.

The ability to interact with these In5 attributes is enabled through an Experiment in a Box (hereafter referred to as XB; explained in more depth in Section 2.1). An XB is a type of experiment, based around testing different actions and their effects on the In5. Due to the connected nature of In5 and the InBodied model, this will also have a systematic effect on the participant's overall wellbeing. It becomes a question, then, of what actions are taken within an XB affect the participant's overall wellbeing. This is explored within the XB model, and this experiment-model of interaction has already been used with two research projects emerging from the University of Southampton.

In the two currently released research projects (SleepBetter XB and EatBetter XB), the researchers have implemented similar features and yet have had to develop an independent android (or iOS) application for each. This requires both time and resources for the researcher, as well as expertise with technology to oversee properly. As a result, it limits the accessibility to this kind of research. This prevents the focus of the project being purely the results of the experiments, but instead, the focus becomes on how to develop the application to properly test the experiment. This means the time of production for an XB is extremely long.

This report focuses on the implementation of a Framework application for launching XBs, reducing the overall impact on researchers' time and attention, and therefore allowing for more research to be produced in this field. The challenges this project will tackle are producing a low storage impact application, with a dynamically produced Graphical User Interface (GUI) that allows for the fast deployment for new XBs from experiments. It also explores how to implement the user side of the XB so that the boxes can quickly attain new users, without requiring a new application launch. These involve answering the questions of what is important in an XB experience, and what is the information required for an XB to be launched.

2 Background Research

This section will discuss the current knowledge on the In5 model in depth. It will also explore the current implementation of research and the challenges that are associated with this approach.

2.1 In5 and Experiments in a Box

As previously explained, In5 is a method of describing areas from which to interact with the InBodied model. Each of the In5 processes (the MEECS) can be measured on a scale from too little, to too much [7]. The ideal case is to be somewhere in the middle of the range in the current environment. For example, someone could over-eat, or under-eat, neither of which are good for overall health, however, to be sated (somewhere in the middle) is the most desirable state. Keeping all the In5 within these desired equilibriums is not always possible, but has the most beneficial to health.

To ensure homeostasis in different environments, one must be aware of where these equilibria are, and when one deviates from said equilibria. Discovering where one is not over- (or under-) engaging with any one of these MEECS is the first step towards a better lifestyle with the In5 model. An Experiment in a Box is a package designed to support people in building personal health practices to help find and maintain these equilibria. An XB does this by testing out how well a given approach to improving an In5 works for the participant who is exploring a variety of heuristics over a short time. Participants are guided on how to record the effects of the intervention. The data collected by the participant is also anonymized and then sent to the experimenter. This methodology allows the participant to improve their health, whilst providing research data that can lead to conclusions on the health implications of certain lifestyle habits. To date, XB's have been created in individual applications (e.g. the SleepBetter XB developed by George Muresan from the University of Southampton, which explored different methodologies to try and improve sleep).

There are three main issues with this approach as it stands. The first being, to develop the application the researcher must be experienced in coding or employ someone who can create the desired application with the required functionality, a timely and resource-intensive process. The second is that the user base will have to be migrated each time a new XB is launched, which may reduce the number of participants. As a result, the statistical result of the research may not be as strong as it could be. Similarly, the last issue is that when launching a new XB in a new app, researchers must create a new workflow to intake and manage the data. In this respect, having a single application would allow a single researcher to produce many XBs. The entry barrier to this kind of research would be greatly lowered by creating a framework application, hopefully increasing the number of research projects in this field.

2.2 Current “SleepBetter” Implementation

The current SleepBetter XB developed by George Muresan, from the University of Southampton, will be examined for the current key features. The EatBox (another XB from the University of Southampton has also been examined but will not be analysed in this section). SleepBetter XB was launched in the summer of 2019. The XB is still in

active development at the time of writing. When investigating before the project began, all the user accounts and consent forms were handled outside of the application. These are now automatic and done on the first launch of the application.

The key feature of the box is that it holds different experiments. Figure 1 shows each integral page of the overall XB experience. The identified elements are an experiment selection per box, a data tracker, a journal/goal diary, and a calendar to be able to review progress. It should also be noted that there is a survey response section of the application (not displayed in Figure 1). The survey response is a questionnaire that generates data for the data tracker.

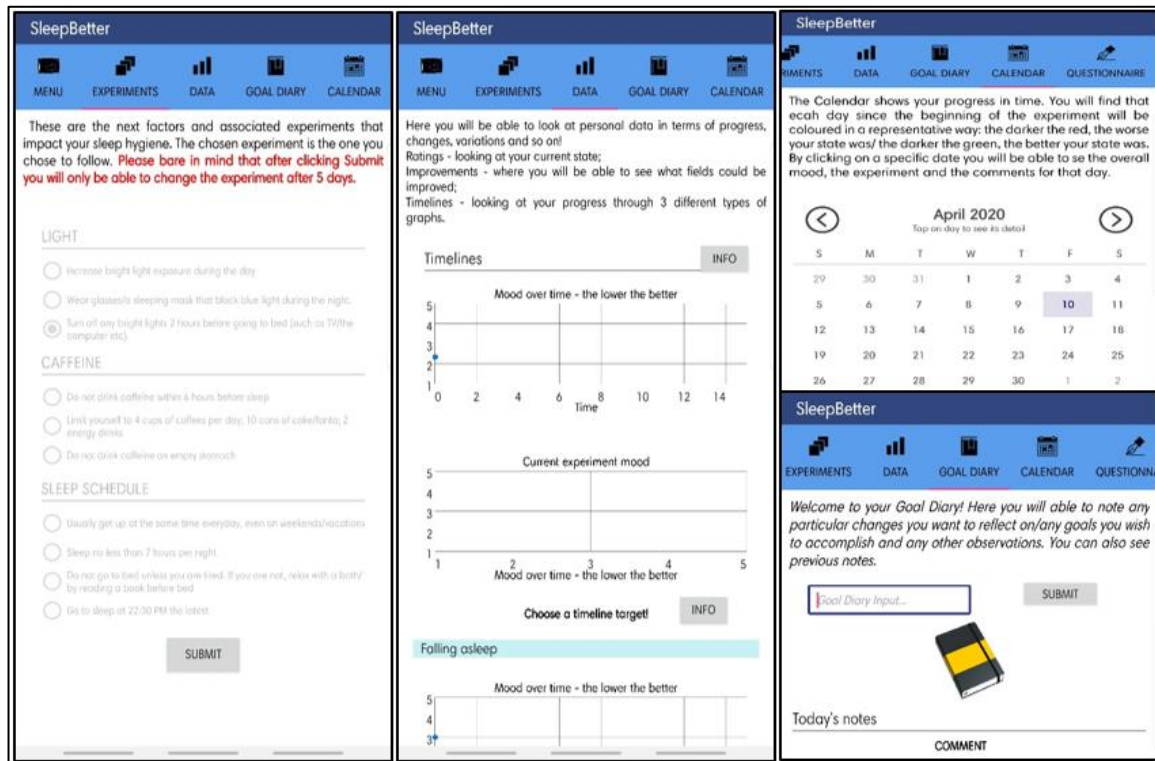


Figure 1: SleepBetter XB app features

2.3 Challenges of the Approach

In building an application that can produce a feature-full experience such as SleepBetter XB, yet is not entirely dedicated to just SleepBetter XB, there are some challenges to explore. Not first is the storage impact of the application. Whilst SleepBetter XB is an entirely dedicated application, it's storage impact can be particularly low. The challenge then becomes how is it possible to scale the application to enable more XBs data sets, without growing the storage impact of the application.

Another key challenge to consider is that the GUI of the SleepBetter XB, whilst quite particular, can also be defined statically as the list of experiments being shown is defined (as well as the graphs, calendars, etc). The challenge becomes being able to dynamically implement the GUI without overloading the application with predefined elements.

This report will explore both the challenges listed above, to build a low-impact, modular experience that allows researchers to quickly launch Experiments in a Box.

3 Specification

The following section describes the initial goals and specifications of the project. The latter part explains how this was refined with guidance, as the initial scope was far too large for the time frame.

3.1 Initial Goals and Specifications

This section focuses on the goals and specifications produced at the beginning of the project.

3.1.1 Goals

The end goal of this project is to develop an application that acts as a single point of interaction for both experimenters and those keen on improving their health. The final goal is to allow a researcher with no coding knowledge to be able to produce a fully functioning XB. The application should have a socket-like design, to enable a download of “data-packs” that enable different XBs, allowing the application to have a small storage impact.

Initially, a stretch goal was to produce a physical artefact as a proof of concept for general use devices that could enhance the experience of the experimentation (collecting data without thinking). An idea for a saltshaker that collects photos of the food you are eating was an idea designed for this project.

During the process of discussing the project, it was advised to either put the focus onto the artefact or not develop it at all, as the time frame did not allow for both. As a result, the software took focus. All the work that had been done for the artefact is included in appendix A. The subject of the rest of the report will be the software development of the *XB Framework*.

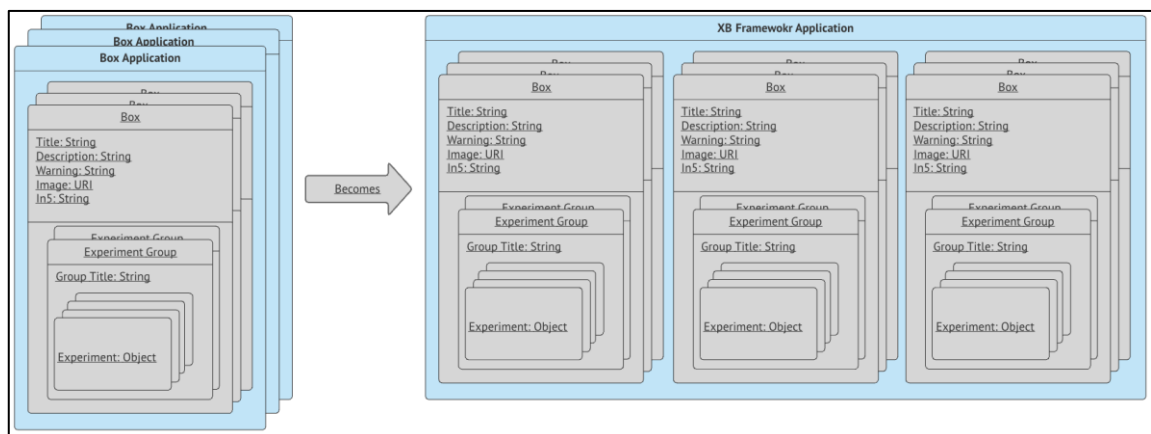


Figure 2: Current XB process versus proposed *XB Framework*

Figure 2 shows the current process for creating XBs. The *data structures* are shown as just a rough idea of the implementations. Both applications will have peripheral functionality around the inclusions of the boxes, however, the figure demonstrates that the change proposed it to move boxes from their applications into a shared application that hosts all boxes.

3.1.2 Application Specification

From the research into the SleepBetter XB in Section 2.2, there were several key features to an XB that need to be included. These key features are:

- *Displaying a box/experiment with:*
 - 1. A calendar that displays interaction/progress
 - 2. A progress page with graphs
 - 3. A journal/diary entry for user comments
 - 4. Notifications for the experiment
 - 5. A survey response page with the questions associated with that experiment
- *Be able to produce a box package from within the application with:*
 - 6. A choice of one, or more, experiments within the box
 - 7. Instructions / Guidance for the experiments within the box
 - 8. A custom survey to be completed daily for each experiment individually
 - 9. Customizable notifications for each experiment

The project also looks to include some of the more complicated features of some boxes, including image capture and processing for data. This, however, is not a core functionality of a box and will be held as a stretch goal, due to the complexity of implementing a customizable image processing algorithm in addition to the rest of the application.

4 Planning the Project

This section focuses on the plan devised at the beginning of the project, including the timeline, the budget, the risk assessment, and the tools. It should be noted the project was planned around doing both the artefact and the software. As discussed in Section 3.1.1, there was not enough time to complete both. The revised planning explains the changes made when the artefact fell out of scope. The project was split using the 12-week mark to distinguish between the first sprint and the second sprint (this coincided with the Christmas break).

4.1 Planning for First Sprint

4.1.1 Timeline

The timeline of the project is close to eight months long, starting on the 30th of September 2019, and lasting until the 15th of May 2020. At the beginning of the project, both a program evaluation review technique (PERT) chart (shown in appendix B) and a Gantt chart were created (in appendix C).

These tools were useful in putting an estimate on the timeline of the project, however, being inexperienced with such tools, the maintenance did not last the whole project. This meant that the use these tools are known for was not used to their potential. Deadlines did not often follow the Gantt chart, with some being completed quicker, and other beings finished much slower.

4.1.2 Budget

The budget was mostly based around the undeveloped artefact. Some of the components were even ordered before the project was revised. The full bill of materials can be seen in appendix D. In terms of the software, all the libraries intended to be used were free and available, and the software to use was also going to be free. The total cost for the software was £0 and the total cost of the hardware was £14.59 at the time of publication.

4.1.3 Risk Analysis

Risk analysis was done in addition to timeline planning. This allowed for contingency planning for certain foreseen difficulties. The full risk matrix can be seen in appendix E (this was produced at the beginning of the project).

This grid, much like the Gantt chart, was not used as much as it could have been. With the project evolving it would have been useful to update the risk matrix with new information, such as how to overcome previously faced challenges, or perhaps predicting new risks as the project developed.

Alongside the risk matrix, a thorough investigation into patents was performed. The closest design was [8], a “mobile device that monitors an individual's activities, behaviours, habits, or health parameters”. This was distant enough not to be a problem.

4.1.4 Tools

The application will be built for the Android firmware. This decision was made due to the native support for Windows development, as well as the overwhelming majority of the

market share that Android has¹. Whilst there are several IDEs (and languages) that allow development for both iOS and Android simultaneously, Android Studio IDE was chosen due to its greater library support, and a more established community for debugging. At the time of starting the project, Android Studio only allowed development in Java or Kotlin (an Android-specific coding language), being more familiar with Java, this is what was chosen.

4.2 Revised Planning for Second Sprint

Just before the Christmas break, a method to log daily progress was advised. This begun with including certain points such as which goal it was helping to improve, or the estimated amount of time until completion. This quickly became more laborious than useful. A better format found was a physical logbook, which was used to keep track of daily progress and to be an outlet for running thoughts on the project starting with the second sprint. This became the most effective tool for the project and a full scan of the logbook can be found in the project archive (directory map in appendix H).

It was just after the Christmas break that it was advised that either software or hardware became the focus of the project. The software was chosen because it was already under development and the hardware would still require the majority of the software to be developed.

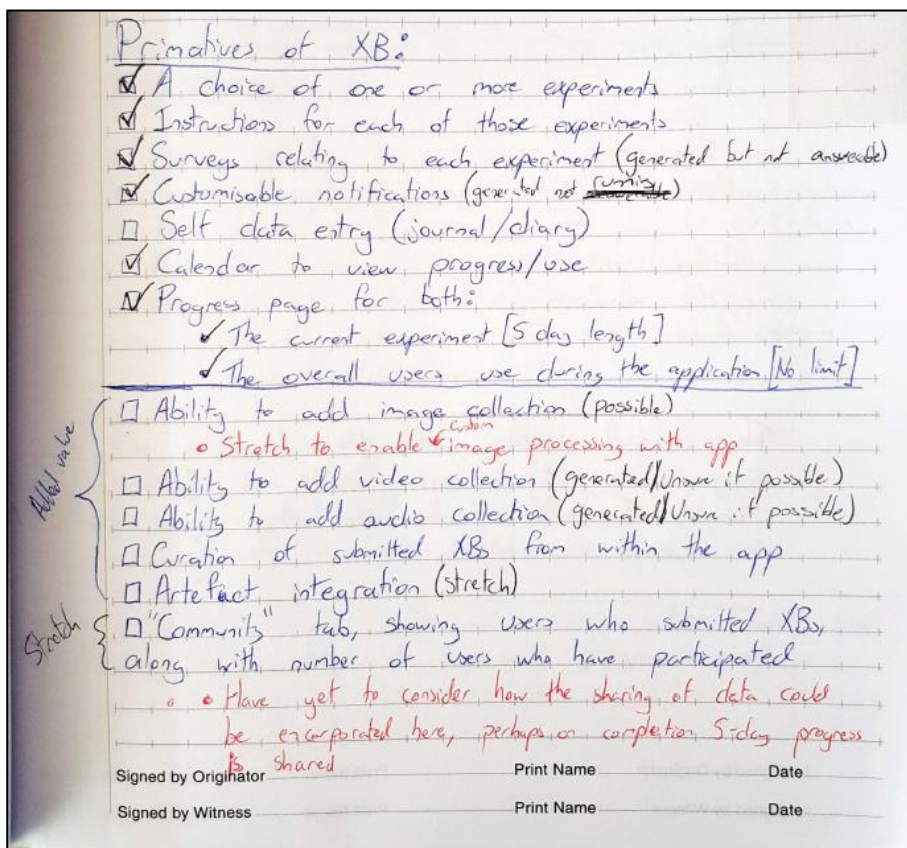


Figure 3: A scan of the project checklist in the logbook as an example of its primary use

¹ Statistics taken from [9], based on data released in 2018.

The software had been slow-moving until this point due to much of the focus being on researching how to implement both the software and hardware harmoniously. It was also difficult to adjust to managing the project independently. It was during this restructure that the code was moved to Git (a full list of commits can be seen in appendix F). There was also a shift in development style. Rather than a traditional waterfall development process, the project took a hybrid approach between agile development and waterfall development. This meant that each page of the application was implemented, then tested, then iterated until the implementation was fully complete. It was only when the results were verifiably correct for all foreseeable use cases that the development then moved onto the next page. This, in integration with the git, enabled for much faster, more organized, development.

One of the most effective planning tools was to write all the specification points down in the logbook. After doing this, treating that page as a checklist was extremely beneficial, as each point could be expanded and then the checklist would work like a cascade from one into another. This did not indicate the time frame that each implementation would take. A revision of work done, and work left to do, by referring to this list was much more intuitive. This checklist can be seen in Figure 3.

Evaluation of this plan and the overall success of the project is discussed in Section 8.

5 Implementation

The following section includes the technical progress achieved during the project. Many of the features and design process will be explained in this section. An assumption is made that the reader is familiar with the concepts from the introduction and background research. New concepts introduced in this section will be explained.

The section is split into the first sprint, and then the second sprint sections of the project. Each sprint is also then sub-divided up into page implementations, which are explained in the order they were developed.

It should be noted for understanding in this section, all the data is held in the JavaScript Object Notation (JSON) format. To prevent using this jargon repeatedly, and to ease the reading process, these JSON Objects and JSON Arrays will be referred to as *data structures*, and *data arrays* respectively.

5.1 First Sprint

This section focuses on the development in the first sprint, before the Christmas break.

5.1.1 Static Pages

During the first sprint, templates for all the pages (or activities as they are known in Android development) that will be used in the application were created. This meant that most pages (especially those which would be generated by data structures) were blank. However, at this point, all the statically defined GUI had been created, therefore all that was left to be implemented were the user interactions and the dynamically generated GUI. Figure 4 shows the only pages in the application that are completely statically defined.

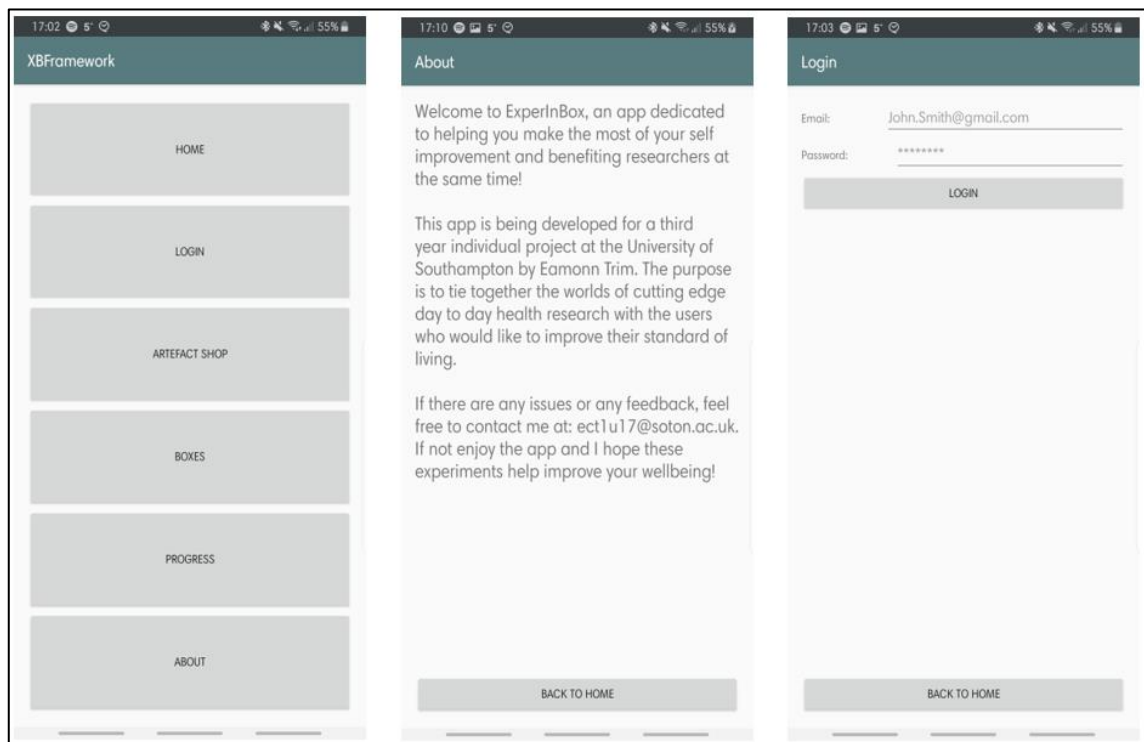


Figure 4: (Left) The home page, (Middle) About page, (Right) Login page

5.1.2 Shop Page

The first page that uses a *data structure* to be developed was the shop page. This *data structure* is entirely local to the application. It holds the mock data for artefacts, as none are developed yet. The implementation of this page does nothing but display the information, upon clicking an entity, the page simply displays a message explaining that the device is not currently available for purchase. The *data structure* and the page generated are shown in Figure 5.

