Designing and Developing a Time Management App For and With Informatics Students

Julia Castillo Trujillo

MInf Project (Part 2) Report

Master of Informatics
School of Informatics
University of Edinburgh

2021

Abstract

As students transition from high school to university, many have issues adapting to a new unstructured type of schedule, where most of their time is spent on independent study. This makes it harder for students to manage their time in a way that suits their needs. This report follows the first part of this project, where I designed and evaluated a time management app for Informatics students at the University of Edinburgh. In the second part of the project, I redesigned some of the functionalities and implemented the app including the newly designed functionalities and requirements gathered during the first part. The app was evaluated by students of Informatics as well as a few students outside of the School of Informatics. The app received good feedback from the students and many showed interest in using it once it is made available to the public.

Table of Contents

Chapter 1

Introduction

1.1 Motivation

Starting university for the first time can be difficult for many students, as this transition translates into new responsibilities and an unknown educational system which mainly consists of independent study [60]. According to the students and staff members I interviewed last year, having an unstructured schedule causes students to have a schedule driven by coursework, which leads them to skip other responsibilities or even skip sleeping, which can greatly affect their mental health as well as their academic performance.

Having been through this transition myself, I know how important it is to have the right tools to help you out. This is why last year I decided to design a time management app for students. I started by doing research on time management and the best tools for students, through literature review as well by attending workshops organised by the Institute of Academic Development (IAD) [18] and the Counselling Services at the University of Edinburgh [40]. I met with 15 students to discuss what exactly they would require form a time management app that is catered to their needs. With the results of my research and the interviews, I designed a system that would help students manage their schedule, manage their tasks, track their time and manage their coursework. I created two iterations of the design for this app, which were followed by an evaluation each. This design included several features, such as a calendar, a task manager, a time-tracker, a coursework and exams manager and a reward system. The final evaluation showed that this system was promising, but there were a few areas for improvement. For example, some students liked the design of the time-tracker, but many other thought the design of that functionality needed changes, which made me wonder what the best design for such a functionality would be. Furthermore, I still needed to find out how I would implement this app in a way that suits Informatics students while being easy to use as well as consider the remaining requirements and suggestions resulting from the studies.

Additionally, I focused on the needs of Informatics students last year, as I wanted the design to cater for them first. However, all university students have time management problems, so this year I wanted to see if this app could be used by students of any

discipline and if it could potentially help them with their own time management issues.

1.2 Research Goals

The aim for this project is to continue the work from last year by implementing an app for Informatics students that cover their needs in terms of time management and to find out if the same app could be used by students.

This aim can be broken down into the following research questions:

1. How can we design a timer functionality and enhance the time-tracking functionality design, which had inconclusive outcomes last year, so that it suits Informatics students at the University of Edinburgh?
2. What are the technologies that we can use to implement this app?
3. How is the usability of the app perceived by Informatics students?
4. How is the usability of this app perceived by student of other disciplines at the University of Edinburgh?
5. What is the potential impact of the app for students, in terms of:
   (a) keeping up with their courses?
   (b) meeting deadlines?
   (c) reducing procrastination?
   (d) managing their personal life?

1.3 Dissertation structure

The dissertation is divided into seven chapters and is structured as follows.

In chapter 2, I summarise the project from last year as well as its results. Furthermore, I review a few time management tools, design methods and app development tools.

In chapter 3, I describe the methodology I used for each step of the project.

In chapter 4, I discuss how I worked with students to create a design for the timer and time-tracking functionality.

In chapter 5, I explain how I developed the app, what tools I used, and the issues I ran into during the process.

In chapter 6, I discuss the process and results of the evaluation of the app.

In chapter 7, I present my thoughts on how this project went, the issues I encountered and what the next steps would be.

Chapter 2

Background

2.1 Introduction to time management

As I found out last year, time management does not seem to have one set definition, but as I did last year, I will be following the definition of time management suggested by Claessens *et al.* in their review of time management literature [51], which describes time management as a set of "behaviours that aim at achieving an effective use of time while performing certain goal-directed activities".

Time management practices have not only been proven to help people deal with their workload, but also to help reduce stress, especially among university students [61][51][63].

2.2 Previous work carried out

In the first part of this project, I focused on designing a time management app that would suit Informatics students. There were three phases: requirements gathering, first iteration of the design and second iteration. I started by conducting interviews with students to gather requirements for the design of the app, which I then used to create two iterations of the design, each followed by an evaluation.

2.2.1 Final design

In this section, I will describe the final design I created using some of the requirements and suggested changes gathered during the requirements gathering and first evaluation, which are all presented in tables A.1 and A.2 in appendix A and table B.1 in appendix B.

The home screen (see figure 2.1 (a)) is comprised of two sections: upcoming events and today's tasks. I designed the home screen this way as I thought it would be a good way to show useful information for the user at a glance. At the bottom left, I added an add button to add events and tasks from the home screen as this was one of the suggested changes (see #3 in table B.1).

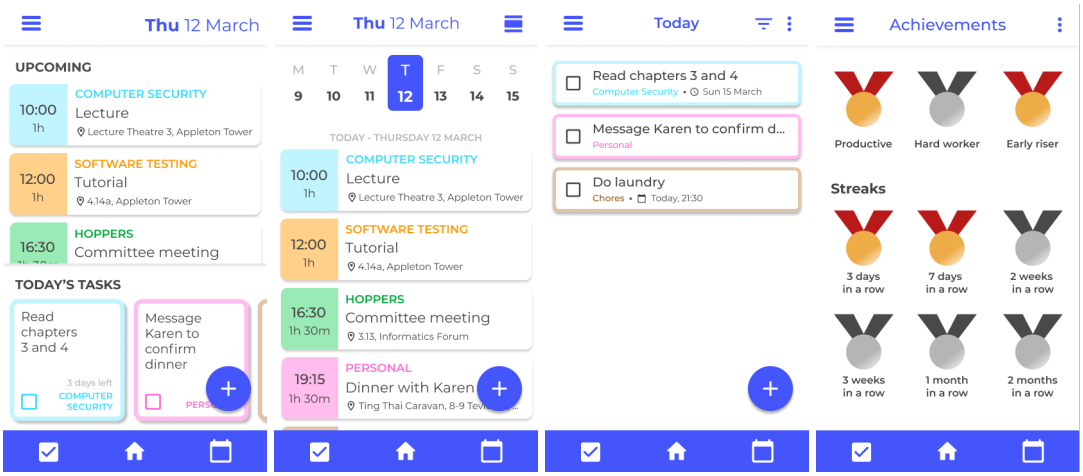


Figure 2.1: (a) Home screen (b) Calendar screen (c) Tasks screen (d) Achievements screen

The calendar screen (see figure 2.1 (b)) displays all the events the user has. It has a small date picker at the top of the screen so the user can choose a date and quickly see the events for that date. One of the main requirements was to have a calendar system (see #1 in table A.1). The tasks screen (see figure 2.1 (c)) displays the tasks for the user. Having to-do lists was also one of the main requirements (see #10 in table A.1). Each task card had a checkbox to be able to check the tasks off, which was another requirement (see #12 in table A.1). Both screens have an add button on the bottom right corner as well to add new events or tasks, respectively.

The event cards (seen in figure 2.1 (a) and (b)) show the title of the event, the category and another detail (location or notification) on the right, and the start time of the event as well as the duration of the event on the left. I thought this would be a good way to display the details of the events clearly. The task cards (seen in figure 2.1 (a) and (c)), although different in the home screen and the task screen, both show a checkbox, the title of the task, the category and another piece of information (deadline, reminder or date and time on the calendar).

The achievements screen (see figure 2.1 (d)) displays the rewards the user has won by using the app and doing things like completing tasks and using the time-tracker. To have a reward system was also one of the original requirements (see #22 in table A.1).

The time-tracker screen (see figure 2.2 (a)) presents in the top half of the screen the time that has passed, the label and category for that sessions, and controls: a big round button to start and pause the tracker, a stop button to finish the session and a *Take a break* button to allow the user to take a break. In the bottom half of the screen, it displays past sessions. In the top of the screen there are tabs, which let the user navigate between the tracker screen and the statistics screen. The statistics screen (see figure 2.2 (b)) show a simple bar chart showing the time the user has spend on each course. I chose this way of visualising the data, as I think it would be an easy way to compare how much time the user spends on each of their course. Both the time-tracker and the statistics were requirements (see #19 in table A.1) and #28 in table A.2, respectively).