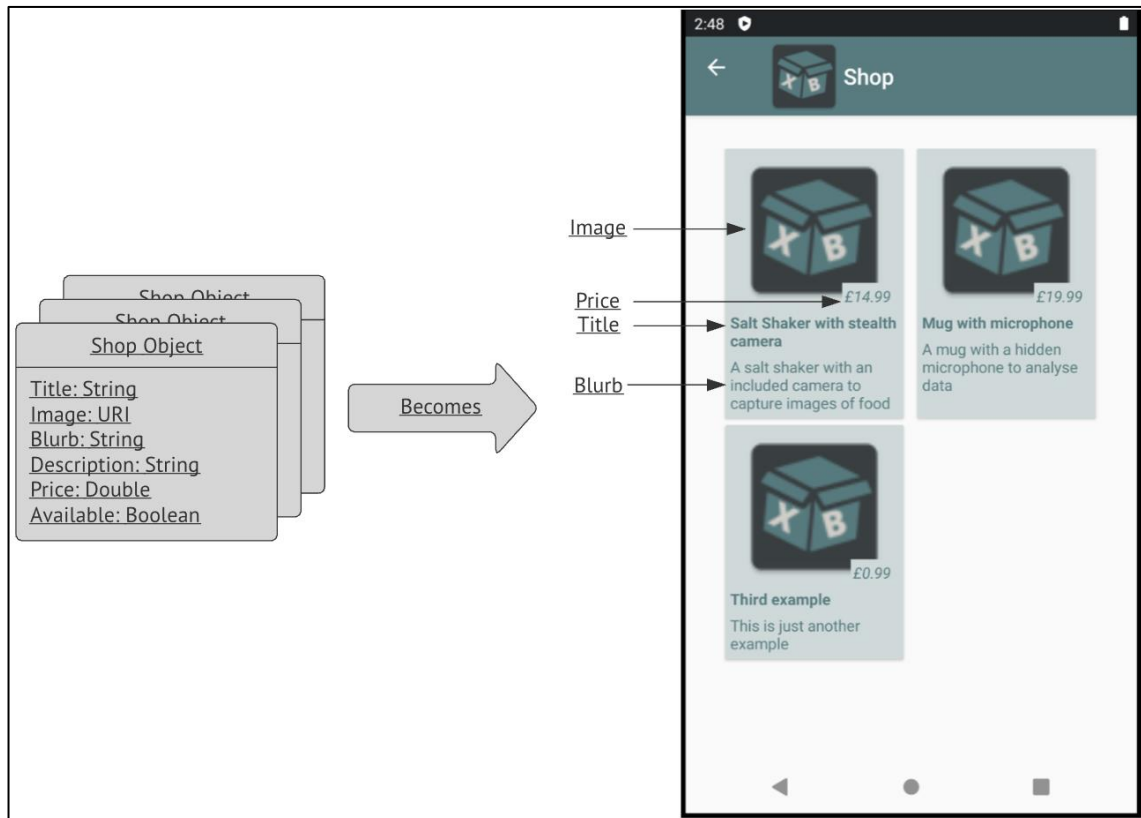


Figure 5: Mapping of *data structure* to GUI elements in the shop page

5.1.3 Box and Experiment Page

During this sprint, the first iteration of the “boxes” GUI was developed. It takes a *data structure* and converts it to a list of available XBs. Figure 6 shows the *data structure*, and how it maps to GUI elements on the boxes page. It should be noted that the only data in the example is the SleepBox, which is why the rest of the image displays a “No box available” under each of the other MEECS. The key points to note are that only the relevant aspects of the *data structure* are generated and displayed. The only three important points from the box in this position are the Title, Description, and Image.

It was important to keep the structure in its entirety because when clicking on an active box, a new page is opened. This page is given the entire “Box” *data structure*, to then use it to generate the relevant information. Within each of these “Box” *data structures*, there is a *data array* known as Experiment Groups. This *data array* holds the *data structures* for the included experiments. The associated experiments to the SleepBetter Box are shown in Figure 7. Only the experiment titles are used on this page and during the first sprint. Upon clicking submit, the selected experiment would be printed to the console, but no functionality was implemented yet.

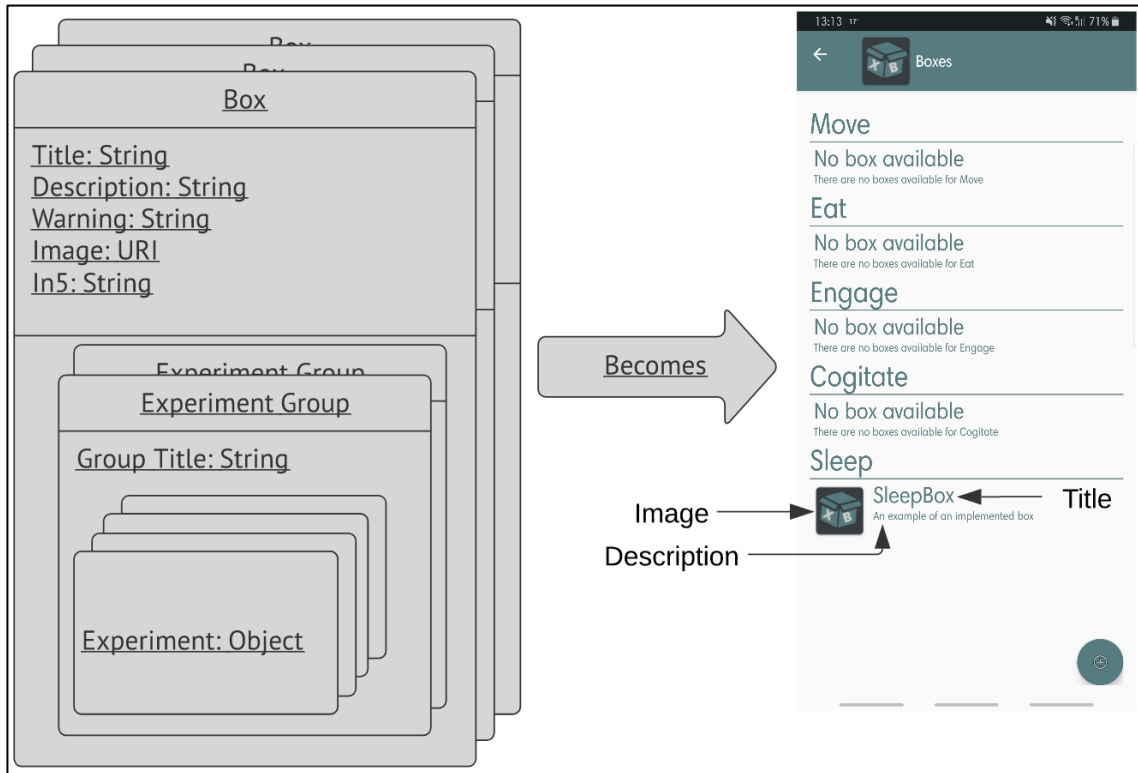


Figure 6: The *data structure* is converted to a Boxes GUI Object

After creating the basis for displaying the *data structures* within the application, it seemed relevant to also be able to create these *data structures* from within the application. This is the functionality a prospective researcher would use to launch an XB. As such, at the end of this sprint, the New Box Creation functionality was under development. It had reached a point where the GUI was being dynamically created and destroyed based on button clicks, but much of the backend data generation and deletion were not implemented yet. Section 5.2.1 describes the process of finishing this feature.

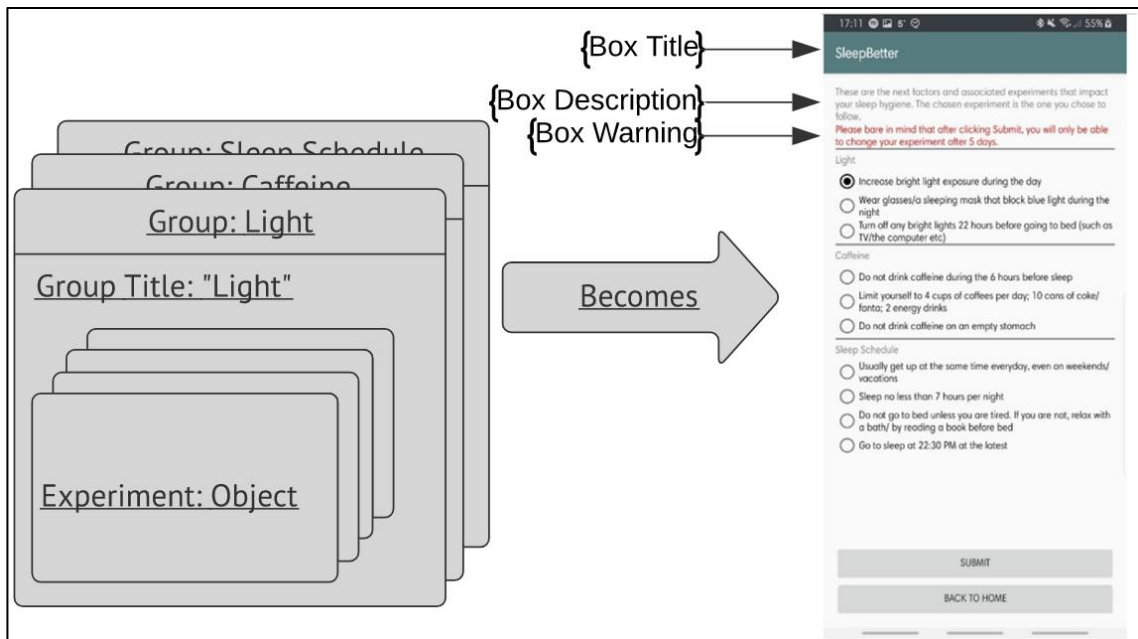


Figure 7: An example of the mapping from the *data structure* to GUI elements for the Experiment Selection Page (raw data is shown in appendix E)

5.2 Second Sprint

The second sprint took the work done in the first sprint and extended it further. This section describes the technical implementation in chronological order of implementation. It will also explore some of the design decisions that were considered.

5.2.1 Finishing the Box Creation Page

The creation of a new box takes the information a user inputs into a GUI and generates the “Box” *data structure* (and all its sub- *data arrays*). Figure 8 shows how the different GUI objects are mapped to the different elements in the *data structure*.

The indexing of GUI objects to back-end variables was the most difficult aspect of this feature. Each GUI object that could be added or removed dynamically (this is every element below Experiments in the Box title) required a global variable to know what the index of the array was. The Group Text was the easiest, as this only required a single dimension array. However, the Experiment Title, Experiment Description, and Experiment Instructions arrays were all two dimensional, with the first-dimension mapping to the Experiment Group to which it belonged, and the second being the index of the experiment within that group. It was important to keep track of these indexes because if an experiment was removed, it was important to remove the correct backend data, rather than just an arbitrary index, and re-display the correct numbers for the experiment indexes on screen.

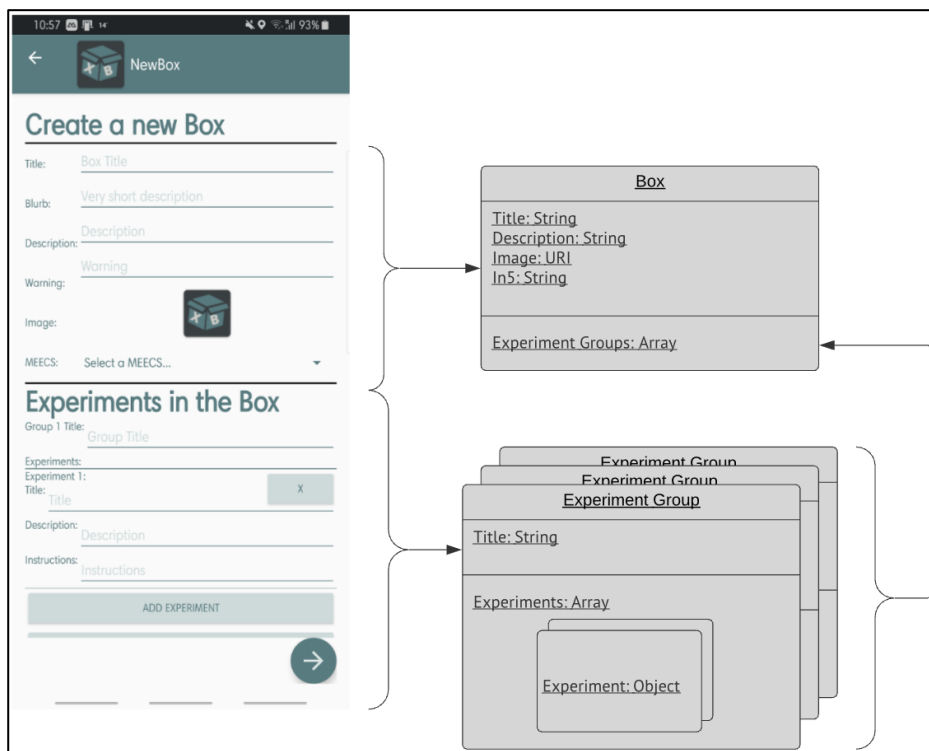


Figure 8: The mapping of the New Box GUI to the “Box” *data structure*

This generates the three high-level data labels for the experiments (Title, Description, and Instructions). This is not enough data for a user to participate in the XB activity. As a result, on submitting the data (by clicking the arrow in the bottom right corner), a new

page was opened to complete the *data structure* for each experiment. The Experiment Creation Page is explained in Section 5.2.2.

5.2.2 Experiment Information Generation

This page is used to complete the *data structures* associated with each experiment. This data populates an object *within* the Experiment Group *data structure*, which itself is held within the “Box” *data structure*. Each property (if enabled) must be filled out before the user can continue onto the next page. The mapping of GUI elements onto *data structures* can be seen in Figure 9.

The information generated on this page is for an individual experiment. Each experiment can either enable or disable their custom survey (this is locked to enabled for this version as no general box survey has been implemented). The experiment can have any number of questions within this survey, which will have responses from 1-5 (corresponding to strongly disagree, to strongly agree). Each experiment can then set a specific time range for the survey to be responded between, otherwise, it can be answered at any time. Each experiment also can have any number of custom notifications that run repeatedly at intervals of anywhere from one minute to one day. Also, the data for enable image, audio, and/or video data collection are generated from sliders towards the bottom of the page, along with a slider to enable data generated from artefacts (these are all locked to false in the application as their implementation user side has not been created, but for demonstration purposes in Figure 9 Image Upload is set to true).

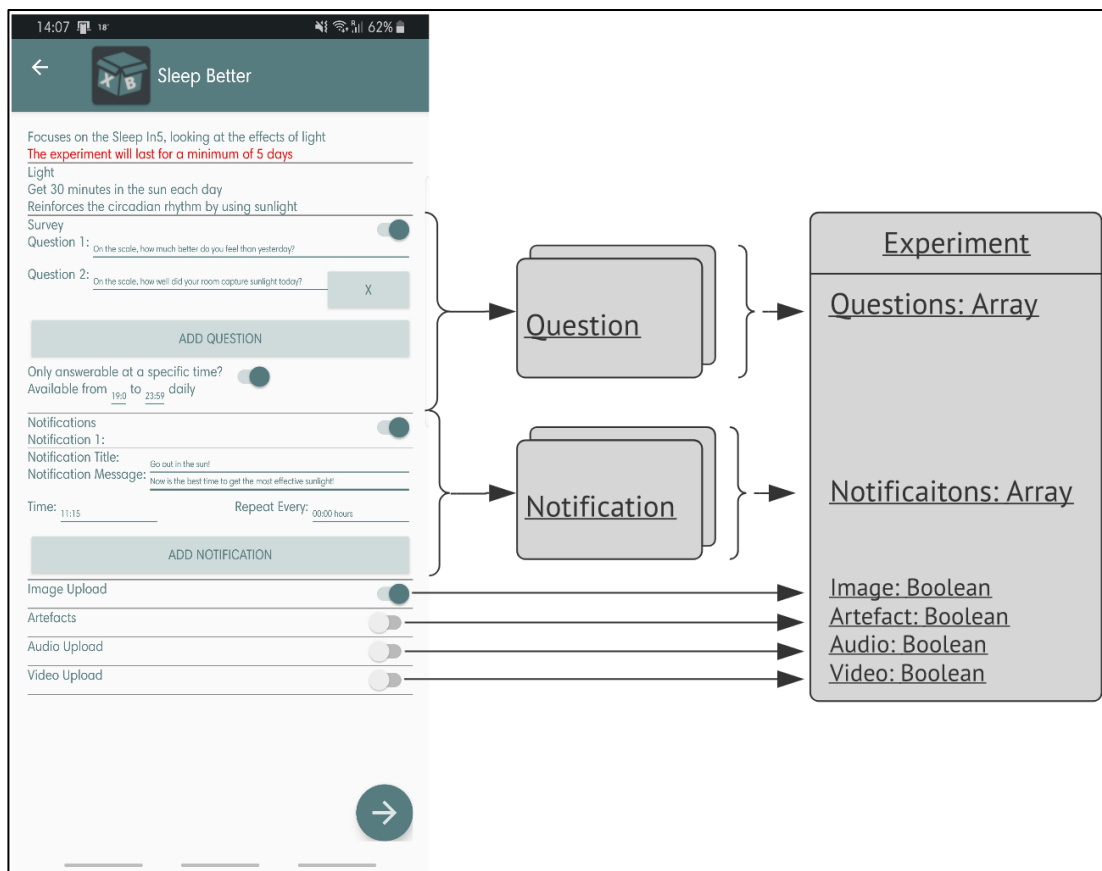


Figure 9: The mapping of the New Experiment Page GUI to the experiment *data structures*

A design decision was made to re-launch the page for each experiment, rather than to have a scrollable design holding all the information on one page. This is to enable the user to better distinguish between each experiment, and not get lost in an extremely long screen. To do this, the page must be re-opened repeatedly. As a result, three variables were passed into the page. These were the current group index, the current experiment index, and the entire *data structure* (including its updates from previous iterations of this page). At the end of the page, the relevant indexes were incremented, and if the *data structure* had not been completed, passed back into the page to relaunch. This allowed the same code to be reused for each index and meant that the boxes could be any scale, yet still manageable. The page flow (along with variable changes during each interaction) is shown in Figure 10.

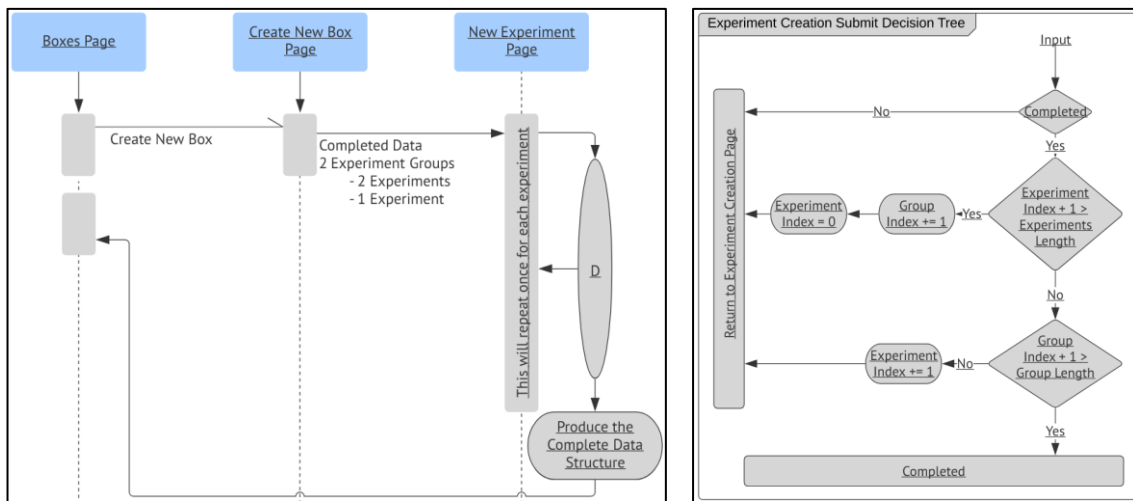


Figure 10: (Left) The page flow for creating a new XB (Right) The expansion of “D” (the decision and variable change) in the page flow

5.2.3 Local User Data

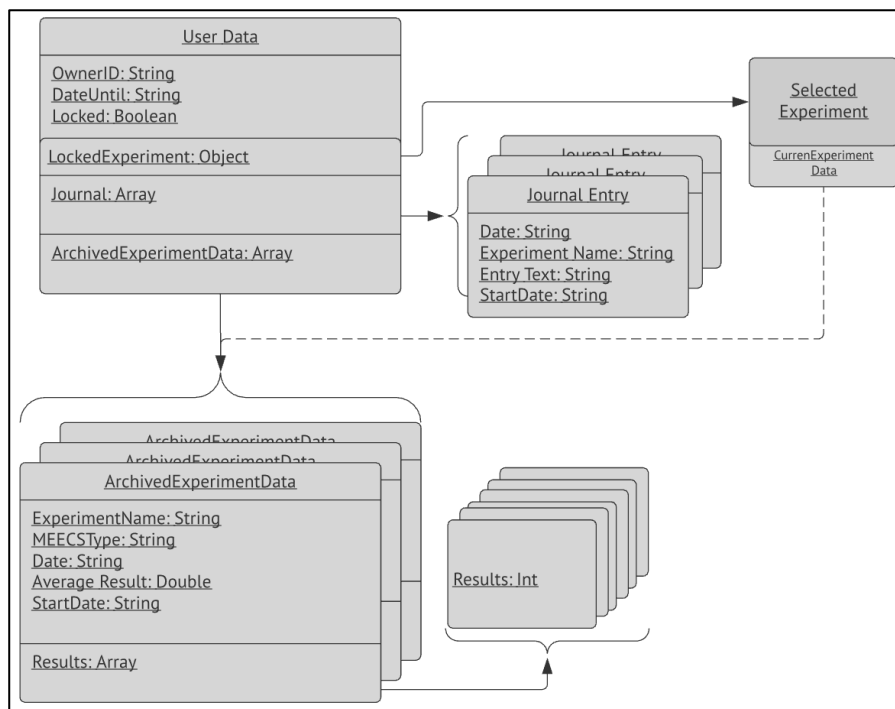


Figure 11: The overview of the “Local User Data” data structure

This section focuses on the “Local User Data” *data structure*. This is the *data structure* that holds all the relevant information for the user side of the application. This *data structure* is one of the most complicated due to its fragmented nature. The reason it became so expansive is that the integration of an online database was added towards the end of the project (discussed in Section 5.2.6). This meant that during the development of the structure, pointing to another data collection was not even a consideration.

Figure 11 shows the entire *data structure* for “Local User Data”. The important distinction between this *data structure* and the “Box” *data structure*, is that this is unique to each user. All the data within this structure is generated, and maintained, in the daily operation of the application. All the data from the user progress and user journal is stored within this structure but as their independent sub-structures. The currently “LockedExperiment” object is also included in this structure (if the user has selected an experiment).

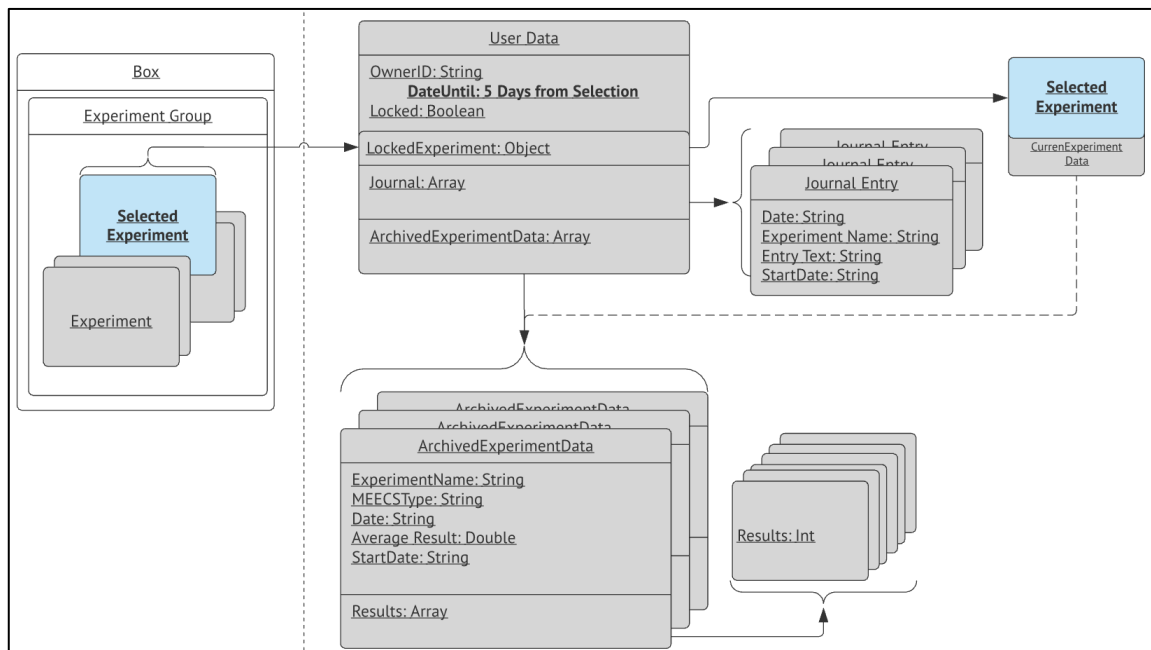


Figure 12: The transaction of data when selecting an experiment (Changes in bold)

After this *data structure* was created in the application, the experiment page was updated to enable submission. When selecting an experiment, the “DateUntil” element is updated with a string of the date five days from the day of submission. Also, the “LockedExperiment” is simply a copy of the experiment object from the “Box” *data structure* previously mentioned in Section 5.1. Figure 12 shows the process submission on an experiment, and what that does to the “Local User Data” *data structure*.

5.2.4 User Progress – Graphs

This section is going to focus on the “CurrentExperimentData” and “ArchivedExperimentData” sections of the *data structure* shown in Figure 12 (as they are both the same *data structures* in a *data array*, but one is segregated for the current experiment).

The “ArchivedExperimentData” *data structure* holds all the relevant information for a user’s responses to the daily surveys including an array that holds the individual responses for each question in the survey. This was done to enable future developers to be able to do

more data processing, such as a weighted average, or categorizing the different questions to a different kind of attribute for the experiment. Now, however, the average is a simple mean of the values within the response and is stored at the top level of the *data structure* (to prevent re-calculation each time the average is requested to be used).

The response page is generated via the “LockedExperiment” *data structure* and shows all the questions relevant to the experiment. Once the user has answered all the questions and submits, the *data structure* is saved to the “CurrentExperimentData” *data structure*. This ensures a distinction between old experiment data and the information of the current experiment. The process of the survey response being completed and the relevant data changes in the structure are shown in Figure 13. If the survey can be completed more than once in a day, then the data is overwritten, with the latest information being the data that is kept.

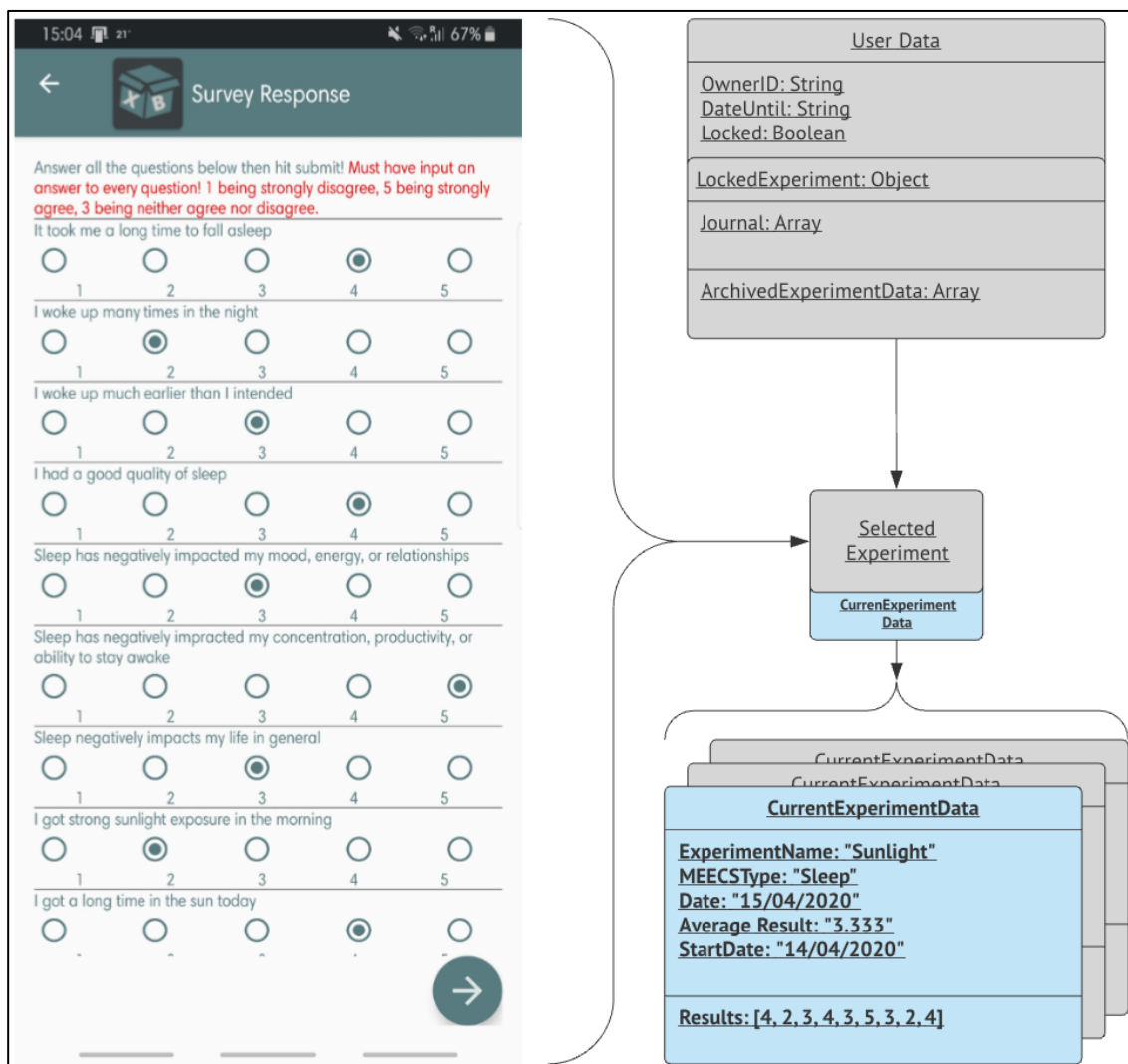


Figure 13: Survey Response Page inputting new data into the “CurrentExperimentData” *data structure* (changes in bold)

Once the experiment has finished (and the application is opened), the current data is transferred to the “ArchivedExperimentData”. This is then only used for display purposes and the structure is never edited. This helps to prevent the current data from overwriting the old data accidentally and means that processing the current data is quick, in

comparison to indexing through an increasingly long *data array*. During the transfer, the whole *data structure* is changed to release the user from the experiment and take any of the response data to be archived. A graphical representation of this is shown in Figure 14.

The use of this data in the application is in the graph section of the Progress page. This section is the default screen upon opening the Progress page. The responses the user has generated are displayed on this page, in both a calendar and graphical format. The calendar uses spots under the dates to indicate if there was a response to the survey on those days. It is colour-coded to show the distinction between different experiment blocks and is the interface through which the user interacts with the graphs.

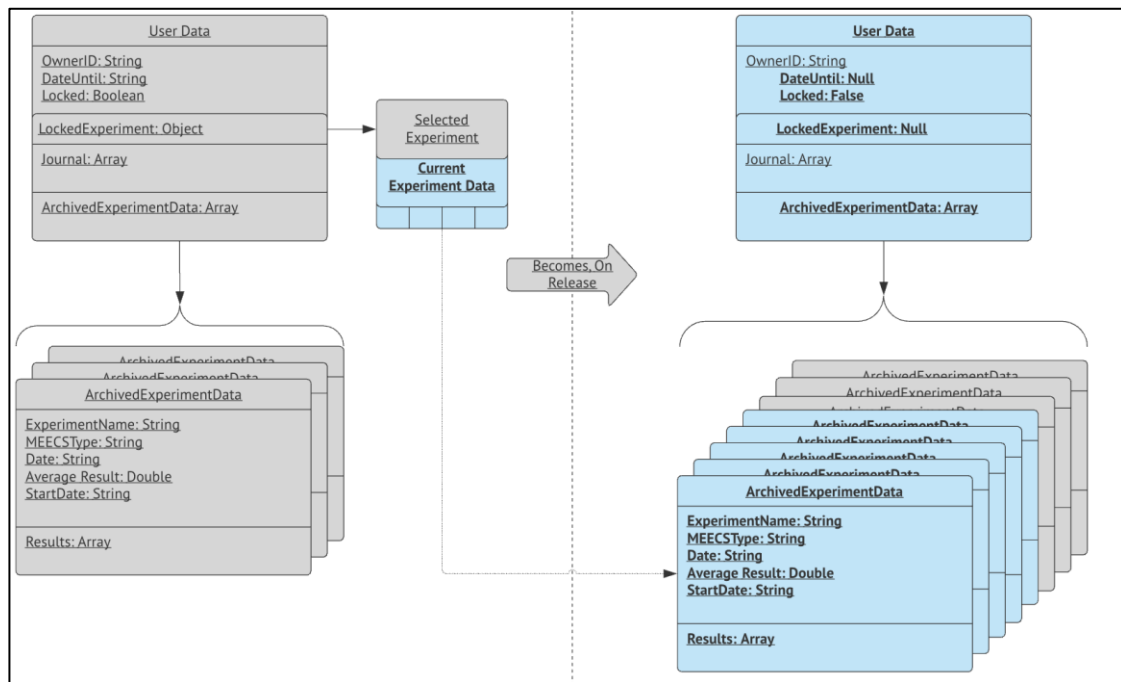


Figure 14: *Data structure* change to User Data on release from an experiment (changes in bold)

The attraction of this page is the graphs, which use the “Graph View” library by jjoe64 from GitHub [11]. The library supports dates as the x-axis, and also being updated dynamically on screen, but is extremely limited in the ability to change the views without direct user input. A feature that had to be discontinued as it was simply too complicated to add, was to scroll the overall graph to the date selected in the calendar. The library, however, does not support any simple way to scroll the view programmatically, and so this feature was left out. No other free-to-use library for graphs seems to be available online for Android.

There is a potential of three graphs displayed on the progress page, these are the current experiment progress, the selected graph (from the calendar), and the overall progress graph. The current experiment graph and the selected graph have a range of 5 days, due to the nature of the lengths of XBs. The overall graph, however, shows a 15-day window and allows the user to scroll forwards and backwards to view their overall progress within the application. The results displayed are all the same but are in different formats to enable the user to attain a different perspective on their current progress. If there is no available data for the current experiment (i.e. there is no information on the “CurrentExperimentData”) then only the selected experiment (which will be set to the

most recent experiment on launching the page) and the overall graph will be displayed. If there is no data whatsoever within the application (the user is brand new and has opened the progress page) then no graphs will display, and instead a text message stating that there is no data will be shown. The three possible scenarios for displaying graphs are shown in Figure 15, with different calendar properties in each as well.

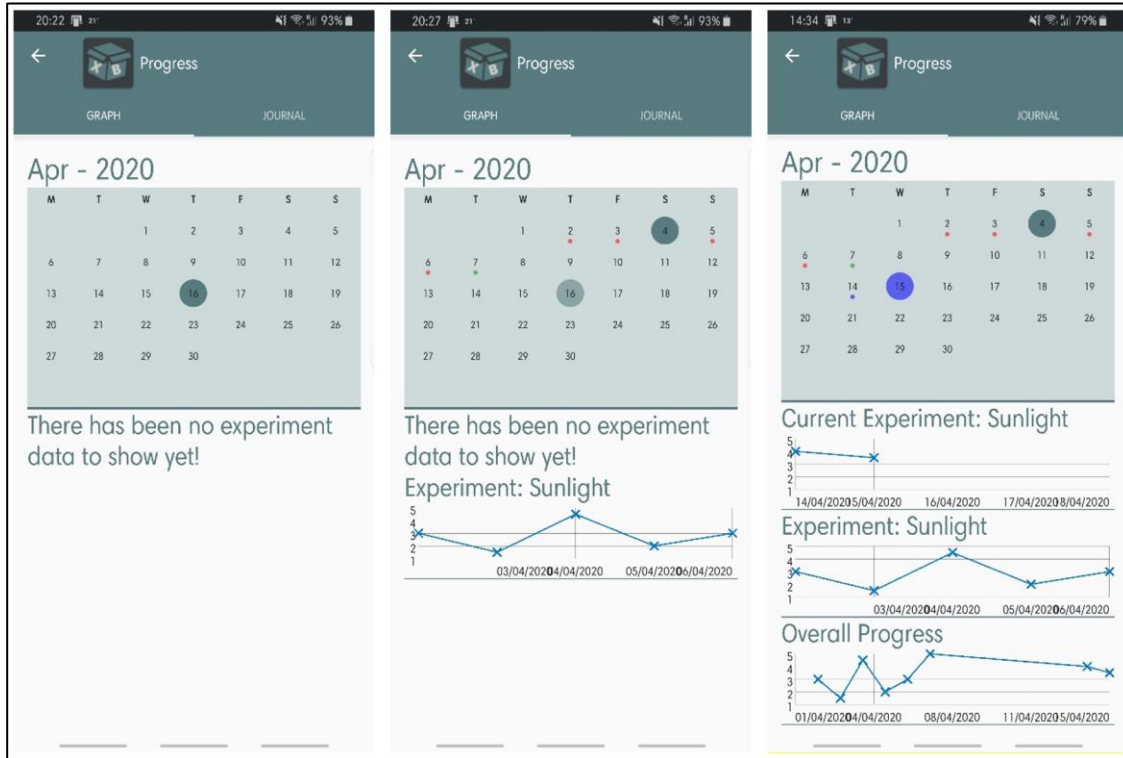


Figure 15: The three scenarios for data display in the Progress page (Left) No data available (Middle) No current experiment data available (Right) Current and archived experiments data available

5.2.5 User Progress - Journal

The user journal is displayed under the Progress page, but the user must swap the tab to the Journal via the AppBar at the top of the screen. This page uses the *data structure* known as “Journal” under the User Data *data structure*. Much of the index information is the same as the response data for experiments; these being the current date, the start date, and the type of the experiment. The difference, however, is in the journal, the data is a string that holds the input from the user. The entire *data structure* can be seen on the right of Figure 12.

With this *data structure* in place, and as the process had already been developed for the graphing functionality, much of this implementation is the same. The calendar shows the dates that have a journal entry, colour-coordinated to distinguish between different experiments. This page (and the graph progress page) use the library “Compact Calendar View” by SundeepK on GitHub [12]. This library proved invaluable, as generating the GUI structure for a calendar would have taken far too much time. The library was, however, limited in some respects. Many simple features, such as not selecting a new date on scrolling, or even only allowing selection of certain dates, had to be implemented on top of the library. The reason SundeepK’s library was chosen over any of the other free and available calendar GUI libraries for android (which had these simple implementations included) was that it supported the ability to add colour-coded events to the days, which

was crucial to be able to visually display the breaks in experiments (most other libraries did not support displaying events in the same way).

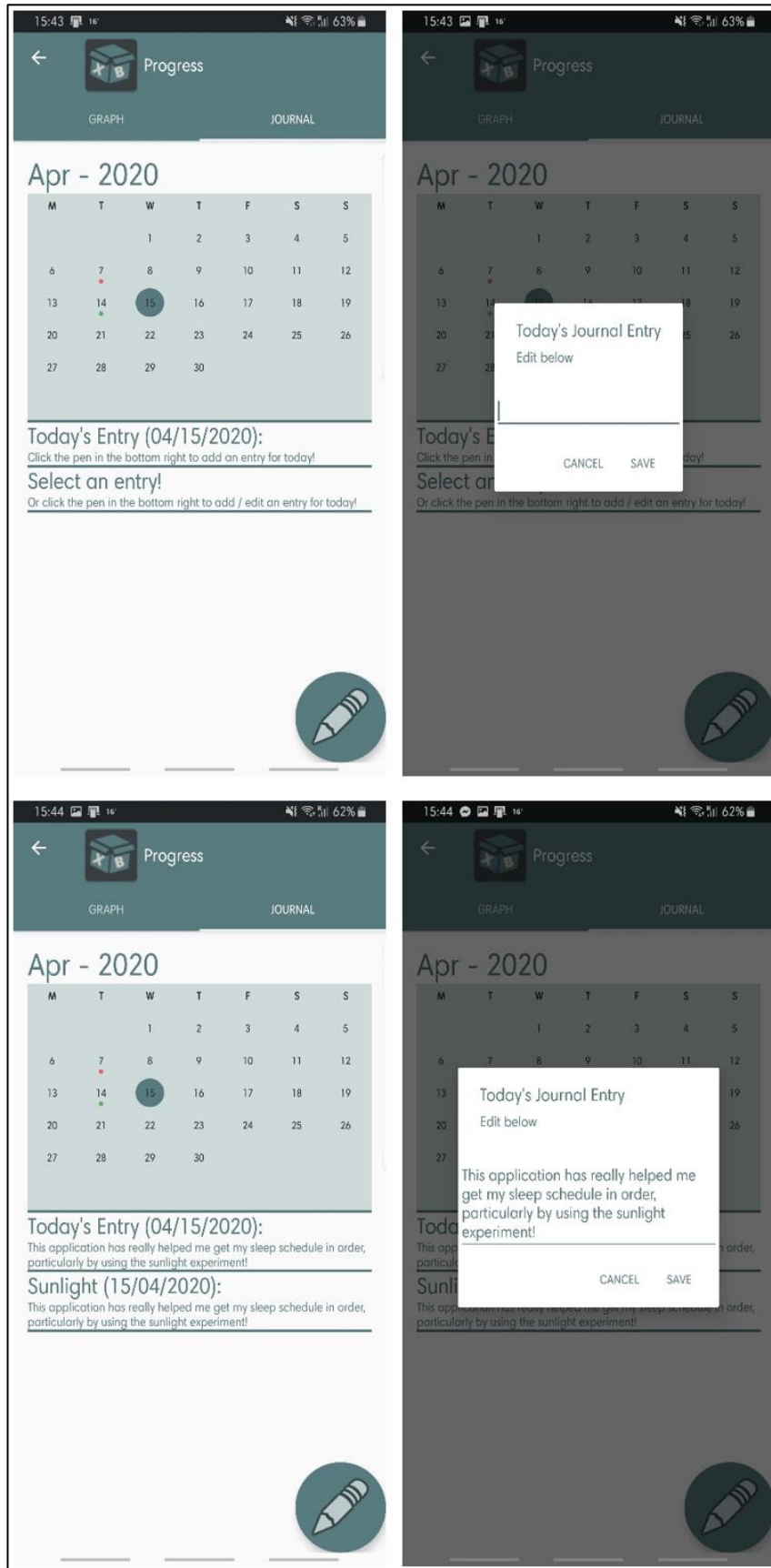


Figure 16: The different scenarios that the edit journal entry pop-up can be displayed from

Additional functionality on this page is the ability to edit today's entry. If there is no entry in today, then the edit button will simply create a new *data structure* to host the information. The user can enter their text (as short or detailed as they would like) and then upon saving, the data is immediately rendered on screen. If the user is to then click the edit button, the data will be loaded into the edit text element to be altered as required. The reason this edit text appears in a pop-up as opposed to remaining on the screen with the rest of the elements is to indicate to the user that this is a different kind of element to the others on screen. Everything on the Progress page is for displaying information to the user, whilst the pop-up indicates that this is the only area that can be edited. Figure 16 shows the screen with no data, and how the pop-up appears in that case and compares it to the screen when there is data, and how the pop-up appears when there is already text in the entry.



Figure 17: A demonstration of the journal selection feature (note that the 04/15/2020 entry is repeated)

The journal entries are only shown for the current (or selected) experiment. This means that upon selecting an experiment, the GUI is updated with the corresponding experiment

information, and displays the relevant entries. This is done by searching for the *data array* for experiments with the same name, and the same start-date, to ensure that only a maximum of 5 entries is displayed at one time. The selected entry is also repeated, in a larger font, just below the current day's entry. This is to highlight the users desired entry. The reason this was done, rather than to enlarge an entry in the middle of the array on the screen, is because if the user had selected the final day, and there were particularly long entries in the rest of the experiment, the user would have to scroll for a while to see their selected entry. The operation of the entry selections is shown in Figure 17.

5.2.6 Data Repository Using MongoDB

Towards the end of development, the implementation of a database to sync user data was implemented. This database makes use of the free MongoDB online database. The interfacing with MongoDB was done using the Mongo Stitch Android library [13]. The reason MongoDB was chosen, was because the information within the application was stored in a JSON format, and MongoDB has native support for transforming the objects downloaded straight to a JSON object. This helped avoid writing code for conversion between a different format and JSON format within the application.

Also, the login page benefitted from the MongoDB integration. Previously it was being handled by Auth0 (another authentication library). This was removed in favour of MongoDB's implementation, despite the lowered security, the integration was deemed a greater benefit. Security was not that large of concern because no personal information was being stored on the database.