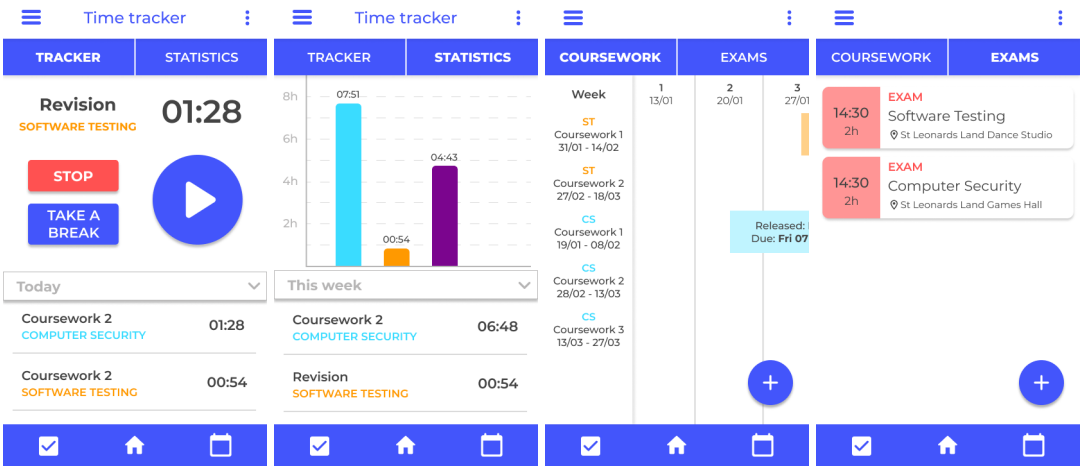


Figure 2.2: (a) Time tracker screen (b) Statistics screen (c) Coursework screen (d) Exams screen

The coursework screen (see figure 2.2 (c)) shows a Gantt Chart [52] that displays the release date and deadline of an assignment. I decided to add a coursework manager as coursework is a big part of an Informatics student's life and I think they should be able to see their deadlines in an easy way. Furthermore, having a Gantt Chart to display deadlines was one of the requirements (see #4 in table A.1). Through tabs at the top, the user can also access the exams screen (see figure 2.2 (d)). This was not part of the requirements, but as with coursework, I thought it would be a good idea to have the exams clearly displayed to be able to be seen at a glance.

The screen to edit an event (see figure 2.3 (a)) gives the user the option to add a title, set a category for that event, a date and time and allows the user to mark if the event is an All-day event. It also lets the user choose if the event should repeat, if they should get notified before the event, if they want to add notes or if they want to invite people to the event. These are all features which other calendar apps include and I thought this app should have too. Furthermore, having notifications for events and recurring events were also requirements (see #5 and #7 in table A.1, respectively). Additionally, the user can also add tasks to their event, as one of the requirements with high priority was having tasks linked to calendar events (see #17 in table A.1).

The screen to edit a task (see figure 2.3 (b)) gives the user the option to add a title, subtasks, a category, and to add the task to today. The latter option is not common in task manager apps, but I thought it would be a good idea as this would allow the user to mark the tasks they wanted to complete today and to have them clearly displayed in the home screen. Furthermore, the user can add a reminder, add a deadline, add the task to the calendar, set a priority, set the task to repeat and add notes. Most of these options can be seen in other task manager apps, and both adding a deadline and prioritising the tasks were requirements (see #11 and #13 in table A.1, respectively). The add a task to the calendar option was something I thought could be useful to some users, so that they can say when exactly they want to complete this task and they can see it on their calendars, which might help them complete it when they said they were going to.

The screen to edit coursework (see figure 2.3 (c)) gives the user the option to set a title, category, release date and due date and time. All of this data would be used to populate the Gantt chart mentioned before. Furthermore, to cover requirement #21 in table A.1, help distribute work, and #25 in table A.2, monitor how much time the user spends on something and notify behind schedule, the user can select how much time they want to spend on this coursework, they can select if they want to get notifications to keep on track and they can ask the app to suggest times to work on the coursework and these will be added to the user's calendar as events.

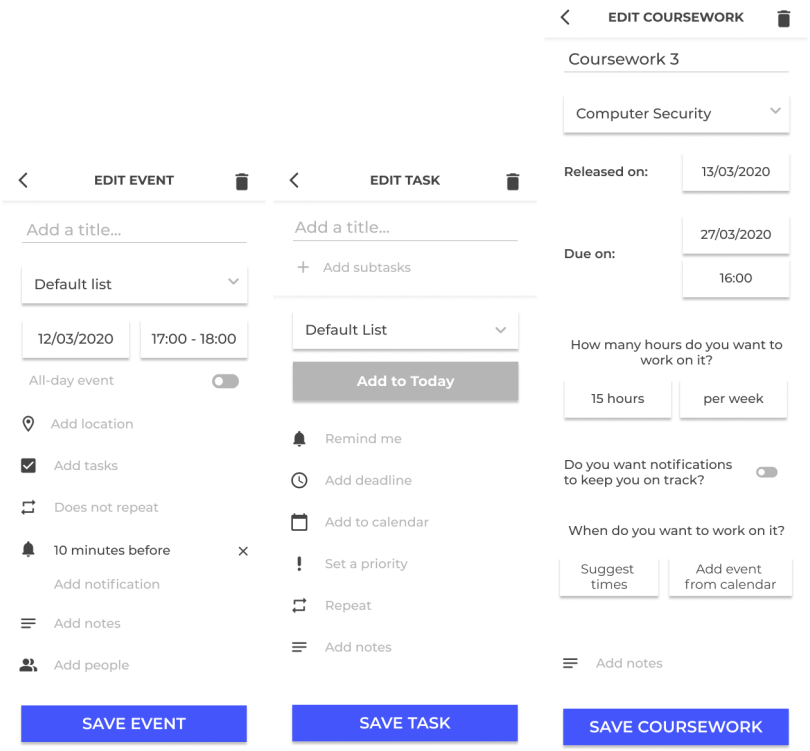


Figure 2.3: (a) Edit event (b) Edit task (c) Edit coursework

Lastly, to cover requirement #6 in table A.1, I used different colours to colour-code every element in the app by its category: the events, the tasks, the time-tracker sessions and the coursework.

2.2.2 Final evaluation

During this phase we had many positive comments about the app, regarding the interaction, the ease to use, the colours, etc., but also some of the functionalities, like the coursework manager and the time-tracker. We also gathered new requirements, the main ones being: having confirmation messages after every action (see #30 in table A.2), having a study timer (see #34 in table A.2) and have the app send notifications about the tasks left to complete today (see #37 in table A.2). We also gathered new suggestions, such as making the font smaller (see #5 in table B.1) and making the button in the time-tracker smaller (see figure 2.2 (a)) (see #12 in table B.1). There were, however, some discrepancies in the opinion of the participants about the design of the

time-tracking functionality and on how the design could be improved. Some people wanted to remove things, such as the take a break button and the history panel at the bottom half, that other people liked, which left me unsure on how we could proceed with its design.

Lastly, the students were asked to rate the app from 1 (very unhelpful) to 5 (very helpful) in terms of how they thought the app could potentially help them with several aspects of university life. The results were overall positive, with most these aspects having a mean above 4 (somewhat helpful), except the last aspect, managing personal life, which had a median of 3.88. This might be due to students not being used to using this kind of apps to manage their personal life. Students were also asked to complete the System Usability Scale (SUS) [49]. The results of the SUS had a mean of 75.59 and a median of 80. This puts this design in the acceptable range according to the acceptability scale [48] and gives it a grade of C in the grading scale [47].

2.2.3 Moving forward

In the last evaluation, I received interesting and useful feedback regarding the design of the app, however, there were mixed opinions about the time-tracking part of the design, as mentioned previously. As part of the background research, my aim was to start looking into improving the design of this functionality, and the idea of combining it with a study timer functionality, which was not only mentioned during the last evaluation, but it is also a practice recommended by the University of Edinburgh Institute for Academic Development (IAD) [18] in their workshops and in their Study Hub [41]. Early on we also decided to use Participatory Design methods [57] to design this functionalities, as we thought it would be useful to go back to the user and have them be more involved in their design. To do this, I also aimed to review Participatory Design methods [57] (see section 2.4 for some of these methods).

The main goal of this part of the project is developing the app, which entails finding the best tools and approaches for its implementation, finding a way to store the data and to authenticate the users. Furthermore, I also wanted to look into adding more of the requirements and suggestions that were unaddressed last year.

2.3 Timer and time-tracking methods

In this section, I review the literature on the timer and time-tracking methods since, as I mentioned in section 2.2.3, my aim this year was to design a timer and time-tracking functionality that suits the target users.