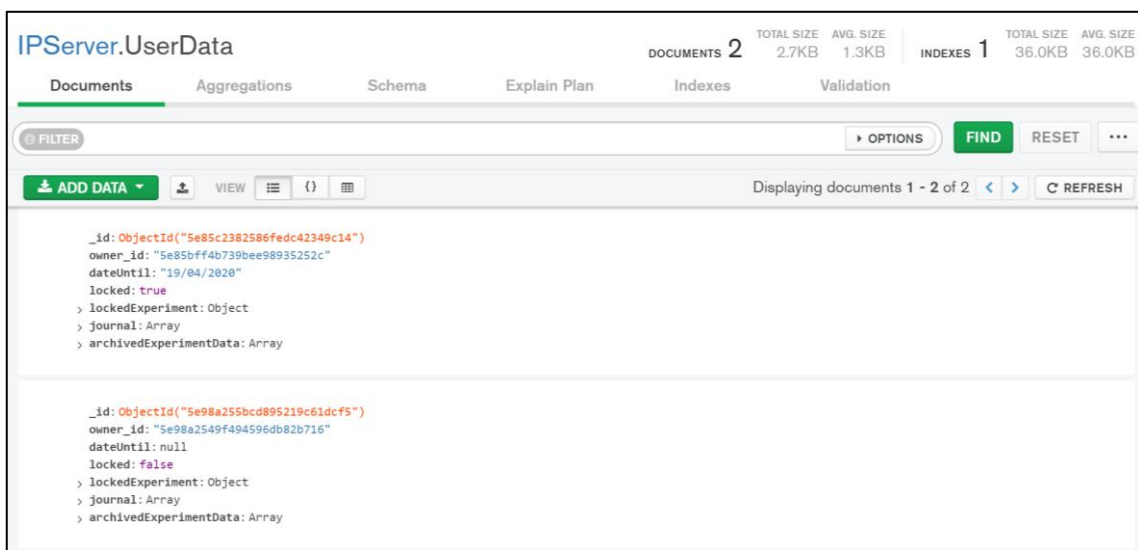


Figure 18: The collection of user's data in the MongoDB Atlas computer client

MongoDB also provides a desktop application, known as MongoDB Atlas, to navigate through the data-structures in a manageable, easy to understand format. Two collections (the naming of the containers for the *data structures* in MongoDB) were created to keep in line with the file formats used in the application. This means that there is a “Box” collection and a “User Data” collection. The operation of the two collections is slightly different.

The “Box” collection holds the information for the already-approved and pending-approval box submissions. This means the data has been produced by the New Box

Creation and Experiment Creation pages, whereupon completion it is uploaded to the collection with the flag of “approved” set to false. The proposed process is to then have a moderator look through this upload, and either approve the *data structure* (by changing the flag) or rejecting it (by deleting the object). The application syncs with the database before the Box page is opened, this allows only the most up to date, approved *data structures* to be displayed on the Box page. Only the approved documents are downloaded. The actions that cause the database to be synced are included in appendix I.

The “User Data” collection holds the information for each user. To access the *data structure*, the user needs to be logged into the MongoDB service (via the application) and then their associated user id is used to access the relevant *data structures*. The functionality of this is simply to keep an online version of the local user’s data. There is no personal information associated with any of the *data structures* and as such, the data is anonymized and can be extracted for the relevant boxes (although this must be done manually). The collection of all the currently available “user boxes” (the two testing users) is shown in Figure 18.

The database is kept up to date and in sync with the local application in two main ways. If the user generates any information that pertains to the user *data structure*, it is uploaded at the time of creation. Each time, before the box structure is used, it is resynced with the client. This ensures that there is a direct one to one relation between the local data and the online data. There is currently no check to see if the sync was successful or not, and so sync issues may occur if the user's wi-fi is not enabled.

5.2.7 Overall GUI Theme

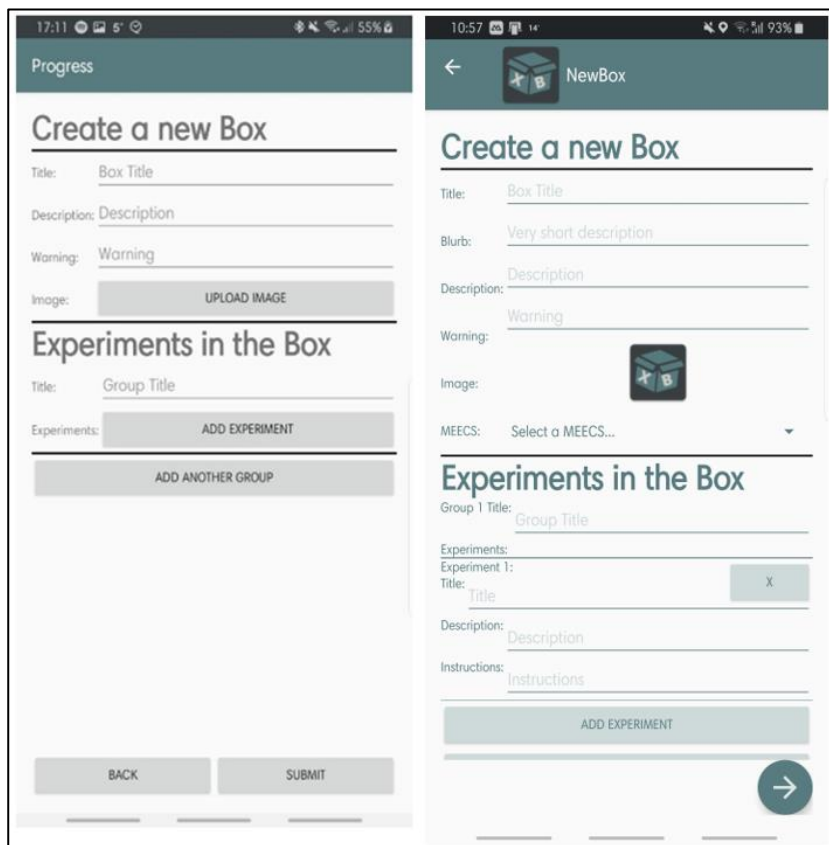


Figure 19: An example of the changes between the old GUI style and the new GUI style (using New Box Creation page as the example)

The final change to the application was to do with aligning the look of the GUI. Colours were changed to be more in line with an overall turquoise colour aesthetic. Android already has a “Style” file included with standard projects which control how GUI elements look by default (meaning tracking down each instance where an element was generated was not required). This was quite a simple process but gave a finished, polished feeling to the entire application. However, the GUI could still use some tweaking, especially depending on the device it ran on. For instance, the application was simulated on both a Google Pixel 2 emulator (built into the Android Studio IDE) and onto a physical Samsung Galaxy S8. The difference in appearance on these two screen sizes is quite notable and would need some tweaking before the application would be launched. Figure 19 shows the comparison between the old colour scheme and the size difference in the GUI elements.

6 Testing

The following section demonstrates the user interaction with the *XB Framework* application, as well as the creation of a box within the application. An analysis of this testing is discussed in Section 7.

6.1 Reproduction of the SleepBetter XB

The first test of the application was to ensure that a current XB could be recreated within the application. The methodology of producing the XB within the application was to:

1. Curate the information for how the XB is already run (for this example, George Muresan (the creator of SleepBetter XB) was emailed, and he was able to provide the required data for this test)
2. Start the Creation of a New XB within the application
3. Complete the relevant information
4. Submit the information to be saved to the server
5. Approve the box on the MongoDB Atlas client
6. Compare the results with the old XB application through user testing

The comparison between the Sleep Better XB application and the *XB Framework* recreation is shown in Figure 20. The data input into the Create New Box page and the Create New Experiment pages are shown in appendix J, with the associated document in the MongoDB Atlas included.

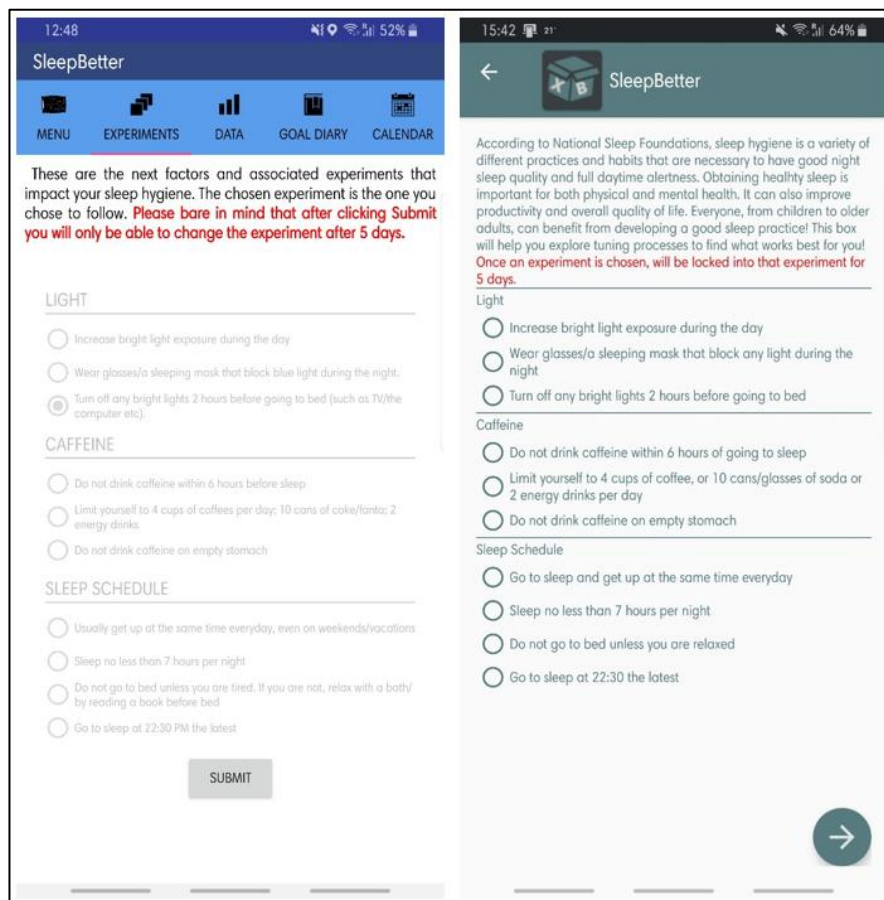


Figure 20: Sleep Better XB recreation process into the *XB Framework*

6.2 User Experience of the SleepBetter XB

The SleepBetter XB application has features that can be directly compared to the *XB Framework's* features. These are the Journal, Survey Response, Calendar, and Graph pages for each application. Each direct comparison can be seen in this section.

6.2.1 Journal

Figure 21 shows the pages side-by-side and their direct comparison.

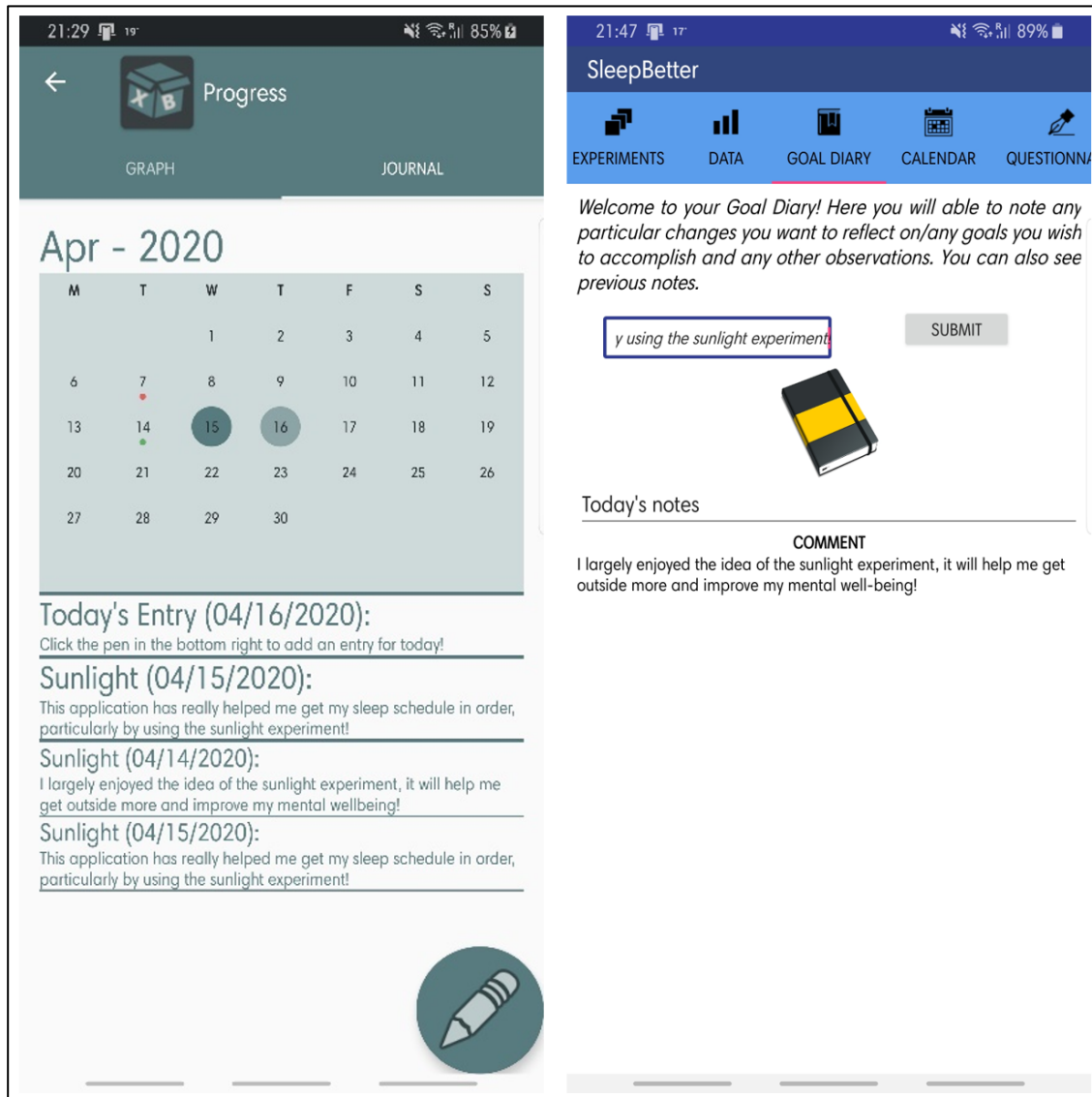


Figure 21: Direct comparison between (Left) *XB Framework* and (Right) *SleepBetter XB* application

6.2.2 Survey Response

Survey Response

Answer all the questions below then hit submit! **Must have input an answer to every question! 1 being strongly disagree, 5 being strongly agree, 3 being neither agree nor disagree.**

It took me a long time to fall asleep

1 2 3 4 5

I woke up many times in the night

1 2 3 4 5

I woke up much earlier than I intended

1 2 3 4 5

I had a good quality of sleep

1 2 3 4 5

Sleep has negatively impacted my mood, energy, or relationships

1 2 3 4 5

Sleep has negatively impacted my concentration, productivity, or ability to stay awake

1 2 3 4 5

Sleep negatively impacts my life in general

1 2 3 4 5

I got strong sunlight exposure in the morning

1 2 3 4 5

I got a long time in the sun today

1 2 3 4 5

PRE-QUESTIONNAIRE

Your current experiment is: INCREASE BRIGHT LIGHT EXPOSURE DURING THE DAY. Below you are asked to complete the questions on whether you followed the experiment.

DAY REVIEW

Do you think you feel better than yesterday?

A lot worse
 Worse
 The same
 Better
 A lot better

LOGGING

Did you get sunlight exposure in the morning?

No
 Yes

Did you stay at least half an hour in the sun today?

No
 Yes

Did your room capture the sunlight today?

No
 Yes




Figure 22: The comparison between (Left) *XB Framework* questionnaire and (Right) *SleepBetter XB* questionnaire

The direct graphical comparison is shown in Figure 22. It should be noted that the *SleepBetter XB* application kept crashing before the full questionnaire page layout could be seen, but it can be assumed to be laid out similarly to the pre-questionnaire.

6.2.3 Calendar

Figure 23 shows the direct GUI comparison.

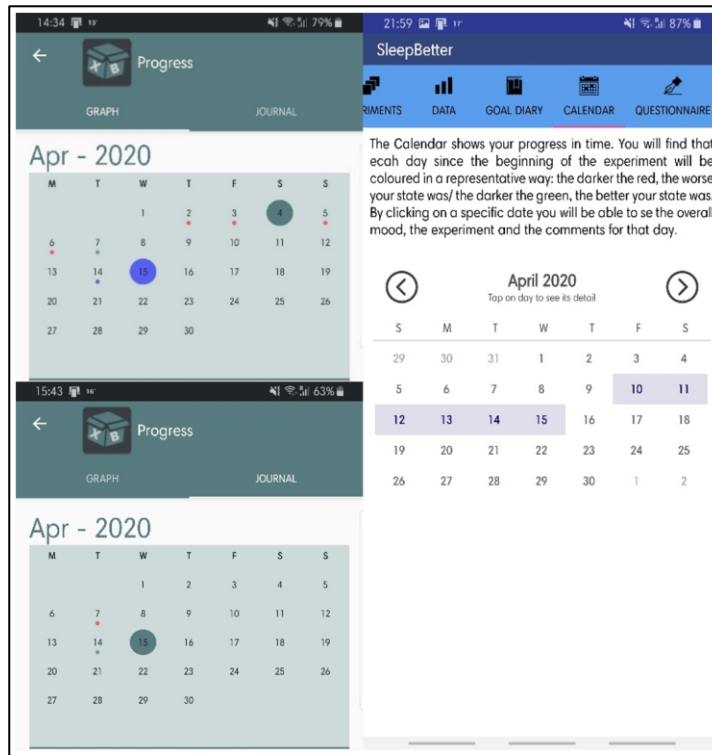


Figure 23: Comparison between (Left) The two calendars in the *XB Framework* and (Right) the SleepBetter *XB* calendar

6.2.4 Graphs

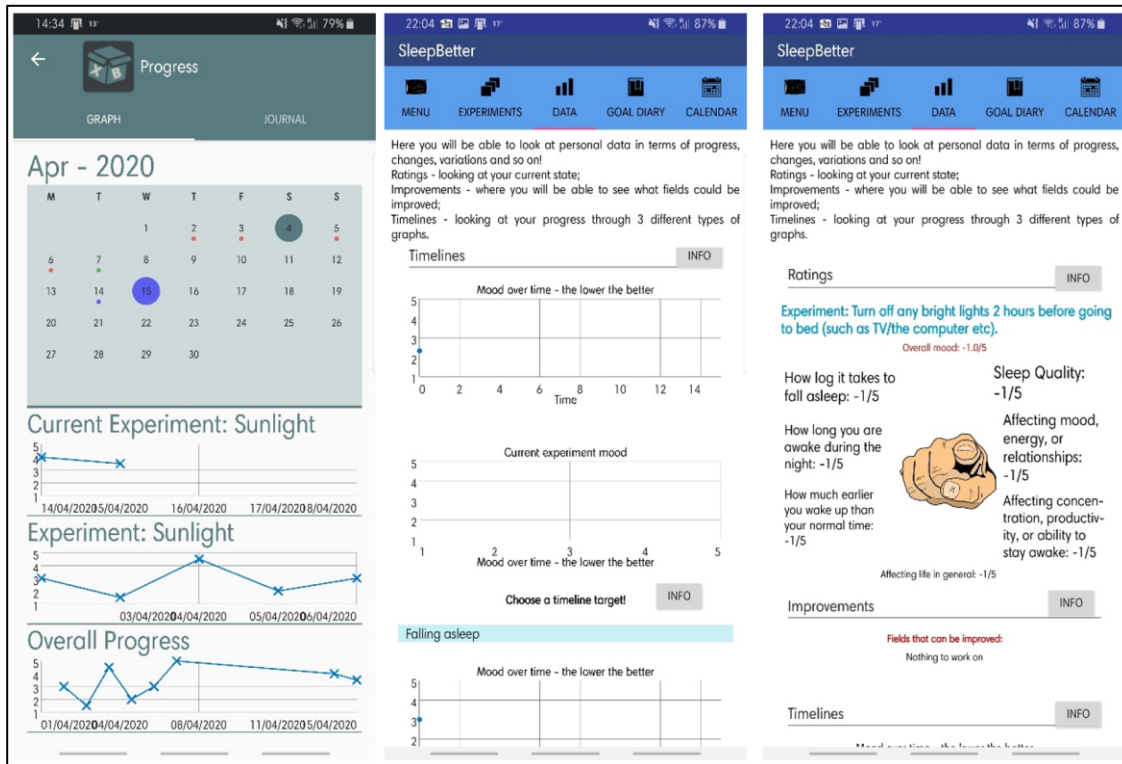


Figure 24: Comparison between (Left) *XB Framework* Graph page, (Middle) SleepBetter *XB* Graph page (Right) SleepBetter *XB* Data analysis page

The direct graphical comparison is shown in Figure 24. Unfortunately, none of the graphical data could be input on the SleepBetter XB as the survey response kept crashing on testing.

6.2.5 Notifications

The notifications from both SleepBetter XB and *XB Framework* are displayed in Figure 25.

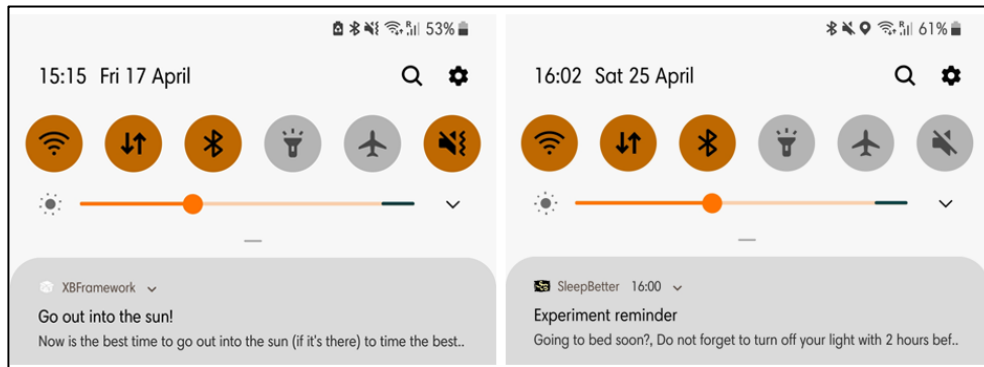


Figure 25: Notification for *XB Framework* (Left) and SleepBetter XB (Right) [Note: the colours are inverted to improve readability]

6.2.6 Login and Data Syncing

This test does not apply to the SleepBetter XB. The process of storing and loading data per user account was tested across the emulated devices (a desktop personal computer and a laptop) as well as on a Samsung Galaxy S8. The process of testing involved answering the survey daily, on a different device on each occasion. The different applications, showing the same synced data can be seen in Figure 26.

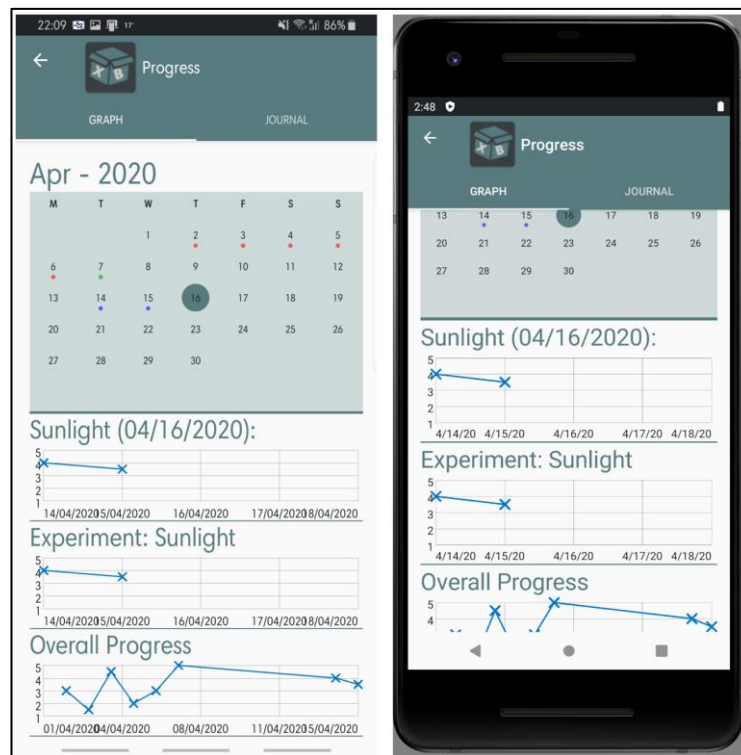


Figure 26: (Left) Samsung Galaxy S8 *XB Framework* application (Right) the same user data on the Google Pixel 2 emulated application

7 Analysis

This section focuses on the analysis of the result seen in Section 6, with the SleepBetter XB and specifications mentioned in Section 3.1.2 as references.

7.1 Reproduction of the SleepBetter XB

This test was to ensure the ability of the *Framework* to produce an XB. The importance is that as this is an already running experiment, if the application can replicate it, then it is on par with currently available independent applications for XBs. Also, the ability to produce an XB is a key part of the specification, with points six to nine on the specification list (found in Section 3.1.2) dedicated to the production of an XB.

The process of creating an XB is explained in Section 6.1. To be successful, it had to be able to produce “A choice of one, or more, experiments within the box”. The Sleep Better XB has three experiment groups (Light, Caffeine, Sleep Schedule), with three experiments in the first two groups and four in the last group. This was replicated successfully in the *XB Framework*, as seen in Figure 20.

The next criteria were to be able to have “Instructions / Guidance for the experiments within the box”. Despite both an instruction and warning text being displayed for the Box level information, the experiment level Instructions tag is not displayed anywhere within the application. This has therefore not met the specification.

“A custom survey to be completed daily for each experiment individually” was the next point in the specification. The experiments each have their survey questions, meaning this point was a successful implementation, shown by the test in Section 7.2.2.

The final point was to have “Customizable notifications for each experiment”. Section 6.2.5 shows the notifications working correctly from the *XB Framework*, as a result, it is deemed another successful specification point, shown by the test in Section 7.2.5.

Overall, the production side of the specification successfully produces all the information required. However, since the experiment level instructions are not displayed, this point is the only one deemed not successful.

7.2 User Experience of the SleepBetter XB

This section focuses on the comparison between the user experiences on the SleepBetter XB application and the *XB Framework* application.

7.2.1 Journal

With the specification of “Displaying a box/experiment with a journal/diary entry for users’ comments”, the *XB Framework* successfully provides a feature to handle this requirement. The *XB Framework* journal feature enables an editing functionality above and beyond the comparable features found in the SleepBetter XB.

7.2.2 Survey Response

The specification this test addressed was to “display a survey response page with the questions associated with that experiment”. With each question displayed, and with the

results being saved to the user's data, this point is deemed successful. It should be noted, however, that whilst the *XB Framework* does successfully implement this feature, SleepBetter XB has more complete functionality. In SleepBetter XB these questions are categorized and so a more in-depth analysis of the results can be made. Also, some questions enable yes or no answers, and a time of day response.

7.2.3 Calendar

This section addresses the specification point of “displaying a calendar that displays interaction/progress”. The calendars shown in Section 6.2.3 show clear markings of displaying when the user has interacted with the application, be that through responding to the daily questionnaire, or through a journal entry. The SleepBetter XB application shows the calendar on a separate page from the others and allows the user to view their previously entered journals. By comparison, the same features are available in the *XB Framework* and will show the user the relevant data graph or journal data on the page.

7.2.4 Graphs

As the specification point for graphs states “Displaying a progress page with graphs”, the *XB Framework* completes this requirement. However, as SleepBetter XB can categorize results, this allows SleepBetter XB to display these categories. This ability of analysis is lacking from the *XB Framework*; however, it was not a requirement. The addition of a calendar in the *XB Framework* is also an added chance for reflection on older experiments.

7.2.5 Notifications

“Displaying notifications for the experiment” was deemed a requirement in the specification for this project. As the results in Section 6.2.5 shows, the *XB Framework* successfully implements this functionality. Also, these notifications can be set by the experimenter upon creating the XB.

7.2.6 Login and Data Sync

Above and beyond the specification for this project, the *XB Framework* also includes a login and syncing feature. This was shown to be successful and has helped to simplify the process of authentication drastically as a previous implementation had a third-party application (known as Auth0) handle authentication, with MongoDB only being used as a storage application. The data-syncing capabilities were very effective, with updated information being available for download seconds after the syncing had been produced. One fault is if two devices had opened the progress page con-currently, and both edited the information. In this case, there would be a de-sync of information, and the most recent update would be kept. This, however, is not a foreseeable issue that users will encounter during normal operation.

8 Evaluations and Future Work

This section focuses on the evaluation of the project, both the effectiveness of the planning and the analysis of the overall project's success.

8.1 Evaluation of Planning

This section will focus on the quality of the planning used during this project, and a short reflection of the knowledge gained from the process, including a reflection of what would be changed if the project were repeated.

8.1.1 Time Management

The time management for this project was adequate to get the project to a level of completion I am happy with, however, could have been improved. The biggest time sink that occurred was during the first half of the project when my focus was split between the hardware and software research. If I were to repeat the project, I would start with a smaller idea and expand the functionality if time was available. The issue was that despite having a module dedicated to this project in the first semester, and two in the second semester, the prediction of my time management between progress and keeping up with my other modules was too optimistic. As a result, the project didn't begin to make good progress until the second semester.

If the project were to be repeated, I would not use a Gantt chart again. Despite it being well known for its use in project management, it seemed to be estimating the timeline too much, especially without iterations being made throughout the project. Instead, a tool like Trello could be used. The reason for this is it produces a list of tasks that cascade into one another (much in the same way as a Gantt chart) but is easier to follow and estimates of how long tasks would take are not required. It is also revisited and revised as the project progresses.

A logbook would be used again in a future project, as the ability to keep both the thought process for the project and general planning together was extremely useful. This would be used in combination with the online Trello, with the logbook being a micro, detail-oriented version of the macro, dynamic Trello version. The benefit of an online tool of Trello (along with the logbook) is the ability to allow team members, or supervisors, an insight into the progress being completed and the tasks remaining.

If the project were to be repeated, the order in which the implementation was performed would be changed. I would start with the *data structure* in the application. The reason for this is that MongoDB allows for objects to be linked to one another through their "DocumentID" field. In particular, the "LockedExperiment" object, rather than being a copy of the data in the Box information, could be a pointer to the original document. This would prevent data from becoming obsolete. It would also reduce the file size of downloads and uploads greatly, which of course when the app is scaled would be required for both server-side and client-side performance.

8.1.2 Risk Management

The use of the risk matrix was useful, even though many of the situations that were considered did not appear. This project management tool was the most useful in terms of

time taken to produce, compared to the effectiveness. I will be using a risk matrix in my future projects. The patent research was also beneficial to ensure that my scope was not already explored, nor that there would be any issues with publishing or developing such an application. I do not underestimate the importance of patent research before a project begins and will be repeating this practice.

8.1.3 Tool Usage

If the project were to be repeated, Android Studio IDE would be used again, however, rather than using the Java language, I would change to using C++. This was not an option at the beginning of the project, however, an update to the IDE was introduced to allow C++ coding. Since I have far more experience with C++ than Java, this would be the language I would use if repeating the project.

Some of the GUI libraries used were quite limited, but with every alternative library, there seemed to always be some added features, but also some removed features. This meant that the decisions of what libraries to use were based on a select feature that was deemed the hardest to implement on top of the library. In the case of the graph library, this was the only one available for Android development.

I spent a fair amount of time trying to find a library that had all the features required. Often swapping a library out for another, trying to test the features (as many of the open-source documentation are limited, and simply trying different methods is more telling of the limitations). Knowing this now, if repeated, I would not spend as much time researching alternative libraries, and instead, pick the one with the implemented functionality that is closest to what I desire, and then handle the limitations myself. If the project were to be repeated, I would have no issue with using these same libraries again as they worked as intended and have allowed functionalities that I would not have been able to implement in the time frame.

8.2 Project Evaluation and Future Work

Despite being completed to a good level, the project still has many features that would be beneficial to be developed. The first of which would be a port to iOS. With the combination of both Android and iOS ecosystem development, the application would have the potential to be on every phone in the market. Unfortunately, there are no tools that can be used to directly port the application from the Android code to iOS. Instead, the development process would either have to be established on a tool such as React Native by Facebook. This essentially uses HTML, CSS, and JavaScript to build a cross-platform project. This is a very easy way to build a modular GUI, with intractability. This method is highly supported by many libraries.

Another avenue of further development would, of course, be the artefact development and interaction. This, unfortunately, fell out of scope for the project, however, any form of device that can go alongside an XB experience to collect and enhance understanding of the environment without the user having to actively think about this data collection would be a great step forward in the HCI domain, allowing the user to think more about *how they feel*, rather than *how many* steps they have a walk or *how much* water they drank. The information produced by this project, but not included in the main body, is available in appendix A.

The application still needs more development in the user interface, before the application can be launched to the public (and after passing ethical considerations). The application should have many more tooltips explaining the functionality of certain aspects (such as on submission that the Box is under review), not to mention terms and conditions (and GDPR agreement, etc.) at the time of registration. Once these are implemented, a formal test of the User Interface with new users would be a good evaluation of the progress.

9 Conclusion

Concerning the specifications outline in Section 3.1.2, the project has successfully implemented an *XB Framework* for researchers to launch XBs from.

After understanding the current process for developing XBs, and their supporting research, the *XB Framework* was designed around being able to create the most simplistic XB with a view of expansion in the future.

Based on the testing performed in Section 6 and analysis from Section 7, it is clear to see that the core functionalities of an example XB are available in the *XB Framework* application, with eight of the nine specification points defined in Section 3.1.2 being successfully implemented. The project has been deemed a success due to completing a large majority of the specifications.

At the current stage in development, there are still some small functionalities missing, but the general process has been defined and implemented, however, there are many avenues for future work available.

The development process was started on Android due to its larger market share than iOS. As a result, a good project would be to develop an iOS equivalent or even develop a cross-platform version of the application.

Another great avenue for future work is in artefact development. This explores not only the XB model but also a fundamental HCI question of how to produce data without manual input.

To summarize, the project was successful, with a simple XB being able to be created through the *Framework*, and there is a large case for future development, with many potential avenues for development, from hardware integration to database optimization that would greatly improve the usability of the application.

The word count of this report was calculated by Microsoft Words included word count by selecting the whole main body from the “Introduction” header to the final word (including textboxes, footnotes and endnotes). It reported a word count of 9840.

10 Bibliography

- [1] S. Fox and M. Duggan, "PIP Tracking for Health," *Pew Internet & American Life Project*, California, 2013.
- [2] C. M. Swinkels, C. S. Ulmer, J. C. Beckham, N. Buse, and P. S. Calhoun, "The Association of Sleep Duration, Mental Health, and Health Risk Behaviours among U.S. Afghanistan/Iraq Era Veterans," *Sleep*, vol. 36, no. 7, pp. 1019–1025, Jan. 2013.
- [3] M. Kredlow, M. Capozzoli, B. Hearon, A. Calkins, and M. Otto, "The effects of physical activity on sleep: a meta-analytic review", 2020.
- [4] M. C. Pachucki, E. J. Ozer, A. Barrat, and C. Cattuto, "Mental health and social networks in early adolescence: A dynamic study of objectively-measured social interaction behaviours," *Social Science & Medicine*, vol. 125, pp. 40–50, 2015.
- [5] G. Wendel-Vos, A. Schuit, M. Tijhuis, and D. Kromhout, "Leisure-time physical activity and health-related quality of life: Cross-sectional and longitudinal associations," *Quality of Life Research*, vol. 13, no. 3, pp. 667–677, 2004.
- [6] M. Schraefel, "Mobile Being: How Inbodied & Embodied Practice May Inform Mobile Cognition," *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct - MobileHCI 15*, 2015.
- [7] M. Schraefel, "in5: a Model for Inbodied Interaction," *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems - CHI EA 19*, 2019.
- [8] J. Proud, "Mobile device that monitors an individual's activities, behaviours, habits or health parameters," *United States Patent 15/098,070*, June 2013.
- [9] S. O'Dea, "Mobile OS market share 2018 | Statista", *Statista*, 2020. [Online]. Available: <https://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems/>. [Accessed: 17- Apr- 2020].
- [10] "Android-Image-Cropper", *GitHub*, 2020. [Online]. Available: <https://github.com/ArthurHub/Android-Image-Cropper>. [Accessed: 17- Apr- 2020].
- [11] "GridView", *GitHub*, 2020. [Online]. Available: <https://github.com/jjoe64/GridView>. [Accessed: 17- Apr- 2020].
- [12] "CompactCalendarView", *GitHub*, 2020. [Online]. Available: <https://github.com/SundeepK/CompactCalendarView>. [Accessed: 17- Apr- 2020].
- [13] "stitch-android-sdk", *GitHub*, 2020. [Online]. Available: <https://github.com/mongodb/stitch-android-sdk>. [Accessed: 17- Apr- 2020].
- [14] M. Streeter, "Incorporating real-world non-coding features into block IDEs," *2015 IEEE Blocks and Beyond Workshop (Blocks and Beyond)*, 2015.
- [15] A. Khalil and S. Abdallah, "Harnessing social dynamics through persuasive technology to promote a healthier lifestyle," *Computers in Human Behaviour*, vol. 29, no. 6, pp. 2674–2681, 2013.
- [16] M. Wu, T.-J. Lu, F.-Y. Ling, J. Sun, and H.-Y. Du, "Research on the architecture of Internet of Things," *2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)*, Aug. 2010.
- [17] S. Aggarwal, D. A. Bau, and D. Bau, "A blocks-based editor for HTML code," *2015 IEEE Blocks and Beyond Workshop (Blocks and Beyond)*, 2015.
- [18] M. Ichinco and C. Kelleher, "Towards block code examples that help young novices notice critical elements," *2017 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 2017.

-
- [19] N. Goyal, C. Hoch, R. Krishnamurthy, B. Meckler, and M. Suckow, "Is GUI programming a database research problem?", *Proceedings of the 1996 ACM SIGMOD international conference on Management of Data - SIGMOD '96*, 1996. Available: 10.1145/233269.233369 [Accessed 17 April 2020].
- [20] L. Wegner, D. Wilke, and J. Thamm, "Database-driven GUI Programming", *Conference: 2. Int. Conf. on Visual Information Systems (Visual '97)*, 1997. Available: https://www.researchgate.net/publication/2330864_Database-driven_GUI_Programming. [Accessed 17 April 2020].
- [21] "Make a 3d Printed Pepper Grinder," *YouTube*, 03-Apr-2019. [Online]. Available: <https://www.youtube.com/watch?v=I5AL4H83fQE>. [Accessed: 03-Dec-2019].

11 Appendix

A Saltshaker Device Design

A.i Artefact Specification

The artefact was a part of the project very early on, and was a completed idea, however, with the fact that the artefact could not work without an application first it was not completed in time. None-the-less, the full explanation of the artefact is below as anyone wishing to continue this research may find both the schematic and the information useful.

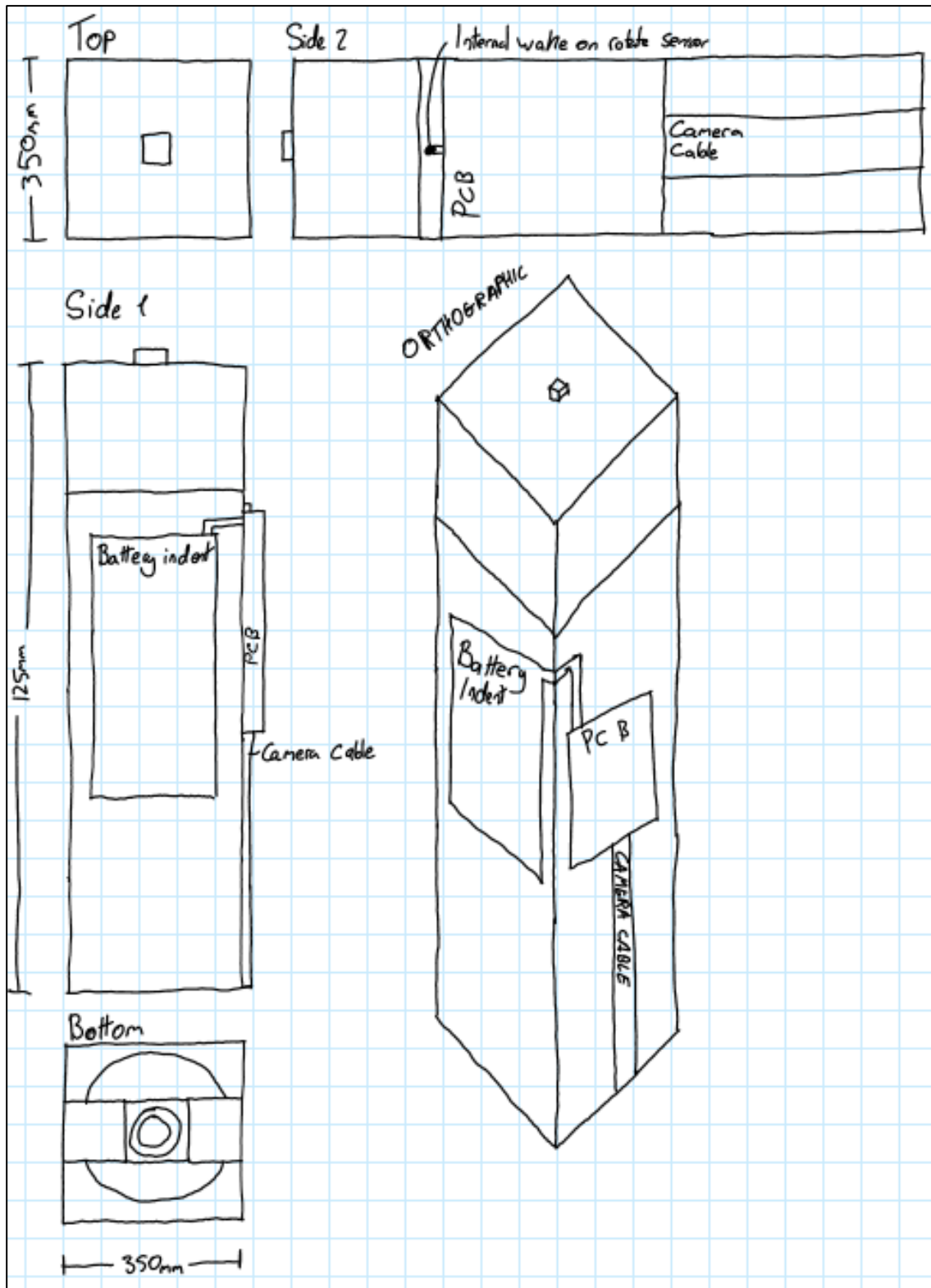
The artefact was designed to be a saltshaker, with a camera attached. The idea being, as the user grinds salt onto their food, the camera takes a picture and syncs to the database. At which point, the application can download this information, and perform some analysis on it. The purpose is to remove the need for a user to think about taking their phone out to take a picture of their food for the data input.

The key features of the saltshaker include:

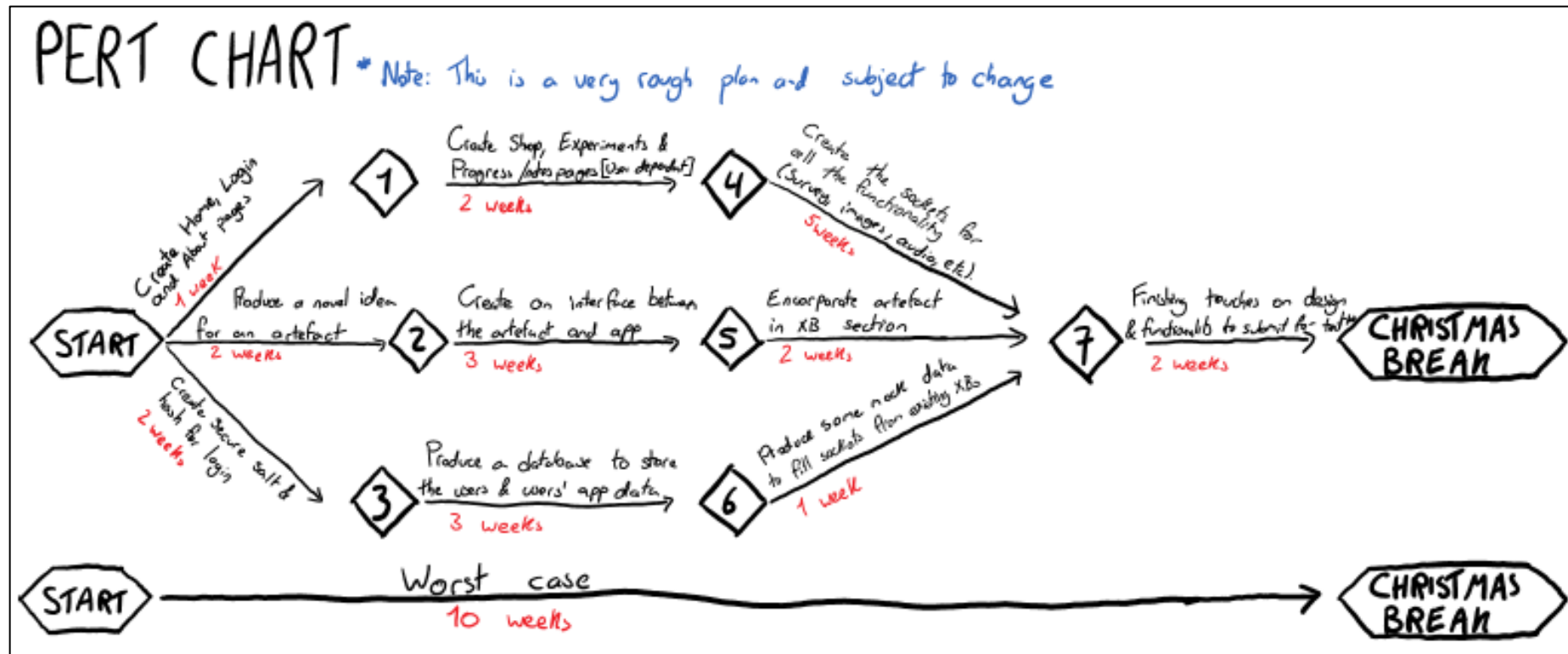
- *Store salt*
- *Grind salt*
- *Have a camera under the bottom the device*
- *Take a picture as the salt is being ground*
- *Sleep when not in use*
- *Use batteries for power*
- *Be able to sense the rotation of the shaker*
- *Use an ESP-EYE with a wi-fi enabled chip to communicate*
- *Be 350 x 350 x 125 mm*

The casing was going to be 3D printed, and the part for the grinder was going to be bought (following the instructions in [21]).