2.3.1 The timer method or Pomodoro technique [50]

The Pomodoro technique is a time management method developed by Francesco Cirillo in the 1992 [50]. This technique consists of setting a timer for a period of 20 or 25 minutes and working on a task for that amount of time avoiding distractions as much as possible. This is called a ”pomodoro”, making reference to a kitchen timer in the shape of a tomato (pomodoro in Italian). Once that time is over, the individual can take

a break for 5 minutes. This process is repeated 3 or 4 times, then they can take a longer break of 15 or 30 minutes.

This technique can be used for any situation, but is often recommended for studying. Both the IAD and the Student Counselling services at the University of Edinburgh [40] recommend making use of this technique when studying or working. This technique has been proven to improve productivity [67] and to help reduce procrastination and distractions by helping the user draw a better distinction between study time and free time [54].

2.3.2 Time-tracking

The time-tracking practice consists in recording what activities occupy one's day and how much time each activity takes to complete [65]. This serves as both a time assessment technique, which allows the individual to increase self-awareness regarding how they use their time, as well as a monitoring behaviour, which allows them to observe how they use their time when performing certain activities (see section 2.1). This technique helps boost productivity [65] as it allows the individual to reflect on and understand how they spend their time. This increased self-awareness can also help the individual to plan their time in a more accurate way, without under-estimating or over-estimating how much time they will spend on a certain activity.

This practice is suggested by the IAD [18] in their workshops as well as their Study Hub [41]. They claim that this exercise in self-tracking can also help to identify any elements that might cause disruptions in one's productivity.

2.4 Participatory Design and its methods

In this section, I am reviewing Participatory Design and some of its methods. The reason for this is that in order to design a timer and time-tracking functionality that suited Informatics students, I wanted to involve them in the process of designing such a functionality.

Participatory Design is a methodology that involves all stakeholders, and especially the user, in the early design stages of a product to gain a deeper understanding of what they want and expect from the product [57]. Participatory Design also aims to have the different stakeholders work together on the analysis and design of a product in order to have them understand each others views [58].

2.4.1 Card sorting

Card sorting is a research method used to draw out underlying mental models and to better understand how users model certain information [64]. This method consists of giving the users a number of cards with terms written in each one and having the user sort the cards into groups and labelling these groups.

2.4.2 Storyboarding

Storyboarding consists of having the users draw on a piece of paper how they think the user would interact with the system to solve a specific problem or to complete a certain task in a number of steps [56][39]. This method can help surface ideas or issues that the user could run into when using the product as well as parts of the experience that lie outside the interaction of the user with the product.

2.4.3 Mockups

Mockups or paper prototypes are simple models created by/with the users to show some functionality[53][66]. They are an easy and inexpensive way to express how the functionality of a system will be presented in a way that users can understand. Mockups also allow users to test and make changes to the design without any prior technical knowledge.

2.5 App development

In this section, I review the different types of app one can develop as well as the tools I considered for the development.

When developing a mobile app, one has to know all the different types of apps that can be developed since, even if the end product will look the same, each type of app offers different development paths and characteristics [16]:

- Native app: This is the most common type of app. A native app is built for specific platforms and each only accepts a specific programming language. This means that when developing an app, you will only be able to develop for a specific platform. This type of apps can be downloaded into a mobile device. Some of the tools used to develop these apps are Android Studio [3] and Java [20] or Kotlin [22] (for Android apps) and XCode [43] and Swift[42] (for iOS apps).
- Web app: A web app works like a website, in that the user can only access it through a web browser and does not need to be downloaded. However, this kind of apps cannot always deliver on standards that users are used to with native apps, and users might get frustrated with performance and usability issues. Some of the tools used for these apps are Python [30] or Ruby [38] for the back-end and HTML [17], CSS [7] and JavaScript[21] for front-end.
- Hybrid app: A hybrid app is a combination of native and web apps, which means that, like a native app, it can be downloaded to the device, but it is essentially a web app, which means it can be used on different platforms. Some of the tools used to develop these apps are React Native [32] and Flutter [12].

For the purpose of developing this app, I considered two options: using React Native or developing an Android [2] app using Android Studio [3]. The reason for considering these options was that, in the case of building and Android app, I was already somewhat familiar with Android Studio, and in the case of React Native, I was familiar with JavaScript.

2.5.1 Android [2] app using Android Studio [3]

Android Studio [3] is the official IDE (Integrated Development Environment) for the development of Android apps. It offers many tools and features that help the developer when building Android apps, such as an Android emulator, version control, testing and debugging tools, etc.

The advantages of using Android Studio are the extensive set of tools available and, due to its popularity, and extensive community of developers and many sources of help when developing. The main disadvantage is that using Android Studio, you are only developing an Android app. To have the app available for iOS, you would have to develop a completely separate app.

2.5.2 React Native [32]

React Native is a framework to build hybrid apps that uses JavaScript [21], a programming language which is an essential part of interactive websites and web apps, together with React [31], an open-source JavaScript library for building user interface components [32].

A 2016 study explored the differences between apps built using React Native and native apps [46], and found that differences where minuscule in terms of user experience (UX), and that, because React Native uses JavaScript, the threshold for developers from knowing JavaScript to start developing in React Native is low compared to other systems. This translates into shorter timeline to an end-product that is available in all platforms compared to developing for Android and iOS separately, since when writing native apps using other frameworks, you will need to implement separate apps for the different platforms.

2.6 Database systems

In this section I am reviewing the different database systems I considered for this app, as I would need to store information from the user, such as their events and tasks. The reason for considering these options is that they offered an easy way for data synchronization between local database and back-end server. Furthermore, I was also familiar with Cloud Firestore [5], as I had used it before. Both services are paid, but have free options for small apps or apps that are just starting out.

2.6.1 Cloud Firestore [5]

Cloud Firestore [5] is a flexible and scalable NoSQL cloud database, which stores and syncs data across client apps through realtime listeners, and offers offline support.

Cloud Firestore is one of the products of Firebase [11], which is a set of products by Google aim to help develop mobile and web applications. One of the other solutions included in Firebase [11] is Authentication.

2.6.2 MongoDB Realm [26]

MongoDB Realm [26] is a serverless platform and mobile database from MongoDB [25]. It allows the sync of data between the client and the server, combining together the two elements into an application layer for the app.

2.7 Other time management apps for students

In this section I review some of the existing time management mobile apps that are specific for students.

- MyStudyLife [28] (free): this cross-platform app offers scheduling, a task manager, an assignments manager and reminders. It also offers different types of calendar entries for classes and exams and it allows the user assign classes to an academic year or term, so that the app knows until when it should repeat the class.
- myHomework Student Planner [27] (free, offers paid premium account): this is also a cross-platform app which lets the user track assignments and tests, track classes, receive reminders for deadlines, and includes a widget for upcoming homework. By paying for the premium account, it also offers enhanced app widgets, shared planner, file attachments, etc.
- iStudiez Pro [19] (paid): this is also a cross-platform app which offers a schedule, a planner, task/assignment manager, reminders and integration with other calendar apps.

All of these apps offer some type of combination or calendar/schedule, task manager and assignments manager, and they are all cross-platform which will probably be very helpful for many users. However, the issue with some of these student-specific apps is that they are made to help the user with their studies-related time management, but none of them seem to integrate things unrelated to the user's studies, such as social events or personal tasks. Furthermore, none of them include a timer or a time-tracking functionality.

Lastly, I would like to review Forest [13], a timer and stopwatch app that I came up during my research on the timer method for time management, as well as in the design workshops described later in this report (see section 4.7). This app only consists of a timer and a stopwatch, and it is not specifically made for students, but it includes gamification, which might be one of the reasons why it is such a popular app (#1 ranked productivity app in 136 countries in the App Store [14]). The app lets the user grow a tree while the timer is activated. When the time is up, the user has a new tree they can see displayed in their virtual forest and they win coins, which they can use to buy other species of trees or bushes. If the user leaves the app while the tree is growing, the tree will die, which motivates the user to not check other apps on their phone. When using the stopwatch, the user will grow a tree in the first 10 minutes of the stopwatch, and after two hours, the stopwatch will stop and encourage the user to take a break. Furthermore, with the premium account, the user can use the coins they won towards planting real trees.

Chapter 3

Methodology

This chapter presents the methodology used throughout the project, including how I approached every phase and the methods and tools I used to carry out my plans.

3.1 Literature review

During this phase my aim was to answer Research Question 2: What are the technologies we can use to implement this app?

At the start of this project, I started by looking into the different tools and approaches I could use to implement this app (see Background section 2.5). Since the design for the time-tracking and study timer functionalities were not yet clear, I also decided to use Participatory Design, so I reviewed its methods (see Background section 2.4) and I reviewed the methods of time-tracking and timer (see Background section 2.3). Lastly, I looked into options for a database to use in the app as I needed a way to store and sync data in the app (see Background section 2.6).

3.2 Design of the timer and time-tracking functionality

During this phase my aim was to answer Research Question 1: How can we design a time-tracking and timer functionality for the app that suits Informatics students at the University of Edinburgh?

To do this I conducted two design workshop with six participants in total, all Informatics students at the University of Edinburgh. During the workshops I instructed the participants to draw mock-ups (see section 2.4.3) of what they thought these functionalities should look like. Once they were done we discussed the results together. I decided to use mock-ups since I thought it would be a very useful tool to see exactly what the participants expected these functionalities to look like and to do.

From this stage I obtained a better idea of what potential users might want to see in such a functionality and these results are presented in section 4.7.

3.3 App development

During this phase I also aimed to answer Research Question 2: What are the technologies we can use to implement this app?

For the development of the app, I chose to use React Native [32] (see Background section 2.5.2) because it provides a faster way to implement an app for both Android and iOS, and I was already familiar with JavaScript.

For the database and authentication, I used Firebase [11], as it provided me with an easy way to store my data as well as authenticate user in one solution. Cloud Firestore [5], which is the database from Firebase, also provides data synchronization between local storage and the backed server, which I thought was crucial for this app.

3.4 Evaluation of the app

During this part of the project my aim was to answer Research Questions 3, 4 and 5: How the usability of the app is perceived among students of Informatics as well as students from other schools, and what potential impact this app could have.

I conducted four group sessions and seven individual sessions with 22 students in total. For the group session, I was not able to use the app itself since I could not run multiple emulators at the same time, so I created a duplicate of the app on Figma [10] and used that to evaluate the app. During these group sessions, I conducted a Stakeholder Walkthrough [62], where I asked each participant to complete three tasks and after each task I would ask them to describe what steps they took to complete the tasks. I used this method as it allowed me to know the different paths the participants had taken to get to the same goal and to have the participants discuss their opinions together. This was followed by some questions and the SUS (System Usability Scale) [49].

For the individual sessions, I conducted Question Asking Protocol [59], where I asked the user to complete the three tasks on the app and while they were completing them, I would ask questions about the app itself. The reason to conduct a Question Asking Protocol was that, since it allowed me to ask questions while the user was using the app, I could get a better idea of what the participant was thinking and what they thought about specific elements of the design. This was also followed by a few questions and the SUS [49].

The reason I decided to conduct individual sessions as well as group sessions was to not mix students of different years, as group sessions were only formed by students of the same year. Furthermore, I thought having some individual sessions would provide me with more detailed feedback. However, due to time constraints I could not conduct as many individual sessions.

From these studies I obtained feedback about the app and ways on how it could be improved. The full results are presented in section 6.7.

Chapter 4

Designing the timer and time-tracking functionality

In this chapter I present the process of designing the timer and time-tracking functionality for the app, with the help of design workshops conducted with Informatics students. These studies were approved by the University of Edinburgh School of Informatics, with RT 2019/94240. The methodology used during this phase is fully described in section 3.2.

4.1 Aims

The goal during this process was to get a better idea of what Informatics students would envision the timer and time-tracking functionalities to look like and what kind of features or elements they would like to be included. I aimed to cover Research Question 1: How can we design a time-tracking and timer functionality that suits Informatics students at the University of Edinburgh?

4.2 Participants

There were six participants in total (see appendix C to see a full list of participants). They were all Informatics students at the University of Edinburgh: four 4^{th}-year students and two 5^{th}-year students. I decided to recruit students in their last year of studies since these students will have had more experience with time management in the context of university, especially with timer and time-tracking apps, so they might have a better idea about what they would like to see in functionalities like these. Half of these students had participated in studies conducted last year: s4, s5 and s6.

4.3 Data Collection Methods

I conducted two participatory design workshops [57] with three students in each session. I decided to use this method since in the final evaluation I conducted in the first

part of this project, I received some mixed opinions on this part of the design (see section **??**) and I wanted to know how students would envision these functionalities themselves, without being biased by my original design.

All studies were conducted over a university account of Zoom [45], and were audio and video recorded with the consent of all participants. I used this recording, together with note taking, to collect data.

I decided to use Zoom [45] because it provides with automatic captioning of the audio which is very helpful when having to transcribe and analyse the data.

4.4 Materials

I composed a Participant Information Sheet (see Appendix D). This document gives the potential participants the necessary understanding of the motivation and procedures of the study and sources of information to answer any further questions. This includes: information about the researchers and the study, the impact of participation, such as risks and benefits, and information about data protection.

I composed a consent form (see Appendix E), which was used to obtain the consent from the participant. In this form, the participant was asked to confirm to have had access to the participant information sheet, to have understood the purpose of the study, and that their participation was voluntary. They were also asked to approve (or not) of being audio and video recorded and have their anonymised data be used in academic publications and in future ethically approved research. Finally, they were asked to confirm agreement to take part in the study.

Finally, I prepared a script for the workshop that explained the intended purpose of the app, as well as the desired functionalities that the participants had to think about (see appendix F). After the mock-ups were presented, I asked them what elements they liked and disliked from each other's designs to get their opinion on the different elements and to start a discussion.

4.5 Procedure

When the potential participants were first contacted, they were sent the Participant Information Sheet (see Appendix D), so they could review all the relevant information before they agreed to participate in the study.

If they agreed to participate, I would ask for their availability, and try to schedule a date and time that would work for some of the other participants. I would also send them the consent form (see Appendix E) and would ask them to sign it if they consented to everything stated on the form. Then I would schedule a meeting on the Zoom [45] platform and send them the information of the meeting as well as a link to join on the day.

On the day of the workshop, I would make sure all participants in the workshop had signed the consent form and then I would inform them that I would start recording.

Once I started recording, I would read the script and answer any questions from the participants. Then I would give the participants 15 minutes to draw a mock-up (see section 2.4.3) on paper or on a device of what they thought the aforementioned functionalities would be presented on the app. Once everyone was finished, I would ask the participants to send their mock-ups, and then I would ask each participant to explain what they included in their mock-up and why. After each participant was finished explaining, I would ask the other participant if they had any questions or comments. Lastly, I would ask them to discuss the different features and elements that they found interesting, liked or disliked from the other participants' mock-ups as well as their own. Once the discussion was finished, I would stop recording and ask if they had any other questions or comments. Lastly, I would thank them for their participation.

4.6 Analysis

After the sessions, I transcribed every recording using the Zoom captions from the sessions and analysed these transcriptions together with the notes taken during the sessions as well as the mock-up pictures, with NVivo 11 [29] using thematic analysis [55] and a combination of top-down and bottom-up coding. When analysing, I started by creating themes for each of the functionalities (timer, time-tracker, statistics), then I created sub-themes for every element mentioned by each participant when describing their own mock-up, and as I noticed common data, I created new themes and sub-themes. The themes and subthemes resulting from the analysis correspond to the sections and subsections described in the results.

4.7 Results

In this section, I will be presenting the different themes found when analysing the data from the workshops.

4.7.1 General

- Gamification (supported by 3/6 students, none had participated in a previous study): one of the students mentioned the idea of gamification, inspired by other timer apps like Forest [13]. They said that the gamification side of the app would encourage the user to stay on track and stay motivated. The two other students in the session agreed.
- Hiding the time passed or time left (supported by 3/6 students): One student said they would like the time to not be shown, as having the time on the screen can be stressful or distracting. The two other students in the session liked the idea.

  "I think when you show the exact time that is left or that has passed that can be stressful and could also be distracting to the user, because something's always moving and changing on the screen." - s2

- Assigning a label to the session (supported by 3/6 students, none had participated in a previous study): all three students in one of the workshop sessions

mentioned how labeling a timer or tracker session, with the topic being studied, for example, would be helpful to the user when seeing past sessions.

- Edit session details at any time (supported by 2/6 students, none had participated in a previous study): Two students mentioned how the user should be able to still edit the details for the session (label, start and end time, etc.) even after the session has finished, in case they ended up working on something different or if they forgot to stop the session when they stopped working.
- Sync timer with tracker (supported by 2/6 students, none had participated in a previous study): One student thought of having the timer synced with the time tracker, so the data from these sessions will later just be shown as the user spending time on one activity or the other, instead of making the distinction of the session being a timer or a tracker session. Another student agreed.
- Remove distracting elements from the screen during a session (supported by 1/6 students, who had not participated in previous studies): One student included the option of removing any unnecessary elements during a timer or tracker in their mock-up, in order to avoid as many distractions as possible.