A.ii Saltshaker Schematic



B PERT Chart from the Planning Phase



C Gantt chart

C.i Gantt Chart Legend

Year 3 Individual Project					
University Of Southampton					
Eamonn Trim					
				Start date:	30/09/2019
				View week:	1
NOTE: All tasks, lengths and description are subject to change throughout the project, as they say I might move the goal posts depending on the current progress of the project.					
Task Description	Category	Progress	Start	No. Days	
1. Brainstorming		100%	30/09/2019	14	
1.1 Generate Idea	Low Priority	100%	30/09/2019	13	
1.2 Produce Final Idea and Project Brief	Milestone	100%	13/10/2019	1	
2. Project Planning		100%	01/10/2019	20	
2.1 Finish Gantt Chart Planning and PERT	Low Priority	100%	01/10/2019	20	
3. Development		45%	14/10/2019	189	
3.1 Create Home, Login and About pages	High Priority	100%	14/10/2019	7	
3.2 Create the shop, experiments, notes and progress/data pages	High Priority	100%	21/10/2019	28	
3.3 Create the sockets for all the functions (surveys, image processing, audio processing, etc.)	High Priority	50%	18/11/2019	70	
3.4 Produce novel idea for simple artefact - the salt shaker	Med Priority	100%	14/10/2019	14	
3.5 Figure out the concept of how the device would connect	Med Priority	100%	28/10/2019	28	
3.6 Encorporate artefact to XB selection and creation	Med Priority	0%	25/11/2019	21	
3.7 Create a secure password salt and has that can be used to login	Low Priority	100%	14/10/2019	28	
3.8 Produce a database (SQL?) to store user and all pertinent app data (push / pulling integrated)	Low Priority	0%	11/11/2019	35	
3.9 Produce some mock / launch "socket" data	Low Priority	80%	16/12/2019	7	
3.10 Finishing touches on design and functionality. Submit for launch to student base	High Priority	0%	27/01/2020	14	
3.11 Produce 3D model for artefact and print (the grinder section will be bought and connected to	High Priority	0%	03/02/2020	4	
3.12 Connect the code for the device and have it power up and run on its own	High Priority	0%	03/02/2020	49	
3.13 Bring the artefact together for a final product creation	High Priority	0%	23/03/2020	7	
3.14 Admin / collecting evidence that might have been missed for final report	High Priority	0%	30/03/2020	21	
4. Testing		19%	09/12/2019	105	
4.1 Apply for ethical consideration	High Priority	75%	09/12/2019	14	
4.2 Produce a feedback page for users to post their feedback to	Med Priority	0%	23/12/2019	21	
4.3 Ensure there are prompts throughout the application for the feedback survey, with	Med Priority	0%	10/02/2020	14	
4.4 Receive Feedback and condense reponses into general feedback	Low Priority	0%	24/02/2020	28	
5. Report Deadlines		30%	10/12/2019	158	
5.1 Progress Report	Milestone	100%	10/12/2019	1	
5.2 Write Final Report	High Priority	10%	06/04/2020	22	
5.3 Final Project Report	Milestone	10%	28/04/2020	1	
5.4 Viva	Milestone	0%	14/05/2020	2	

C.iii Gantt Chart for the Second half of the Project (Week 16 – 33)



D Bill of Materials

Item	Price	Quantity	Reason / Explanation	From	Price x Quantity	Link
ESP-32 CAM	£8.96	1	There are ESP-32 Chips individually on Onecall / RS Components, but they do not include the development board that allows for the camera to be plugged in. If not available from the link, they are available on eBay or other sites online.	UK	£8.96	https://bit.ly/3aUeARX
OV2640	£3.53	1	I need the longer version of the cable, to be able to run it down and away from the PCB for the position of the camera	CN	£3.53	https://bit.ly/3bRM98K
Triple-A Battery Holder	£1	1	Batteries for power / I can do this one myself through the system, but I thought it would be easier to get them all done at the same time? Or will these be available in the labs	UK	£1.00	https://bit.ly/3cY4Ckb
Long Single Push Single Pulse Push Switches	£0.22	5	For "User" selection on the device	UK	£1.10	https://bit.ly/35iGUMJ
Total	-	8	-	-	£14.59	-

E Risk Matrix from the Planning Phase

Risk	Likelihood /5	Danger /5	Severity %	Control	Severity after Control %
Ethics approval delay	2	3.5	28	Apply as early as possible and ensure that there are no clear issues with the experiments (which there are not)	7
Libraries not available for the desired applications	1	4	16	Use whatever small libraries are available, and extend them to support the functionality you desire	8
GUI design not being clear enough for the user to interact with	3	1.5	18	Add additional GUI that gives instructions to the user on what the GUI elements are used for	9
Unable to code successfully in java or kotlin and therefore not be able to develop a fully functioning application	2	5	40	Refresh my knowledge of coding from previous projects and be will and ability to use online resources to remind me of concepts I may have forgotten	10
Failed 3D prints for the saltshaker casing	3	2	24	Set aside a long section of time to allow the print to finish and complete other tasks whilst waiting on the print (such as the software for the artefact)	12
Delay on parts coming for the order	3.5	4	56	Order the parts as soon as possible, and focus on software early, so that the latter part of the project can be dedicated to hardware	14
Unable to connect the database and the client for syncing information	2	4	32	Creating the application to work without a syncing functionality	16
The inability for the user to interact with newly created boxes in a deterministic manner	2	5	40	Ensure that the user side of the application is built after the box creation, to have full knowledge of the process of utilizing these data structures	20
Difficulty ensuring the hardware works correctly	2.5	5	50	Being able to fall back on the software side, with the hardware as a proof of concept and a stretch goal	25
Failure to produce an XB that emulates the core functionality of previous XBs	4	5	80	Start with the simple functionality, and then know to complete the rest of the application, even if the time does not permit	40
Difficulty connecting the artefact to the databale to sync with a user	5	5	100	Have the artefact sync to an "admin" user for debugging to show the concept working, then consider an implementation for user accounts	50

F Git Commit Log

```

2019-11-05 (40494f4): Initial-commit
2019-11-10 (f58c0d8): Change-to-v0.0.1
2019-11-13 (243f769): Change-to-v0.0.2
2020-01-29 (47ababe): Initial-commit
2020-01-29 (3b762d7): Testing-git-GUI-implementation
2020-01-29 (cf6dfbd): Add-LICENSE
2020-01-29 (118912a): Add-CHANGELOG
2020-01-29 (1b8b295): Add-README.md
2020-01-29 (2d444c3): Update-README.md
2020-01-29 (e758272): Update-CHANGELOG
2020-01-29 (b656f60): Update-CHANGELOG
2020-01-29 (b3316b5): Delete-CHANGELOG
2020-01-29 (1a38660): Delete-key.pub
2020-01-29 (2d786d2): Delete-key
2020-01-30 (4dd6d62): Daily-Commit
2020-01-31 (c283cf9): Daily-Commit
2020-02-02 (e36c3b0): DateCheckFromLocalJSON
2020-02-11 (82fb78d): Daily-Commit
2020-02-11 (56ffb6c): Signed-off-by-ect1ul7-ect1ul7-soton.ac.uk
2020-02-12 (909ccb8): Daily-Commit
2020-02-18 (2a6923f): Changes-since-last-commit
2020-02-25 (6a060d3): Changes-since-last-commit
2020-02-28 (3bb955c): Changes-since-last-commit
2020-02-28 (382a89a): Changes-since-last-commit
2020-03-04 (1282c4a): Changes-since-last-commit
2020-03-17 (960651e): Changes-since-last-commit
2020-03-27 (ab2e8dd): Changes-since-last-commit
2020-04-02 (f952604): Changes-since-last-commit
2020-04-28 (a31ef85): Changes-since-last-commit
2020-04-28 (c007609): testing-commit-permissions
2020-04-29 (54843b4): Changes
2020-04-29 (aac76df): Updated-Readme
2020-04-29 (12ecc58): Updating-Readme
2019-11-05 (40494f4): Initial-commit
2019-11-10 (f58c0d8): Change-to-v0.0.1
2019-11-13 (243f769): Change-to-v0.0.2
2020-01-29 (47ababe): Initial-commit
2020-01-29 (3b762d7): Testing-git-GUI-implementation
2020-01-29 (cf6dfbd): Add-LICENSE
2020-01-29 (118912a): Add-CHANGELOG
2020-01-29 (1b8b295): Add-README.md
2020-01-29 (2d444c3): Update-README.md
2020-01-29 (e758272): Update-CHANGELOG
2020-01-29 (b656f60): Update-CHANGELOG
2020-01-29 (b3316b5): Delete-CHANGELOG
2020-01-29 (1a38660): Delete-key.pub
2020-01-29 (2d786d2): Delete-key
2020-01-30 (4dd6d62): Daily-Commit
2020-01-31 (c283cf9): Daily-Commit
2020-02-02 (e36c3b0): DateCheckFromLocalJSON
2020-02-11 (82fb78d): Daily-Commit
2020-02-11 (56ffb6c): Signed-off-by-ect1ul7-ect1ul7-soton.ac.uk
2020-02-12 (909ccb8): Daily-Commit
2020-02-18 (2a6923f): Changes-since-last-commit
2020-02-25 (6a060d3): Changes-since-last-commit
2020-02-28 (3bb955c): Changes-since-last-commit
2020-02-28 (382a89a): Changes-since-last-commit
2020-03-04 (1282c4a): Changes-since-last-commit
2020-03-17 (960651e): Changes-since-last-commit
2020-03-27 (ab2e8dd): Changes-since-last-commit
2020-04-02 (f952604): Changes-since-last-commit
2020-04-28 (a31ef85): Changes-since-last-commit
2020-04-29 (54843b4): Changes
2020-04-29 (12ecc58): Updating-Readme

```

G JSON Data File (Containing Box and Experiment Data for Figure 7)

G.i First of Data Shown in Figure 7

```
[{
  "Group": "Move",
  "Box": [{
    "Name": "Move Box",
    "Locked": true,
    "Image": null,
    "Experiments": [null]
  }]
},{
  "Group": "Eat",
  "Box": [{
    "Name": "Eat Box 1",
    "Locked": true,
    "Image": null,
    "Experiments": [null]
  }],{
  "Name": "Eat Box 2",
  "Locked": true,
  "Image": null,
  "Experiments": [null]
}]
},{
  "Group": "Engage",
  "Box": [{
    "Name": "Engage Box",
    "Locked": true,
    "Image": null,
    "Experiments": [null]
  }]
},{
  "Group": "Cogitate",
  "Box": [{
    "Name": "Cogitate Box",
    "Locked": true,
    "Image": null,
    "Experiments": [null]
  }]
},{
```

G.ii Second Section of the Data Shown in Figure 7

```
}, {
  "Group": "Sleep",
  "Box": [ {
    "Name": "SleepBetter",
    "Locked": false,
    "Image": null,
    "Description": "These are the next factors and associated experiments that impact your sleep hygiene. The chosen experiment is the one you chose to follow.",
    "WarningText": "Please bare in mind that after clicking Submit, you will only be able to change your experiment after 5 days.",
    "Experiments": [ {
      "Group": "Light",
      "Options": ["Increase bright light exposure during the day",
        "Wear glasses/a sleeping mask that block blue light during the night",
        "Turn off any bright lights 22 hours before going to bed (such as TV/the computer etc)"]
    }, {
      "Group": "Caffeine",
      "Options": ["Do not drink caffeine during the 6 hours before sleep",
        "Limit yourself to 4 cups of coffees per day; 10 cans of coke/fanta; 2 energy drinks",
        "Do not drink caffeine on an empty stomach"]
    }, {
      "Group": "Sleep Schedule",
      "Options": ["Usually get up at the same time everyday, even on weekends/vacations",
        "Sleep no less than 7 hours per night",
        "Do not go to bed unless you are tired. If you are not, relax with a bath/ by reading a book before bed",
        "Go to sleep at 22:30 PM at the latest"]
    }
  ]
}, {
  "Name": "Sleep Box 2",
  "Locked": true,
  "Image": null,
  "Description": "Test description!"
}
]
```

H Project Archive Files Table of Contents

H.i First Section of Archive

```

Y3Proj
|-- LICENSE                [General license agreement for anyone accessing the git repository]
|-- README.md             [A readme for explaining the project for the git repository]
|-- XBFramework.iml      [A project based dictionary (including any gradle / project settings files)]
|-- app                  [The folder that contains all the code for the project]
|  |-- app.iml           [Another app based dictionary (only for the code)]
|  |-- build             [An automatically generated build folder that holds temporary files during the build process]
|  |-- build.gradle      [An automatically generated build settings gradle file]
|  |-- libs              [An automatically generated library folder]
|  |-- proguard-rules.pro [A project settings file]
|  |-- src               [The source code folder]
|     |-- androidTest    [An automatically generated test folder]
|     |-- main           [The main source code folder]
|         |-- AndroidManifest.xml [A file containing the manifest for all the pages and permissions in the applicaiton]
|         |-- ic_launcher-web.png [The raw .png for the app logo]
|         |-- java
|             |-- com
|                 |-- yearthreeproject
|                     |-- xbframework [The actual java code (where all the dynamic features are written) anything with "Activity" is the pages code]
|                         |-- AboutActivity.java
|                         |-- BoxesActivity.java
|                         |-- ExperimentActivity.java
|                         |-- ExperimentSurveyCreatorActivity.java
|                         |-- LoginActivity.java
|                         |-- MainActivity.java
|                         |-- MyNotificationPublisher.java [A backend page used to push notifications even when the application is not running]
|                         |-- NewBoxActivity.java
|                         |-- ProgressActivity.java
|                         |-- ProjectMacros.java [A project defined library that holds functions used commonly throughout the project]
|                         |-- ShopActivity.java
|                         |-- SurveyResponseActivity.java
|                         |-- TestingActivity.java
|         |-- res        [A page only used in debugging phase, is not longer used in the final product]
|             |-- drawable [The resource folder]
|                 |-- ic_action_right.xml [Automatically generated XMLs for the launcher icon]
|                 |-- ic_launcher_background.xml
|                 |-- drawable-anydpi [Automatically generated XMLs for the launcher icon for different formats]

```

[Continued on the next page]

H.iii Third Section of Archive

```

Y3Proj
|-- app
|   |-- src
|   |   |-- res
|   |   |   |-- mipmap
|   |   |   |-- raw
|   |   |   |   |-- box.json
|   |   |   |   |-- box_outline.png
|   |   |   |   |-- local_user_data.json
|   |   |   |   |-- logo.png
|   |   |   |   |-- shop.json
|   |   |   |-- values
|   |   |   |   |-- arrays.xml
|   |   |   |   |-- colors.xml
|   |   |   |   |-- strings.xml
|   |   |   |-- styles.xml
|   |-- test
|-- build.gradle
|-- gradle
|   |-- wrapper
|   |   |-- gradle-wrapper.jar
|   |   |-- gradle-wrapper.properties
|-- gradle.properties
|-- gradlew
|-- gradlew.bat
|-- local.properties
|-- settings.gradle
|-- planning
|   |-- Bill of Materials.xlsx
|   |-- Daily progress.xlsx
|   |-- Gantt.xlsx
|   |-- Logbook.pdf
|   |-- Risk analysis.xlsx
|-- reports
|   |-- final
|   |   |-- Project Report Body.docx
|   |   |-- Project Report Body.pdf
|   |   |-- Project Report.docx
|   |   |-- Project Report.pdf
|   |-- project brief
|   |   |-- Project Brief.docx
|   |-- progress
|   |   |-- Progress Report Body.docx
|   |   |-- Progress Report.docx
|   |   |-- Progress Report.pdf
|   |   |-- ShorterReportBody.pdf
|-- designs
|   |-- saltshaker.png

```

[Raw files used for initialising different objects within the application]

[The original local box data structure, no longer used]

[A two bit image of the logo for notifications]

[The original local user data data structure, no longer used]

[The complete logo for the application]

[The shop data structure, used to generate the display in the shop page]

[Project wide files for static resource]

[Contains a MEECS array, used for a spinner and data verification]

[The defined colour scheme for the project]

[A list of static strings to use, the standard way to code GUI in android]

[A description of the default look of GUI elements within the applicaiton]

[An automatically generated folder]

[A project wide description of the libraries used and build settings]

[An automatically generated folder for project settings]

[More project settings]

[A folder containing all the supplementary planning documentation for this project]

[A scanned copy of the logbook entries used for this project]

[A folder containing the two reports written for this project]

[A folder contianing the final report]

[A folder contianing the initial project brief]

[A folder containing the progress report]

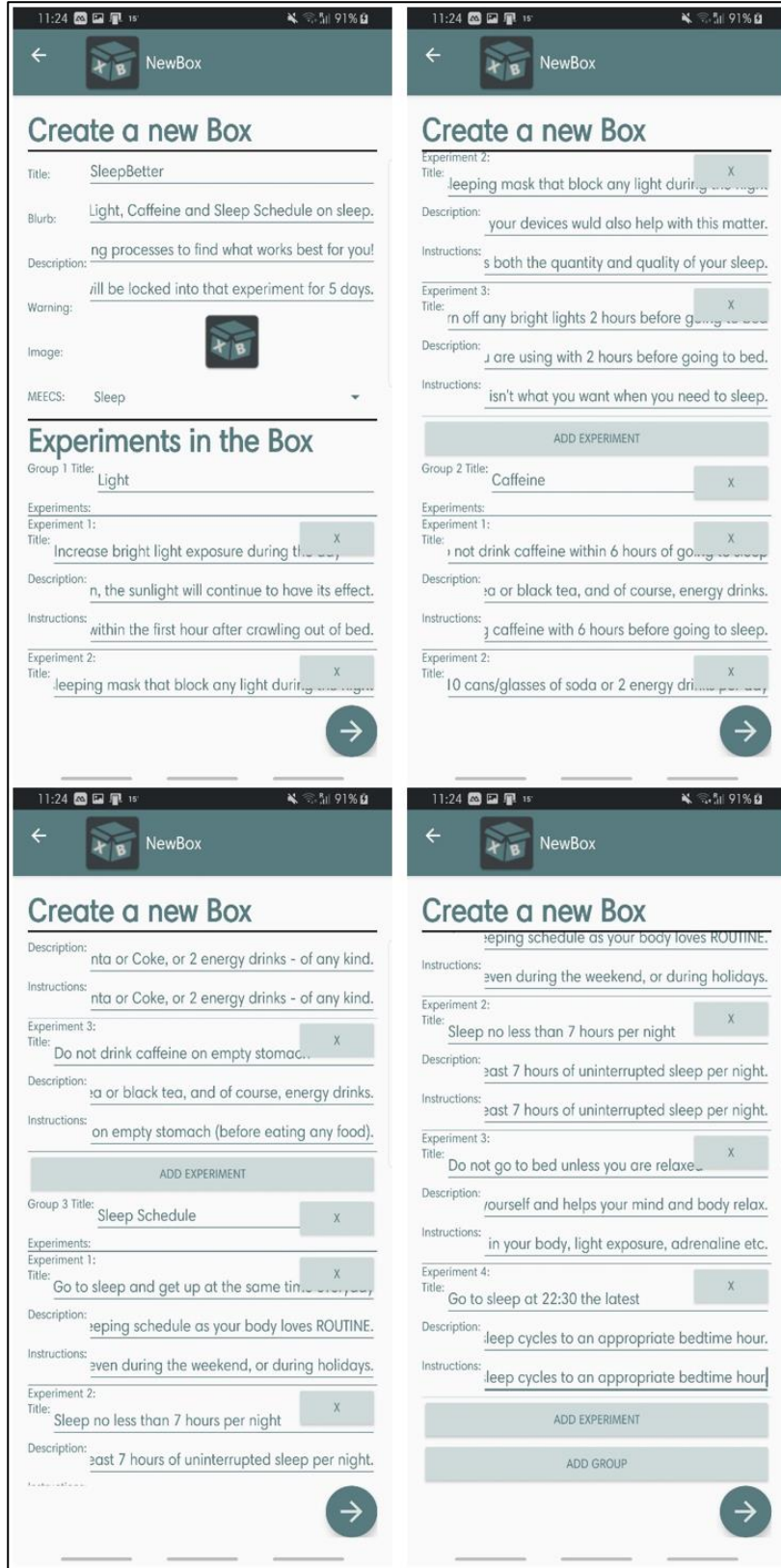
[A folder containing the design files relating to this project]

I Database Syncing Scenarios

Scenario	Data structure to sync	Upload or Download	Purpose
Opening the Box Page	Box	Download	Make sure the boxes shown are up to date
Opening the Progress Page	User Data	Download	Make sure the progress data shown is up to date
Finished adding all the information to a new box	Box	Upload	Ensure that the submission can be seen by a moderator
Added a journal entry in the progress page	User Data	Upload	Ensure that if the application is closed a re-opened, the data would appear
Added a response to the survey	User Data	Upload	Ensure that if the application is closed a re-opened, the data would appear
Opening the application	User Data	Download	Ensuring that any notifications or locked status of the user are correctly updated
Logging in	User Data	Download	Ensuring the new login session has the most up to date information
Logging out	User Data	Delete local entry	Ensure that no user data remains on the device after logging out of the application

J Creation of SleepBetter XB in the XB Framework

J.i SleepBetter Box Creation



J.ii Light – “Increase Bright Light Exposure” Experiment Creation

11:36 16° 92%

← SleepBetter

According to National Sleep Foundations, sleep hygiene is a variety of different practices and habits that are necessary to have good night sleep quality and full daytime alertness. Obtaining healthy sleep is important for both physical and mental health. It can also improve productivity and overall quality of life. Everyone, from children to older adults, can benefit from developing a good sleep practice! This box will help you explore tuning processes to find what works best for you!
Once an experiment is chosen, will be locked into that experiment for 5 days.

Light
 Increase bright light exposure during the day
 It is best to spend 30 to 45 minutes getting direct sunlight exposure into the eyes; Spend at least half an hour in the sun during the day as well, by exercising, walking or just any other sort of activity; It is recommended that your bedroom can capture the sunlight during the day - leaving your curtains open; Moreover, don't worry if it's a cloudy day, because even when filtered through clouds or rain, the sunlight will continue to have its effect.

Survey

Question 1: It took me a long time to fall asleep

Question 2: I wake up many times in the night X

Question 3: I wake up much earlier than I intended X

Question 4: I had a good quality of sleep X

Question 5: Sleep has negatively impacted my mood, energy, or relationships X

Question 6: Sleep has negatively impacted my concentration, productivity, or at X

Question 7: Sleep negatively impacts my life in general X

Question 8:

Question 5: Sleep has negatively impacted my mood, energy, or relationships X

Question 6: Sleep has negatively impacted my concentration, productivity, or at X

Question 7: Sleep negatively impacts my life in general X

Question 10: My room captured sunlight today X

ADD QUESTION

Only answerable at a specific time?