  "I would remove the menu bar at the top and bottom, in case you want the screen to be distraction-free." - s1

4.7.2 Timer

- Show bar around timer representing the time left in the timer (supported by 3/6 students, all of which had participated in previous studies): One student had the idea to have a circular bar around the timer showing how much time is left in the timer, as well as having different colours for working and resting. The two other students in the session liked this idea.

  "And then the bar around shows how much time is left. The bar could be a different colour depending on if you're working or resting, so to kind of visualize what you're supposed to be doing." - s4

- Custom timer periods (supported by 2/6 students): Two students in different workshop sessions thought of allowing the user to set a time for the timer, instead of always following the Pomodoro technique of 25 minutes of work and 5 minutes of break.

  "And you can either click Pomodoro, which is the 25 minutes of work with 5-minute break. It is the preset one. But if you want to set your own custom time, you can click the custom time and you can do it. For example, 40 minute work and 15 minutes rest." - s2

- Choose between showing time left or time passed (supported by 1/6 students, who had participated in a previous study): One student had the idea to give the user the option between showing the time left on the timer or the time that has already passed.

”And then the timer will have either how much time has passed, or how much time is left, and people can choose between the two. I personally would prefer probably how much time is left.” - s4

- Loop through timer sessions (supported by 1/6 students, who had participated in a previous study): One student thought that the user should be able to pre-select how many sessions they would like to go through, and the app would tell them when the time is to take a break or to keep working.

  ”So if you set the work time and the rest time, then you can just keep looping through it until you tell it to stop. Or maybe you could add how many repetitions you want as well.” - s4

- ”Take a break” button (supported by 1/6 students, who had not participated in previous studies): One student thought of including a ”Take a break” button to have the user be able to choose when they would like to take a break.

4.7.3 Tracker

Split time in laps (supported by 1/6 students, who had not participated in previous studies): One student thought it would save time to just have the user be able to press just one button to signal to the app that they are now working on something else, and the user would later be able to tell the app what they were doing during that time.

”So without going through the whole process of, you know, stop the session, create a new session choose, of course, you can just say, okay, new lap. [...] Laps are more or less about helping you remember at which point you started working on something else.” - s1

4.7.4 Statistics

- Split information by course (supported by 2/6 students, who had participated in previous studies): Two students thought of splitting the information by course.

  ”I think having things grouped by course more useful for me because that’s how I tend to think about it.” - s5

- Break down the data from each course (supported by 2/6 students, who had participated in previous studies): Both of them also thought that then the information for each course could be broken down into the different activities.

  And then if you select that course for example, it shows you how it’s broken down by coursework, tutorial, etc. - s6

4.8 Conclusion

In this section, I presented my studies to design a timer and time-tracking functionality for the app and gathered requirements for said functionality (see table 4.1 for all requirements). From the requirements gathered, I decided I would try to implement

the following to start with: assigning a label to the sessions (#3), as this would be useful when displaying past sessions, edit session details at any time (#4), as this allow the user more flexibility, show bar around timer (#7), as this provides a visual way of showing the time that is left, custom timer periods (#8), as this allows the user more flexibility and lets them try whatever timer periods works better for them, loop through timer sessions (#10), which makes the app need less input from the user, so the user can focus on their work, and split information by course (#13), so the user can see how much time they are spending on each course, and adjust how they spend their time accordingly. The rest of the requirements will be considered again in the future, however, I have decided to not implement them yet due to either time constraint reasons or because it might require more research to ascertain the usefulness of these ideas.

#	Requirement	Section	Supported by
1	Gamification	General	Supported by 3/6 students.
2	Hiding the time passed or time left	General	Supported by 3/6 students.
3	Assigning a label to the session	General	Supported by 3/6 students.
4	Edit session details at any time	General	Supported by 2/6 students.
5	Sync timer with tracker	General	Supported by 2/6 students.
6	Remove distracting elements from the screen during a session	General	Supported by 1/6 students.
7	Show bar around timer representing the time left in the timer	Timer	Supported by 3/6 students.
8	Custom timer periods	Timer	Supported by 3/6 students.
9	Choose between showing time left or time passed	Timer	Supported by 1/6 students.
10	Loop through timer sessions	Timer	Supported by 1/6 students.
11	”Take a break” button	Timer	Supported by 1/6 students.
12	Split time in laps	Time-tracker	Supported by 1/6 students.
13	Split information by course	Statistics	Supported by 2/6 students.
14	Break down the data from each course	Statistics	Supported by 2/6 students.

Table 4.1: List of requirements for timer and time-tracker

Chapter 5

Development of the app

In this chapter I will explain the inner workings of React Native [32], how I used it to implement the design developed and how I decided to structure the database to serve my purposes.

5.1 How React Native[32] works

As mentioned in the Background chapter (see 2.5.2), React Native is a framework to build apps that uses JavaScript [21] to access the platform's APIs and React, more specifically React components, to describe how the app looks and interacts [6].

To display information or images in an app, albeit Android or iOS, we use *views*. *Views* are basic building blocks when implementing an app, which can be used to display any kind of element, such as text, button, image, title, etc. With React Native, we invoke views using *components*. Components are similar to views in that we use them to describe how the elements look like and interact but for React and React Native, which React Native then translates into the corresponding Android or iOS view depending on the operating system of the device [6].

5.2 Database structure

Currently, the database structure is comprised of six tables:

- Users: this is the most important table. This table stores the user ID, which is used in other tables as a foreign key, to keep track of what events and tasks belong to which user, and the email of the user.
- Categories (foreign key: user ID): each user has a different set of categories, as each user has different courses they attend. This table stores the category ID, as well as the name and colour of the category.
- Events (foreign keys: user ID, category ID, task ID (optional)): this table stores the events for each user.

- Tasks (foreign keys: user ID, category ID): this table stores the tasks for each user.
- Event tasks (foreign keys: event ID, task ID): the user would be able to add tasks to an event, so this table stores the task ID and the event ID it corresponds to.
- Timer/Stopwatch (foreign keys: user ID, category ID): this table stores the data for the timer and stopwatch sessions which the user has chosen to save.

To see the Entity-Relationship diagram for this database, see appendix G. In the future, there would probably be additional tables as more functionalities are implemented, such as a table to store coursework information.

5.3 Implementation

In the following section I will be describing what I implemented and the biggest issues I ran into during this process.

5.3.1 Home screen

As in my original design, I partitioned the home screen in two section: today's events and today's tasks (see figure 5.1 (a)). I decided to do this as I thought it would be useful for the user to be able to see the things they had to do today in terms of events and tasks at a glance. In today's events I retrieve all events for the logged in user with a start date of between today and tomorrow. To display events I created a component called `EventCard`, which takes each event retrieved and displays its information (see figure 5.1 (c)). When the card is clicked, a new screen appears showing all the details of that event, with an edit button to edit the details of the event. The card displays, on the left, the time the event is starting and its duration, and on the right, the category, title, and one other detail of the event, depending on what the user has added to the event. It could either be the location or the notification (see figure 5.1 (a)).

I run into many issues during this part. First, retrieving the events as this was the first time I was attempting to retrieve data from the database using React Native, so for I long time I could not manage to retrieve the data that I needed. Second, I was trying to add an edit button to the header of the event screen, which proved to be much harder than expected, as the header is separate from the rest of the screen. Because of this, I could not pass the information of the event to the edit screen. To solve this I am only passing the ID of the event to the edit screen and retrieving the data of the event from the database using the event ID, which also took a long time to figure out.

Below today's events, we have today's tasks. This section displays all tasks which have been marked to complete today when creating or editing the task. For this I created a component called `HomeTaskCard`, which again displays the information of the tasks, together with a Checkbox from the *React Native Elements* library [4], so that the user can check the task off directly from the homepage. When the user has set a deadline, it will show how many days are left until that deadline. If there are less than 10 days left or the deadline has passed, it will show the date of the deadline, and in the latter case,

it will show it in red (see figure 5.1 (a)). If there is no deadline, but there is a reminder, the card will show the date and time of the reminder. I decided to add this detail to provide the user with a bit more information, and especially so that the user could see the deadline clearly for each task.

When the card is clicked, the user is able to edit the details of the task. However, for the editing of the task, I ran into some issues that I have not been able to solve. When the edit screen is opened to edit an existing task, the information is retrieved from the database and displayed in each section of the screen. However, the dates for the deadline and reminders are not read properly and shows *Invalid date*. I implemented it the same way it is in the screen to edit an event, but it did not work. I tried different ways of implementing it, following as well recommendations from online development forums, such as changing the format of how the date is saved first or formatting the date in different ways after it is retrieved from the database, but I still have not been able to solve this issue. Nevertheless, the user is still able to edit the details of the task and these will be saved.

Lastly, there is an add button at the bottom right corner. When clicked, a modal opens which lets the user either add a new event or add a new task (see figure 5.1 (b)). All of the features in the home screen follow the design described in section 2.2.1.

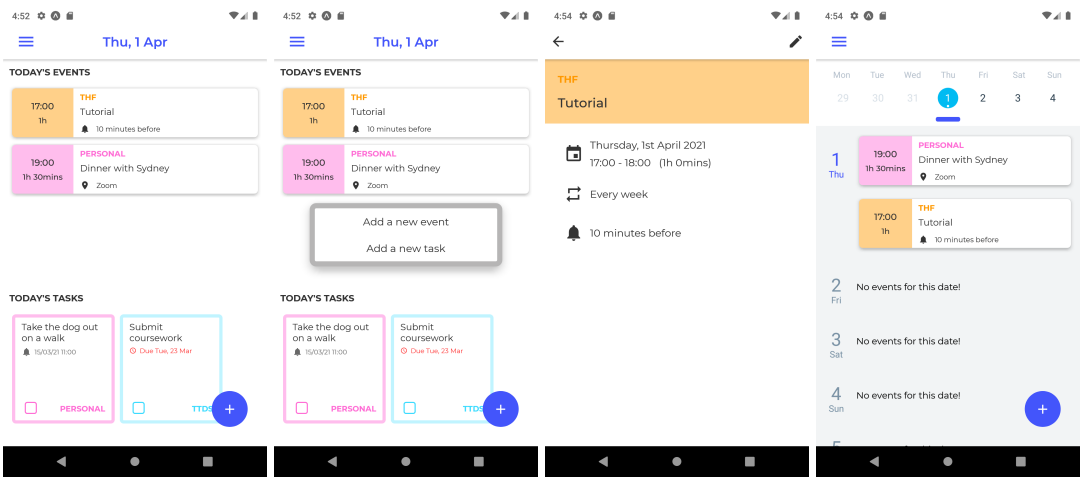


Figure 5.1: (a) Home screen (b) Modal to add event/task (c) Event screen (d) Calendar screen

5.3.2 Events and calendar

For the calendar I used the `Agenda` component from the *React Native Calendars* package [1] together with the aforementioned `EventCard` components to display all events from the logged in user (see figure 5.1 (d)). Using this package also took me longer than expected to figure out, as at first the events would not show up.

When editing or adding an event, the user can input several types of details (see figure 5.2 (a)). They can add a title, location and description, which was implemented using a `Formik` component [15], as well as a category, notification, repetition of the event, implemented using `Modal` components [24] (see figure 5.2 (b) and (c) for the category

and reminder modals), and a start and end time and date, which were implemented using `DateTimePicker` components from *React Native Datetimepicker* [8]. These options were included in the design described in section 2.2.1.

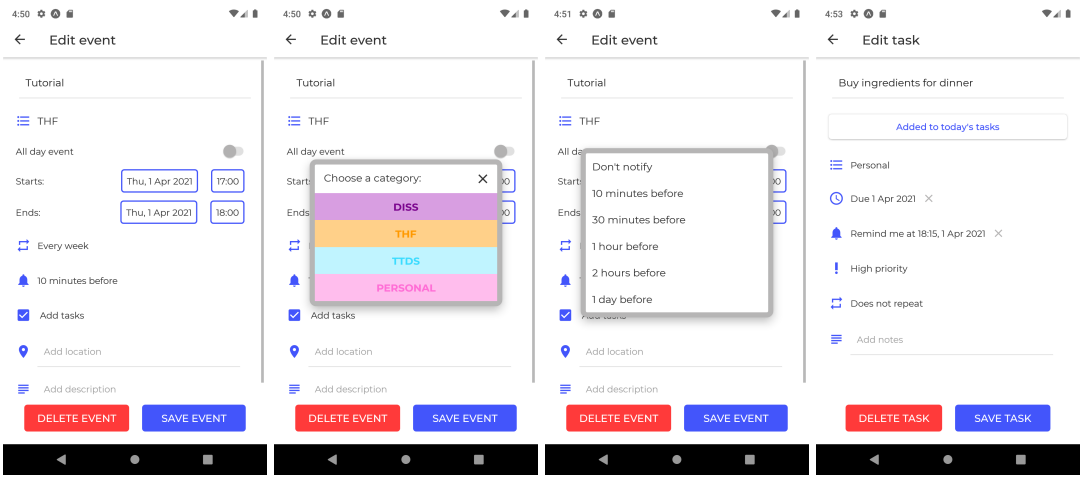


Figure 5.2: (a) Edit event screen (b) Modal to choose category (c) Modal to choose reminder (d) Edit task screen

5.3.3 Tasks

Similarly to editing an event, the screen to edit or add a task (see figure 5.2 (d)) was implemented using a combination of `Formik` [15] (for the title and description), `Modal` [24] (for the category, priority and repetition) and `DateTimePicker` [8] components (for the deadline and reminder). I also added a button below the title for the user to add the tasks to today's tasks, that way these tasks will show up in the home screen. The reason I added this option was because I thought this would be an easy way for the user to say which tasks they wanted to complete today. All of these options were included in the design described in section 2.2.1.

5.3.4 Timer and Stopwatch

The first screen (see figure 5.3 (a)) shows three buttons: Start a timer, Start a stopwatch, Resume previous session. When the Start a timer button is pressed, the user will be able to set the working time, break time, number of intervals, label and category for that timer (see figure 5.3 (b)), following some of the requirements to be able to add a label to a timer, set custom timer periods and loop through timer sessions (see requirement #3, #8 and #10 in table 4.1, respectively) . I used the *React Native Numeric Input* package [34] for the numeric inputs of the work time, break time and number of intervals, as seen in figure 5.3 (b).

I ran into an issue when using the components from the *React Native Numeric Input* package [34]. When the value of one of them is changed by pressing the buttons on the sides, the value of the others reset to their default value. This gives the illusion that the values of the other inputs are always going back to default and the user is not able to set the values they want. However, in the back-end the values are not actually being

changed when the screen shows them resetting, so when the user hits *Start*, the timer is set correctly with the desired values.

For the timer as well as the stopwatch I used the `Timer` and `Stopwatch` components from the *React Native Stopwatch Timer* package [37] (see figure 5.3 (c) for the timer). At the bottom of the screen for both the timer and the stopwatch, the user can find two buttons, one to pause and another one to abort. When the user presses the abort button a modal appears with three options (see figure 5.3 (d)): to save the session, which is yet to be implemented, to discard the session or to continue working. The reason I added this was to give the user the option to not save their sessions as they might have started the session by accident, or might have been interrupted.

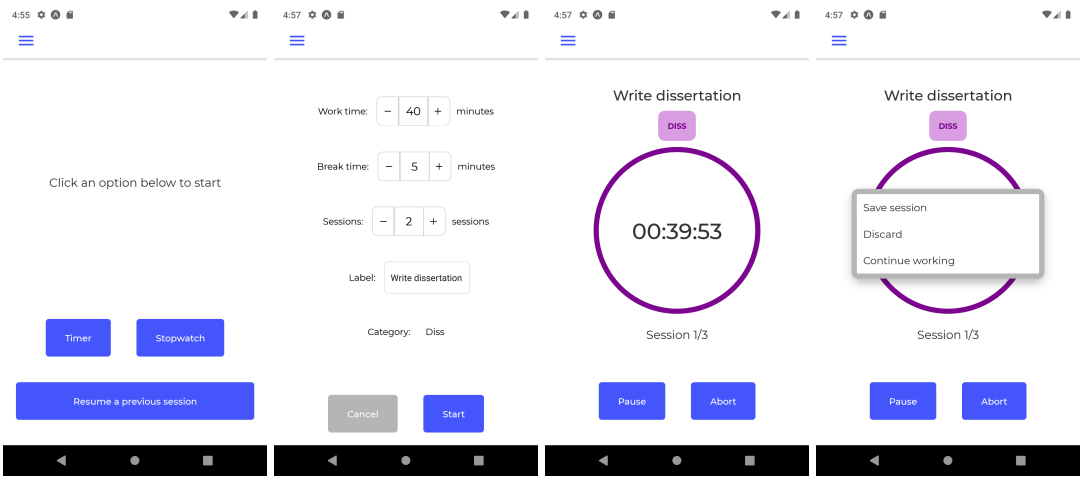


Figure 5.3: (a) Timer/Stopwatch screen (b) Setting the timer (c) Timer in work mode (d) Modal to abort session

5.3.5 Login and Register

For these screens I used `Formik` [15] for the form and Firebase [11] Authentication for the authentication. The login screen (see figure 5.4 (a)) has a small form for the user to input their email and password, a login button and a button to register. The Register button will take the user to a new screen (see figure 5.4 (b)) where they can input their email, their password, and confirm their password. I used `Yup` [44] for input validation and to make sure both passwords matched, which took me some time to figure out.