Notifications

Notification 1:
 Notification Title: Go out into the sun!
 Notification Message: Now is the best time to go out into the sun (if it's there) to time the best reinforcement to your circadium rhythm!

Time: 11:30 Repeat Every: 00:00 hours

ADD NOTIFICATION

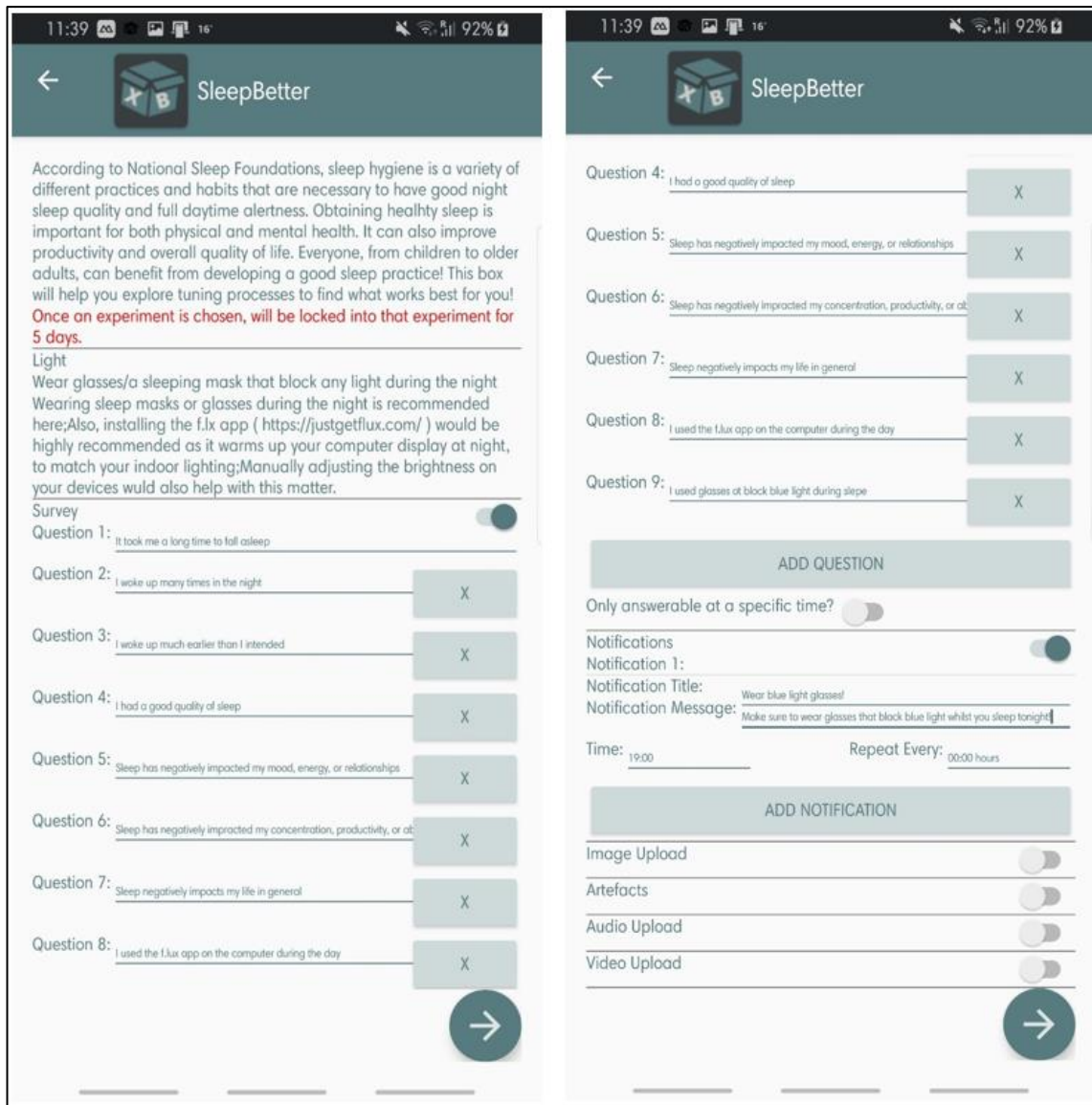
Image Upload

Artefacts

Audio Upload

Video Upload

J.iii Light – “Blue Light Blocker” Experiment Creation



J.iv Light – “Turn Off Any Bright Lights Two Hours Before Going To Bed” Experiment Creation

11:43 16 93%

← SleepBetter

According to National Sleep Foundations, sleep hygiene is a variety of different practices and habits that are necessary to have good night sleep quality and full daytime alertness. Obtaining healthy sleep is important for both physical and mental health. It can also improve productivity and overall quality of life. Everyone, from children to older adults, can benefit from developing a good sleep practice! This box will help you explore tuning processes to find what works best for you! **Once an experiment is chosen, will be locked into that experiment for 5 days.**

Light
Turn off any bright lights 2 hours before going to bed
Therefore, try turning off the TV, the computer, or any other LED-light device you are using with 2 hours before going to bed.

Survey

Question 1: It took me a long time to fall asleep

Question 2: I wake up many times in the night

Question 3: I wake up much earlier than I intended

Question 4: I had a good quality of sleep

Question 5: Sleep has negatively impacted my mood, energy, or relationships

Question 6: Sleep has negatively impacted my concentration, productivity, or at

Question 7: Sleep negatively impacts my life in general

Question 8: I watched tv or stayed on the computer within 2 hours of going to bed

Question 9: I turned off bright lights 2 hours before going to bed

→

11:43 16 93%

← SleepBetter

Question 4: I had a good quality of sleep

Question 5: Sleep has negatively impacted my mood, energy, or relationships

Question 6: Sleep has negatively impacted my concentration, productivity, or at

Question 7: Sleep negatively impacts my life in general

Question 8: I watched tv or stayed on the computer within 2 hours of going to bed

Question 9: I turned off bright lights 2 hours before going to bed

ADD QUESTION

Only answerable at a specific time?

Notifications

Notification 1:
Notification Title: Turn off bright light!
Notification Message: You should start turning off bright lights so that you can have at least 2 hours cooldown before sleeping!

Time: 18:30 Repeat Every: 00:00 hours

ADD NOTIFICATION

Image Upload

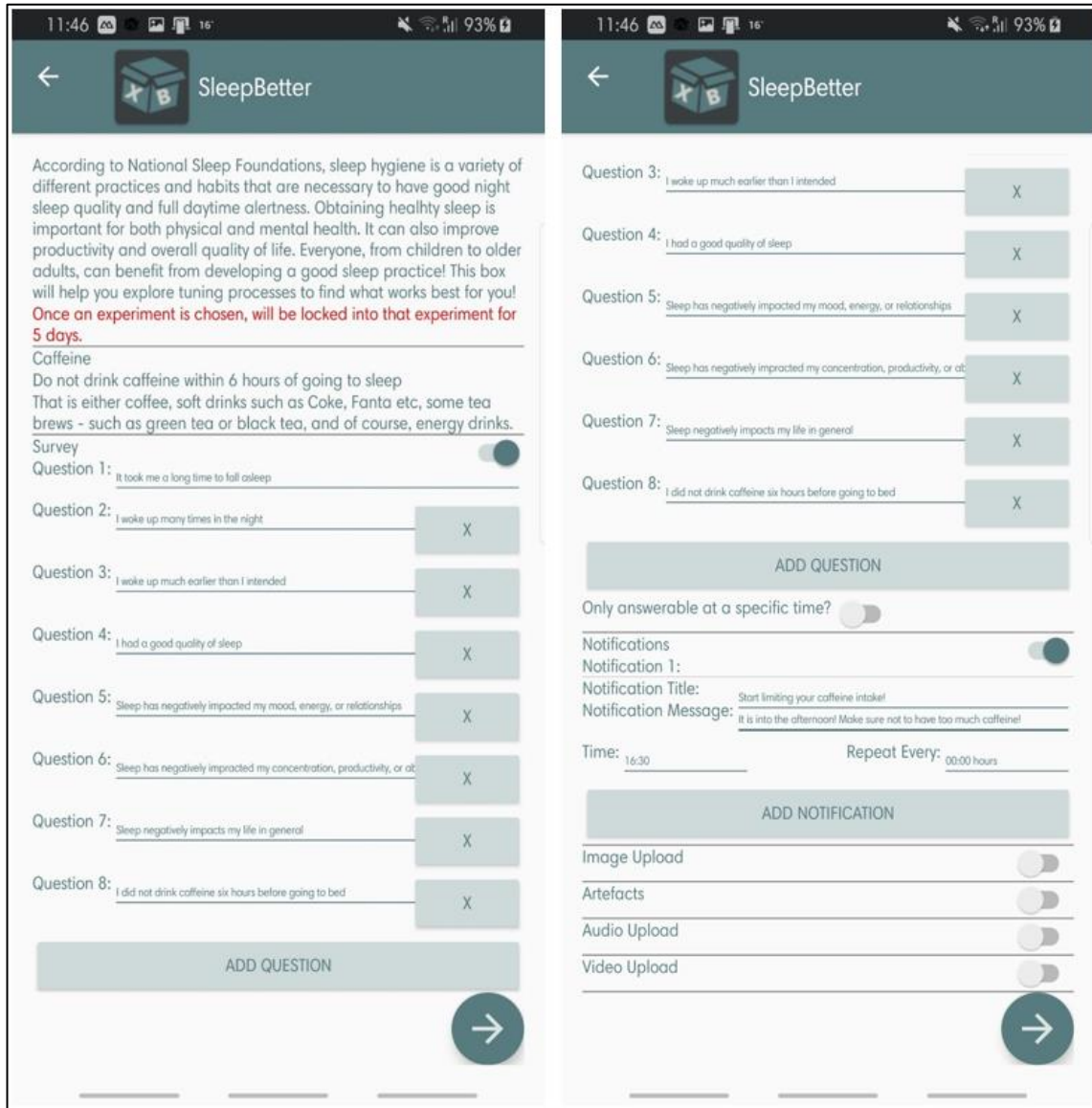
Artefacts

Audio Upload

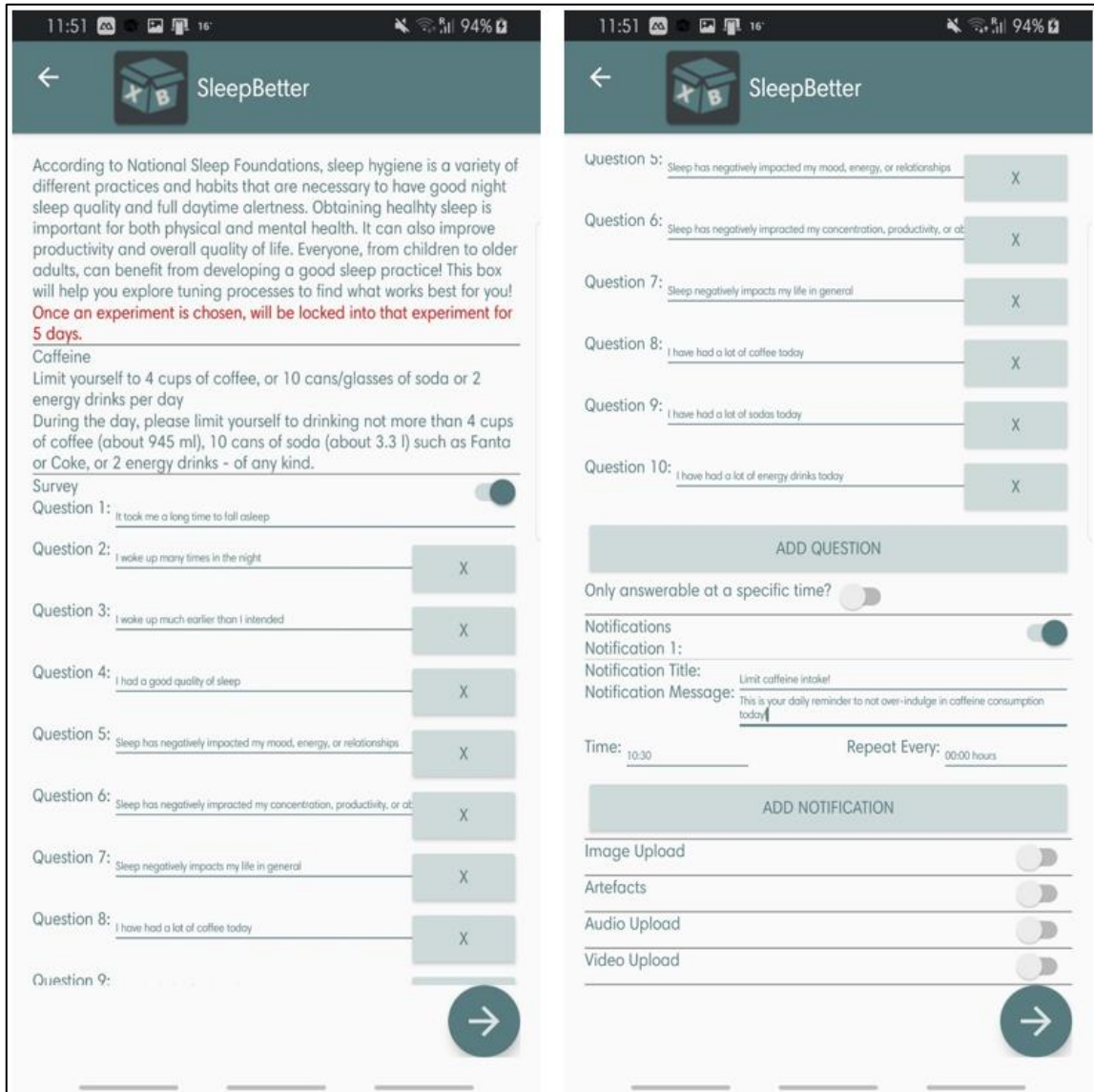
Video Upload

→

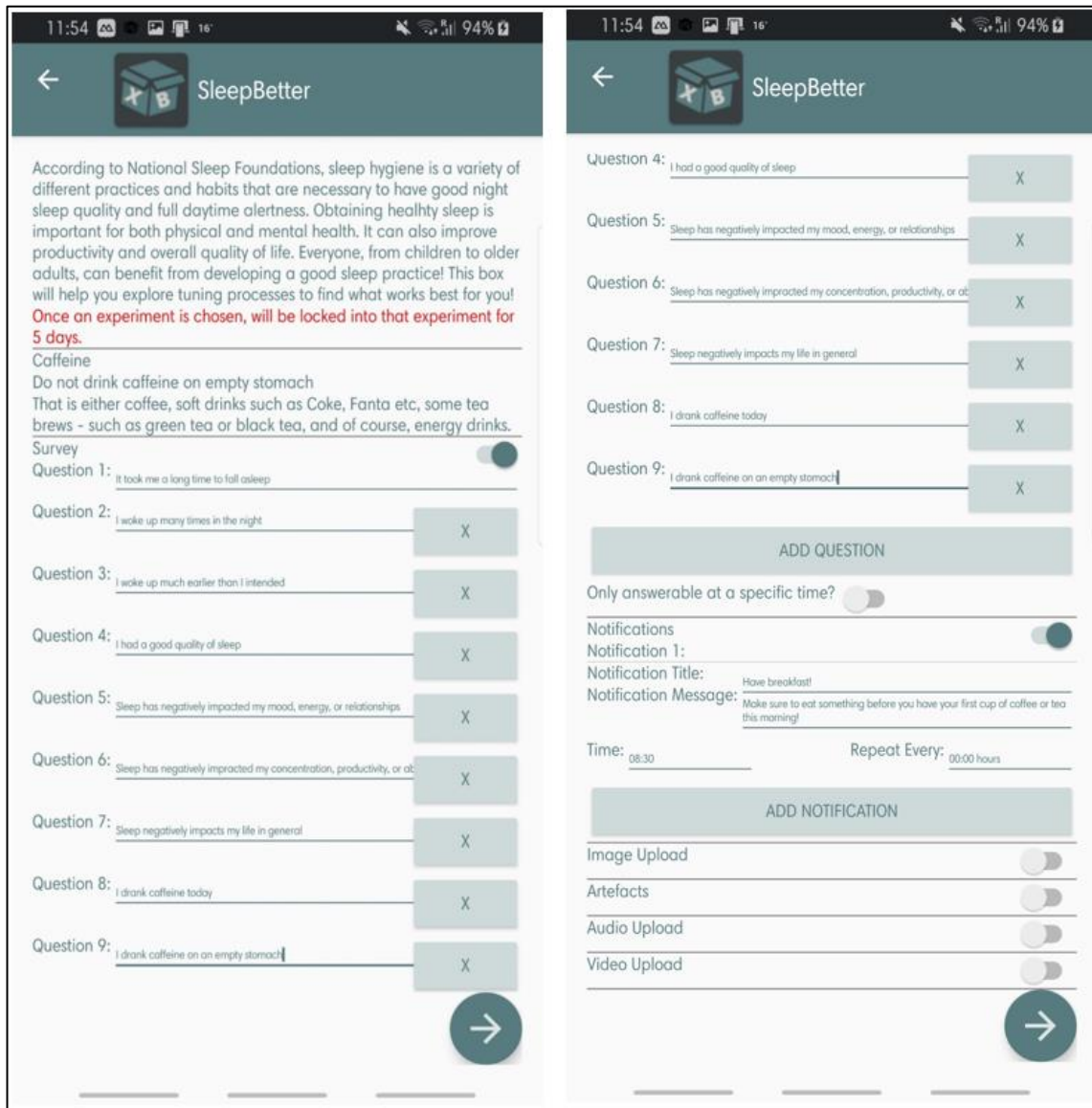
J.v Caffeine – “Do Not Drink Caffeine Within Six Hours of Going to Sleep” Experiment Creation



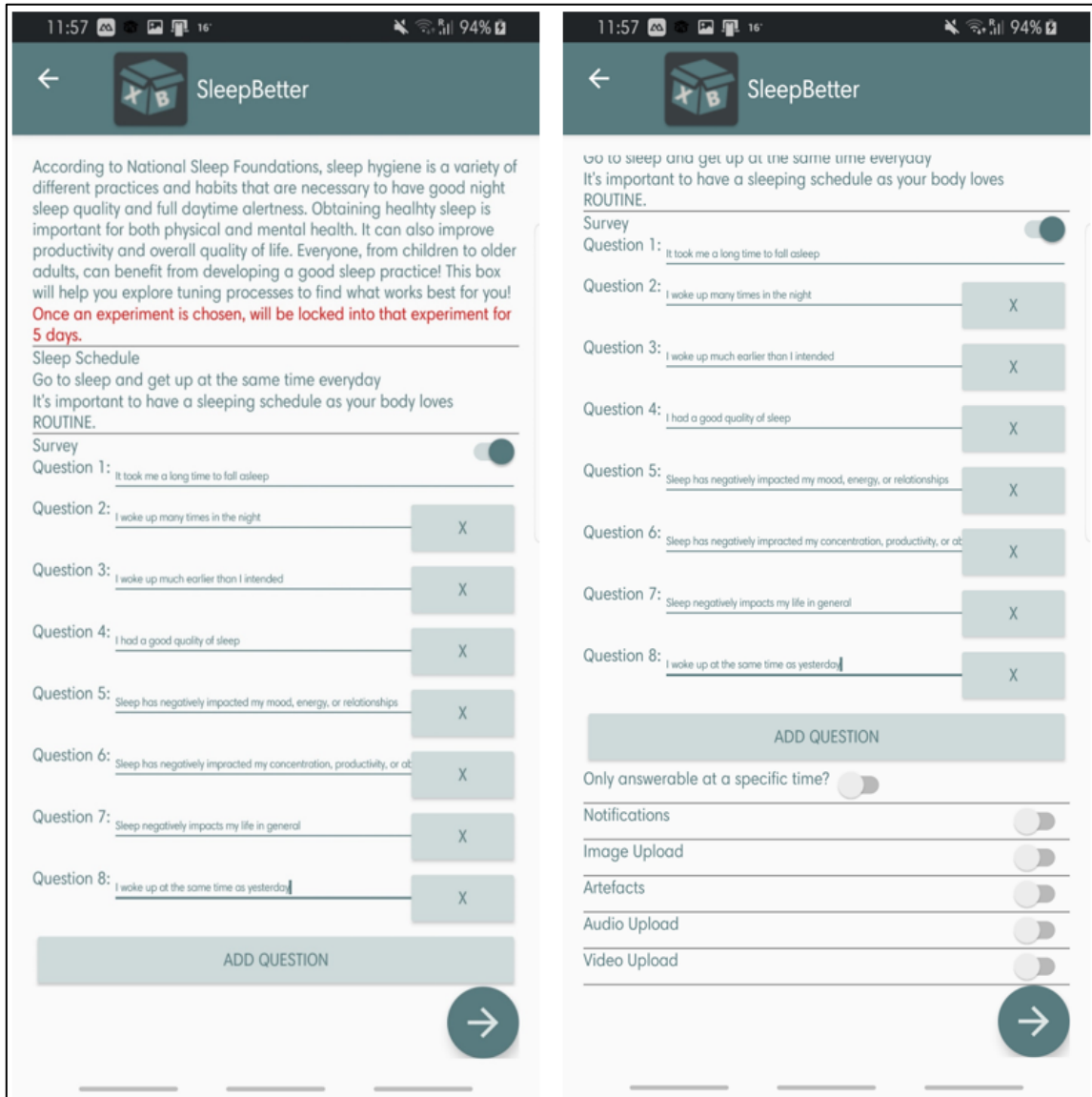
J.vi Caffeine – “Limit Caffeine Intake” Experiment Creation



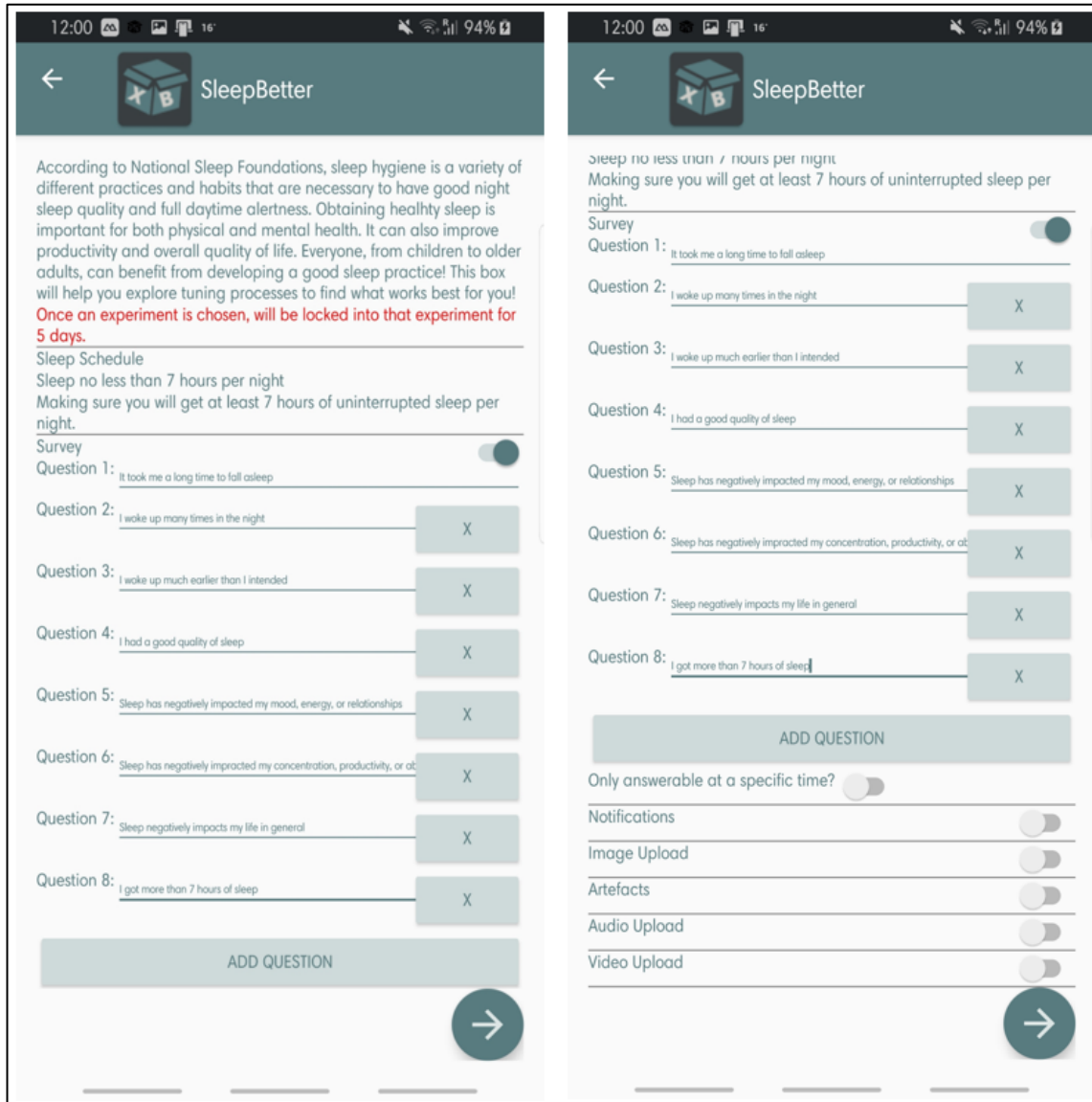
J.vii Caffeine – “Do Not Drink Caffeine on An Empty Stomach” Experiment Creation



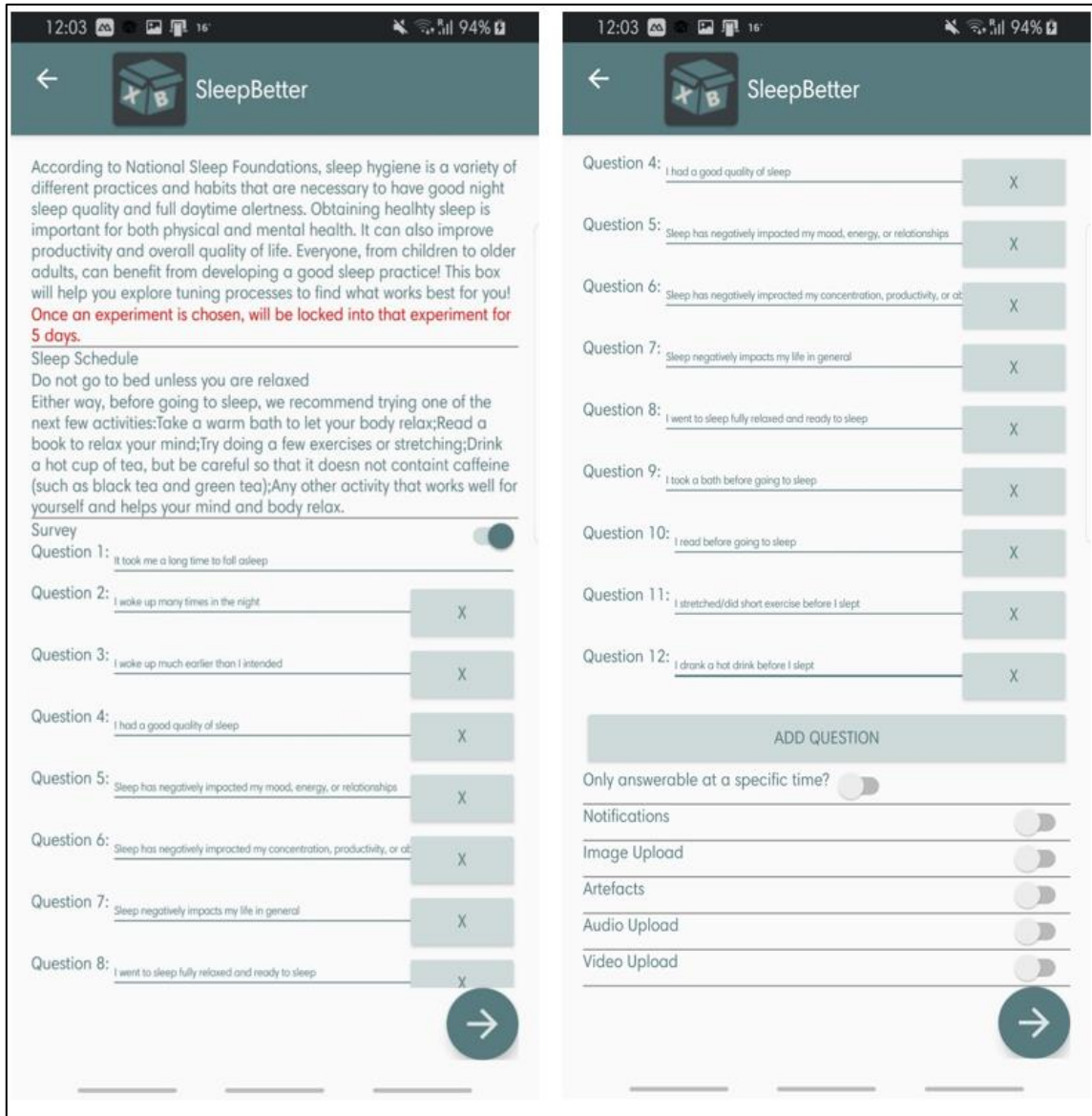
J.viii Sleep Schedule – “Go to Sleep and Wake Up at the Same Time Every Day” Experiment Creation



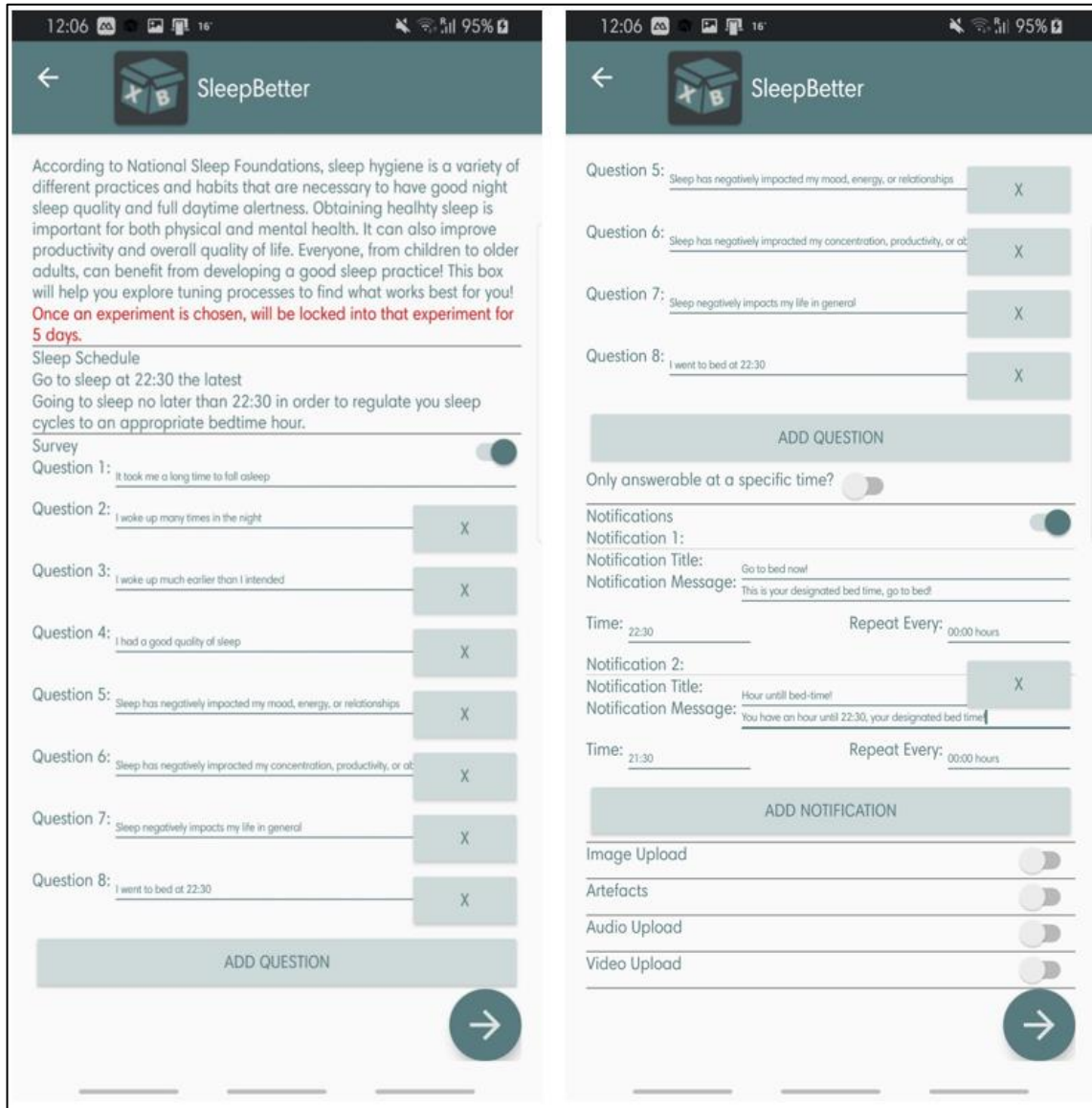
J.ix Sleep Schedule – “Sleep No Less Than 7 Hours Per Night” Experiment Creation



J.x Sleep Schedule – “Do Not Go to Bed Unless You Are Relaxed” Experiment Creation Process



J.xi Sleep Schedule – “Go to Sleep at 22:30 at the Latest” Experiment Creation



J.xii Data Structure Produced (First Section)

```

_id: ObjectId("5e820f7b75f17256943a747b")
Name: "SleepBetter"
Group: "Sleep"
Blurb: "A box that explores the effects of Light, Caffeine and Sleep Schedule ..."
Approved: true
Image: null
Description: "According to National Sleep Foundations, sleep hygiene is a variety of..."
WarningText: "Once an experiment is chosen, will be locked into that experiment for ..."
Experiments: Array
  0: Object
    Group: "Light"
    Options: Array
      0: Object
        Title: "Increase bright light exposure during the day"
        Instructions: "Exposure to sunlight should occur in the morning upon awakening, typic..."
        Description: "It is best to spend 30 to 45 minutes getting direct sunlight exposure ..."
        VideoProcessing: false
        AudioProcessing: false
        ImageProcessing: false
        SurveyQuestions: Array
          0: "It took me a long time to fall asleep"
          1: "I woke up many times in the night"
          2: "I woke up much earlier than I intended"
          3: "I had a good quality of sleep"
          4: "Sleep has negatively impacted my mood, energy, or relationships"
          5: "Sleep has negatively impracted my concentration, productivity, or abil..."
          6: "Sleep negatively impacts my life in general"
          7: "I got strong sunlight exposure in the morning"
          8: "I got a long time in the sun today"
          9: "My room caputred sunlight today"
        Notification: Array
          0: Object
            Title: "Go out into the sun!"
            Message: "Now is the best time to go out into the sun (if it's there) to time th..."
            Time: "11:30"
            Repeat: "24:00 hours"
            ArtefactImplementation: false
      1: Object
        Title: "Wear glasses/a sleeping mask that block any light during the night"
        Instructions: "Blue light - coming from LED screens or even from the sun - in the eve..."
        Description: "Wearing sleep masks or glasses during the night is recommended here;
          A..."
        VideoProcessing: false
        AudioProcessing: false
        ImageProcessing: false
        SurveyQuestions: Array
          0: "It took me a long time to fall asleep"
          1: "I woke up many times in the night"
          2: "I woke up much earlier than I intended"
          3: "I had a good quality of sleep"
          4: "Sleep has negatively impacted my mood, energy, or relationships"
          5: "Sleep has negatively impracted my concentration, productivity, or abil..."
          6: "Sleep negatively impacts my life in general"
          7: "I used the f.lux app on the computer during the day"
          8: "I used glasses ot block blue light during slepe"
        Notification: Array
          0: Object
            Title: "Wear blue light glasses!"
            Message: "Make sure to wear glasses that block blue light whilst you sleep tonig..."
            Time: "19:00"
            Repeat: "24:00 hours"
            ArtefactImplementation: false
      2: Object
        Title: "Turn off any bright lights 2 hours before going to bed"
        Instructions: "Light makes you feel alert, which isn't what you want when you need to..."
        Description: "Therefore, try turning off the TV, the computer, or any other LED-ligh..."
        VideoProcessing: false
        AudioProcessing: false
        ImageProcessing: false
        SurveyQuestions: Array
          0: "It took me a long time to fall asleep"
          1: "I woke up many times in the night"
          2: "I woke up much earlier than I intended"
          3: "I had a good quality of sleep"
          4: "Sleep has negatively impacted my mood, energy, or relationships"
          5: "Sleep has negatively impracted my concentration, productivity, or abil..."
          6: "Sleep negatively impacts my life in general"
          7: "I watched tv or stayed on the computer within 2 hours of going to bed"
          8: "I turned off bright lights 2 hours before going to bed"
        Notification: Array
          0: Object
            Title: "Turn off bright lights!"
            Message: "You should start turning off bright lights so that you can have at lea..."
            Time: "18:30"
            Repeat: "24:00 hours"
            ArtefactImplementation: false
    
```

J.xiii Data Structure Produced (Second Section)

```

~ 1: Object
  Group: "Caffeine"
  ~ Options: Array
    ~ 0: Object
      Title: "Do not drink caffeine within 6 hours of going to sleep"
      Instructions: "It is highly recommended not drinking any product containing caffeine ..."
      Description: "That is either coffee, soft drinks such as Coke, Fanta etc, some tea b..."
      VideoProcessing: false
      AudioProcessing: false
      ImageProcessing: false
    ~ SurveyQuestions: Array
      0: "It took me a long time to fall asleep"
      1: "I woke up many times in the night"
      2: "I woke up much earlier than I intended"
      3: "I had a good quality of sleep"
      4: "Sleep has negatively impacted my mood, energy, or relationships"
      5: "Sleep has negatively impracted my concentration, productivity, or abil..."
      6: "Sleep negatively impacts my life in general"
      7: "I did not drink caffeine six hours before going to bed"
    ~ Notification: Array
      ~ 0: Object
        Title: "Start limiting your caffeine intake!"
        Message: "It is into the afternoon! Make sure not to have too much caffeine!"
        Time: "16:30"
        Repeat: "24:00 hours"
      ArtefactImplementation: false
    ~ 1: Object
      Title: "Limit yourself to 4 cups of coffee, or 10 cans/glasses of soda or 2 en..."
      Instructions: "During the day, please limit yourself to drinking not more than 4 cups..."
      Description: "During the day, please limit yourself to drinking not more than 4 cups..."
      VideoProcessing: false
      AudioProcessing: false
      ImageProcessing: false
    ~ SurveyQuestions: Array
      0: "It took me a long time to fall asleep"
      1: "I woke up many times in the night"
      2: "I woke up much earlier than I intended"
      3: "I had a good quality of sleep"
      4: "Sleep has negatively impacted my mood, energy, or relationships"
      5: "Sleep has negatively impracted my concentration, productivity, or abil..."
      6: "Sleep negatively impacts my life in general"
      7: "I have had a lot of coffee today"
      8: "I have had a lot of sodas today"
      9: "I have had a lot of energy drinks today"
    ~ Notification: Array
      ~ 0: Object
        Title: "Limit caffeine intake!"
        Message: "This is your daily reminder to not over-indulge in caffeine consumptio..."
        Time: "10:30"
        Repeat: "24:00 hours"
      ArtefactImplementation: false
    ~ 2: Object
      Title: "Do not drink caffeine on empty stomach"
      Instructions: "After you wake up, we highly recommend not drinking any products conta..."
      Description: "That is either coffee, soft drinks such as Coke, Fanta etc, some tea b..."
      VideoProcessing: false
      AudioProcessing: false
      ImageProcessing: false
    ~ SurveyQuestions: Array
      0: "It took me a long time to fall asleep"
      1: "I woke up many times in the night"
      2: "I woke up much earlier than I intended"
      3: "I had a good quality of sleep"
      4: "Sleep has negatively impacted my mood, energy, or relationships"
      5: "Sleep has negatively impracted my concentration, productivity, or abil..."
      6: "Sleep negatively impacts my life in general"
      7: "I drank caffeine today"
      8: "I drank caffeine on an empty stomach"
    ~ Notification: Array
      ~ 0: Object
        Title: "Have breakfast!"
        Message: "Make sure to eat something before you have your first cup of coffee or..."
        Time: "08:30"
        Repeat: "24:00 hours"
      ArtefactImplementation: false

```


J.xiv Data Structure Produced (Third Section)

```

v 2: Object
  Group: "Sleep Schedule"
  Options: Array
    v 0: Object
      Title: "Go to sleep and get up at the same time everyday"
      Instructions: "Going to bed and waking up at the same time everyday, even during the ..."
      Description: "It's important to have a sleeping schedule as your body loves ROUTINE."
      VideoProcessing: false
      AudioProcessing: false
      ImageProcessing: false
      SurveyQuestions: Array
        0: "It took me a long time to fall asleep"
        1: "I woke up many times in the night"
        2: "I woke up much earlier than I intended"
        3: "I had a good quality of sleep"
        4: "Sleep has negatively impacted my mood, energy, or relationships"
        5: "Sleep has negatively impracted my concentration, productivity, or abil..."
        6: "Sleep negatively impacts my life in general"
        7: "I woke up at the same time as yesterday"
      Notification: Array
      ArtefactImplementation: false
    v 1: Object
      Title: "Sleep no less than 7 hours per night"
      Instructions: "Making sure you will get at least 7 hours of uninterrupted sleep per n..."
      Description: "Making sure you will get at least 7 hours of uninterrupted sleep per n..."
      VideoProcessing: false
      AudioProcessing: false
      ImageProcessing: false
      SurveyQuestions: Array
        0: "It took me a long time to fall asleep"
        1: "I woke up many times in the night"
        2: "I woke up much earlier than I intended"
        3: "I had a good quality of sleep"
        4: "Sleep has negatively impacted my mood, energy, or relationships"
        5: "Sleep has negatively impracted my concentration, productivity, or abil..."
        6: "Sleep negatively impacts my life in general"
        7: "I got more than 7 hours of sleep"
      Notification: Array
      ArtefactImplementation: false
    v 2: Object
      Title: "Do not go to bed unless you are relaxed"
      Instructions: "It can happen very often that you don't feel sleepy close to bedtime. ..."
      Description: "Either way, before going to sleep, we recommend trying one of the next..."
      VideoProcessing: false
      AudioProcessing: false
      ImageProcessing: false
      SurveyQuestions: Array
        0: "It took me a long time to fall asleep"
        1: "I woke up many times in the night"
        2: "I woke up much earlier than I intended"
        3: "I had a good quality of sleep"
        4: "Sleep has negatively impacted my mood, energy, or relationships"
        5: "Sleep has negatively impracted my concentration, productivity, or abil..."
        6: "Sleep negatively impacts my life in general"
        7: "I went to sleep fully relaxed and ready to sleep"
        8: "I took a bath before going to sleep"
        9: "I read before going to sleep"
        10: "I stretched/did short exercise before I slept"
        11: "I drank a hot drink before I slept"
      Notification: Array
      ArtefactImplementation: false
    v 3: Object
      Title: "Go to sleep at 22:30 the latest"
      Instructions: "Going to sleep no later than 22:30 in order to regulate you sleep cycl..."
      Description: "Going to sleep no later than 22:30 in order to regulate you sleep cycl..."
      VideoProcessing: false
      AudioProcessing: false
      ImageProcessing: false
      SurveyQuestions: Array
        0: "It took me a long time to fall asleep"
        1: "I woke up many times in the night"
        2: "I woke up much earlier than I intended"
        3: "I had a good quality of sleep"
        4: "Sleep has negatively impacted my mood, energy, or relationships"
        5: "Sleep has negatively impracted my concentration, productivity, or abil..."
        6: "Sleep negatively impacts my life in general"
        7: "I went to bed at 22:30"
      Notification: Array
      v 0: Object
        Title: "Go to bed now!"
        Message: "This is your designated bed time, go to bed!"
        Time: "22:30"
        Repeat: "24:00 hours"
      v 1: Object
        Title: "Hour untill bed-time!"
        Message: "You have an hour until 22:30, your designated bed time!"
        Time: "21:30"
        Repeat: "24:00 hours"
      ArtefactImplementation: false

```

K Project Brief

Student Name: Eamonn Trim

Supervisor Name: m.c. schraefel

Project Title:

Experiment in a Box (XB) Framework

Problem:

Experiment in a Box (XB) is a methodology to support people building personal health practices by testing out how well a given health heuristic for that practice works for them. An XB provides a variety of heuristics to explore over a short time. Participants are guided in how to record the effects of that intervention. To date, XB's have been instantiated in individual applications (eg, the SleepBetterXB from the University of Southampton, which is an XB to explore sleep practices). There are issues with this approach:

- The researcher must be knowledgeable in coding (or employ someone to produce the application for them)
- Each new launch of an XB requires new user uptake to an application
- Each new launch of an XB requires a new workflow to manage the data received

Goals:

Develop an application framework that allows developers to quickly launch XBs to users who will have a central location to select XBs in which to participate. This app will:

- Act as a hub for data for both user and researcher
- Be an access point for participants to join an XB (through an approved download feature)
- Enable participants to browse XB artefacts that aid in data collection
- Allow participants to have an overview of their own data

Scope:

This project will test how to design and deploy a single point of access for researchers with little coding knowledge to develop their XBs and collect the data from their ongoing XBs. In correspondence with this, users should be capable of selecting XBs they wish to participate in.

To support this goal given the limitations of downloading functionality within an application, socket-like design, where core functionality of the XBs are downloaded with the core app, but the specific utilisation of these tools is specified as data structures to be downloaded as XBs from the researcher's creation. This would also require a secure way to store these data structures along side the data given by the participants. In addition to this, the application should also look to allow the transfer of data / packages from already existing XB applications to the new single access solution.