5.3.6 Other issues

After one month of developing, the app started having trouble loading. Every time it would load it would freeze right before the loading was complete (see figure 5.4 (c)). Every time this would happen I would have to close the app in the emulator and load it again. When I wanted to open the app, I would have to do this several times, until the app actually loaded and I could work with it and use it. This would make troubleshooting or adding new elements very hard, as if there was anything causing an error, I would have to go through this process of restarting the app several times

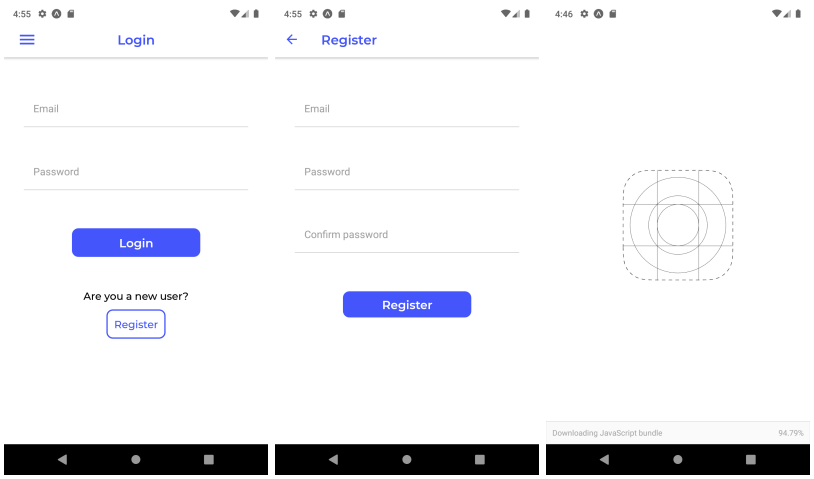


Figure 5.4: (a) Login screen (b) Register screen (c) App stuck when loading

constantly. I spent several days trying to solve it when it first started happening but I did not manage to solve it.

Furthermore, I got the same error message several times during the development of the app: `Cannot perform a React State update on an unmounted component. This is a no-op, but it indicates a memory leak in your application.` This was a very big issue since the error message does not say where the error is, and every time I would get this error it would take me several days to solve it, as every time the solution would never be the same as the previous time I got this error. The first couple of times I got this issue, I solved it by surrounding the code to save the event and the task in the database with an `useEffect` hook, after trying many different things. The next couple of times I had to rethink how to pass data between components. The app actually still returns this error when discarding a timer session, I still have not been able to solve this issue with previous solutions or any of the other things I have tried.

5.3.7 Elements left for future work

5.3.7.1 Calendar and events

The user can save an event, however, they cannot delete an event yet. Additionally, I planned to implement different ways of viewing events, such as calendar view, week view and one day view, as this was one of the requirements.

5.3.7.2 Tasks

My aim was to have a screen that displays all tasks for that user, but the user is also able to choose from the different lists of tasks that are available, which filter task by category or by priority. However, I did not have enough time to finish implementing this section. Furthermore, like with events, the user cannot yet delete a task. Lastly, I planned to implement the possibility to add tasks as events to the calendar or to add tasks to an event, as this was part of the design described in section 2.2.1.

5.3.7.3 Timer and Stopwatch

The timer and stopwatch functionality still has a lot of work to be done, since when using it, there are still some errors, as I described in the previous section. However, I also wanted to add an Statistics and History sections inside that part of the app. This was part of the design described in section 2.2.1, but also I think this would be an integral part of the app, as without this kind of data the user cannot reflect on how they can spend their time better.

Furthermore, I tried to implement a progress circle around the timer that would become smaller as the time in the timer passed, as this was one of the requirements gathered previously (see #7 in table 4.1). However, after many tries and many different libraries and packages used (*React Native Progress Circle* [35], *React Native Progress Wheel* [36], *React Native Animated Progress Circle* [33], and more), I could not make it work correctly. When using *React Native Progress Circle* [35], the progress circle would actually work, but the app would freeze. My next step would be to try to implement it from scratch.

5.4 Conclusion

In this chapter I explained the tools I used for the implementation of the app and the database structure. I also described how I implemented the app, the biggest issues I ran into, and the elements and functionalities I intended to implement but could not due to technical issues and time constraints.

Chapter 6

Evaluation of the app

In this chapter, I describe the process and results of the evaluation study for this app. These studies were approved by the University of Edinburgh School of Informatics, with RT 2019/18995. The methodology used during this phase is fully described in section 3.4.

6.1 Aims

The aim of this study is to evaluate the app as presented in chapter 5 and to cover Research Questions 3, 4 and 5:

3. How is the usability of the app perceived by Informatics students?
4. How is the usability of this app perceived by students of other disciplines at the University of Edinburgh?
5. What is the potential impact of the app for students, in terms of:
   (a) keeping up with their courses?
   (b) meeting deadlines?
   (c) reducing procrastination?
   (d) managing their personal life?

6.2 Participants

We recruited 22 participants in total for this study: eight 1st-year students, seven 2nd-year students, two 3rd-year students, two 4th-year students and three 5th-year students. We focused on recruiting more 1st and 2nd-year students since they are still figuring out what kind of time management system works better for them and we thought it would be a good idea to get their input. Among those students, six of them were student representatives, which we thought would have a better idea of what students in their year need on a day-to-day basis. Furthermore, three of the students had also

participated in a previous study regarding this project, either this year or last year: s14, s15 and s16. To see a full list of the students that participated, see appendix H.

6.3 Data Collection Methods

I conducted four Stakeholder Walkthrough sessions [62], which were followed by a few question and the SUS [49]. I also conducted seven Question Asking Protocol sessions [59], which were also followed by some questions and the SUS [49]. All studies were conducted over a university account of Zoom, and were audio and video recorded with the consent of all participants. I used this recording, together with note taking, to collect data. I decided to use Zoom [45] because it provides with automatic captioning of the audio which is very helpful when having to transcribe and analyse the data.

6.4 Materials

I composed two different Participant Information Sheets: one for the group sessions (see Appendix J) and one for the individual session (see Appendix I). These documents give the potential participants the necessary understanding of the procedures of the study and how to contact the people in charge of it. I used the same consent form I used in the previous study (see Appendix E), but this time I distributed it as an online form on Microsoft Forms [23].

For the sessions, I composed a set of tasks and questions for the participants (see appendix K). The tasks were formulated to cover as many of the functionalities as possible, such as adding an event, adding a task and starting a new timer. Furthermore, I included questions regarding the *Add to today* button and the ability to add tasks to the calendar or to an event, because, since they are not common features in time management apps, I wanted to gather feedback for such features. Lastly, I included other questions, like what elements they liked and disliked, if they would use the different functionalities of the app in particular, and if they would like to use the app in general for their own time management.

6.5 Procedure

When the potential participants were contacted, they were sent the Participant Information Sheet (PIS) by email (see Appendices I and J), so they could review the information before agreeing to the study. In the case of 1st and 2ndyear students, upon agreement of participation, they were sent an online poll to agree on a time and date that suited other participants from their same year. Once they agreed to participate and we agreed on a time and date, I would send them the link to access a scheduled Zoom [45] meeting for that time and date.

6.5.1 Group sessions

On the day of the session, I met the participants through Zoom [45]. I asked them to fill in the consent form. Once they had done that, I quickly went over the purpose of the study and what I wanted them to do. I sent them the link for the prototype on Figma [10]. Once everyone understood and everyone had opened the prototype on their laptops, I started recording and I sent them the first task over the chat feature of Zoom [45]. After they had all finished with the first task, I asked each to explain what steps they took to complete the task and then opened the floor to discussions about any difficulties they may have had, suggestions or comments about the task. We did the same with the other tasks.

After all tasks were completed, I asked them a few questions about what they liked and disliked, and about certain elements of the app. Lastly, I asked them to complete the SUS [49]. Once they were done, I asked if they had any more questions or comments, stopped the recording and thanked them for their participation.

6.5.2 Individual sessions

On the day of the session, I met the participants through Zoom [45]. I asked them to fill in the consent form. Once they had done that, I quickly went over the purpose of the study and what I wanted them to do. I opened the Android emulator on my computer and asked them to request remote control so they could use the emulator in my computer. I then shared my screen so that the screen of the emulator would be recorded, and I started recording.

I send them the tasks to complete, and while they were completing the tasks, they were talking out loud what they were doing and thinking. Sometimes I would ask a few questions regarding why they took a certain step and what they thought about certain elements. Once they were done with each task, I would ask them what they thought of it. Once all tasks were completed, I asked them a few more questions and then asked them to complete the SUS [49]. Once they were done, I stopped sharing my screen, I asked if they had any other questions or comments, stopped the recording, and thanked them for their participation.

6.6 Analysis

After the sessions, I transcribed every recording from the sessions using the captioning created by Zoom [45] and analysed the transcriptions together with the notes taken during these sessions, using NVivo 11 [29] using thematic analysis [55]. The themes and subthemes resulting from the analysis correspond to the sections and subsections described in the section below.

To analyse the SUS results, I followed the procedure described by John Brooke, when he first described the SUS [49]. To attribute the resulting score a meaning, I used the acceptability scale [48] (see table 6.1) and the grading scale [47] (see table 6.2).

Acceptability ranges	Mean SUS score
Not acceptable	below 50
Marginal	50 - 70
Acceptable	above 70

Table 6.1: SUS Acceptability Rating Scale (taken from [48])

Grade	Mean SUS score
F	below 60
D	60 - 69
C	70 - 79
B	80 - 89
A	90 - 100

Table 6.2: SUS Grading Scale (taken from [47])

6.7 Results

6.7.1 General

The overall opinion about the app was positive. Most of the students (18 / 22) stated that they would like to use the app once published and most of them (19 / 22) also said they would recommend it to a friend.

"Yes, I would use it. High school students and university students are always on the look for an app with a simple design while including all the functionalities they need, so I think it is really good for that." - s13

"I would try it out, I like that it has a calendar and tasks all in one, and I like the idea of adding tasks to an event, because I don't think a lot of productivity apps allow you to do that." - s3

Some students (6 / 22) said they felt that app was intuitive, some (8 / 22) said they liked how the app looked, and others (4 / 22) said they thought it would be easy to start using.

"I found it to be pretty intuitive, very standard as with other calendar apps, so it made sense." - s18

Furthermore, some elements participants liked were: the timer functionality (7 / 22), the colour-coding (5 / 22), the organisation of the home screen (4 / 22), the combination of different apps into one (4 / 22), the different categories (3 / 22), the similarity to other apps (2 / 22) and the ability to add a deadline to a task (2 / 22).

"I like the colour-coding, the categorisation and the labels, I like the level of organisation" - s6, one of the students outside of the School of Informatics

"It is a nice way to integrate them all together because I do think they are all very intertwined." - s6, one of the students outside of the School of Informatics

6.7.2 Home screen

As stated before, a few students (4 / 22) mentioned how they liked the organisation of the home screen.

"I also like the homepage, you are able to have everything but it is well separated and spaced out." - s12

"The organisation of the home page is very clear, I can clearly see what tasks I have and I can see everything that is coming up. It is very easy to read." - s15

However, a couple of students (2 / 22) thought that it would be good to be able to change how the home screen looks.

"I think it would be nice to make it possible to move the sections around in the home screen, or to remove or hide them." - s2, one of the students outside of the School of Informatics

6.7.3 Tasks

Most students (18 / 22) said they would use this app to manage their tasks. However, a couple of participants (2 / 22) found the difference between "Add to today", "Add deadline" and "Add to calendar" confusing.

"I found the difference between add to calendar and set the deadline a bit confusing. And there was the button to add to today, I find these a bit confusing." - s18

Furthermore, I asked the participants questions about certain elements of adding a task, such as the *Add to today* functionality and the ability to add tasks to the calendar or to an event, which are covered in the following sections, section 6.7.3.1 and 6.7.3.2.

6.7.3.1 *Add to today* functionality

This functionality was not part of the requirements, but it was my idea to add it, so I wanted to see if this functionality would be understood and if the participants would like it.

A few students either did not understand what it was for (2 / 22) or they thought the button was used to save the task (6 / 22), as the button is very prominent in the screen due to the blue colour, and it is the first button they see.

"It feels like the add to today button is to save the task, the way it looks is confusing, maybe it could be a checkbox, it should just look different than a button." - s6, one of the students outside of the School of Informatics

Some students (7 / 22) liked the idea, and some of these students (4 / 22) thought it would be good to be able to not only add to today, but also add to tomorrow or to another day.

"Is the Add to Today button fixed, or can we select add to Wednesday or add to Thursday?" - s13

"It would be nice if I could also add a task to tomorrow instead of just today." - s15, who had participated previously

6.7.3.2 Add tasks to the calendar or to an event

This was one of the requirements gathered during the first part of this project, but since it is a functionality that is not common, I wanted to know the opinion of the participants and see if they would find this functionality useful.

Most students (15 / 22) said they would find this feature useful.

"Yes, I would find that quite useful. It is a great addition to help you with your productivity and it helps you know what you need to do." - s12

"Yes, I feel the same. If you can sub-categorize what you need to do, then it helps you know how much time you need to allocate to that." - s13

"Yes, I would find it very useful, as it would allow me to further specify what I need to do during certain events. For example, if I created an event to study, maybe for some subject, it would be nice to specify what exactly I need to study as tasks." - s10

6.7.4 Timer and stopwatch

As mentioned before, some students (7 / 22) said they liked having the functionality of the timer and stopwatch on the app, and some other students (10 / 22) said they would use it to manage their studies and work.

"I really like the idea of the timer, that seems really useful and I would definitely use that, having the split breaks, so you can plan it out." - s18

"I think this is really good for how I work, and I think it will remind me to take a break, and I really like that." - s6, one of the students outside of the School of Informatics

However, a couple of students (2 / 22) said they found the timer and stopwatch functionality unnecessary.

"I do not think the timer and stopwatch functionality, I do not know if it needs to be on the app." - s1

"I think this app should be just for scheduling rather than something else. I think most people already have their own Pomodoro [50] apps on their phone." - s2, one of the students outside of the School of Informatics

Furthermore, a couple of students (2 / 22) said they have a hard time differentiating between a timer and a stopwatch.

"I usually forget the difference between a timer and a stopwatch." - s14, who had participated previously

6.7.5 Potential impact

This section covers Research Question 5 about how this app could potentially impact students. Participants were asked to rate the app from very helpful (5) to very unhelpful (1) in terms of 4 different aspects: keeping up with courses, meeting deadlines, reducing procrastination and managing personal life. The results can be seen in table 6.3.

Participant	Keep. up with courses	Meet. deadlines	Reduc. procr.	Manag. pers. life
s1	4	5	4	4
s2	4	5	4	4
s3	4	5	5	4
s4	4	5	2	4
s5	5	5	3	5
s6	5	5	5	5
s7	5	4	5	5
s8	4	4	3	5
s9	5	4	4	3
s10	5	5	5	4
s11	1	1	1	1
s12	5	5	2	3
s13	5	5	4	4
s14	4	5	4	3
s15	5	5	4	3
s16	5	5	4	5
s17	5	5	4	5
s18	5	5	4	4
s19	4	5	4	5
s20	5	5	4	4
s21	5	5	5	4
s22	5	5	4	5
Mean	**4.5**	**4.7**	**3.8**	**4.05**
Median	**5**	**5**	**4**	**4**

Table 6.3: Ratings given by students on how helpful the app could be to them

- Keeping up with courses (mean of 4.5 and median of 5): This aspect is the second most highly rated, with 14 participants rating it as very helpful, seven participants rating it as somewhat helpful and one participant, as very unhelpful.
- Meeting deadlines (mean of 4.7 and median of 5): This was the most highly rated, with 18 students rating it as very helpful, three students, as somewhat helpful, and one students rating it as very unhelpful.
- Reducing procrastination (mean of 3.8 and median of 4): This time only five participants rated this aspect with very helpful. Over half, 12 students, rated it as somewhat helpful, and the rest, five students, rated it as neither helpful or unhelpful, somewhat unhelpful or very unhelpful.

- Managing your personal life (mean of 4.05 and median of 4): For this aspect, eight said they would find the app very helpful, nine said somewhat helpful, four said neither help or unhelpful and one said very unhelpful.

Overall the results were positive, with most aspects having a mean above 4, except for reducing procrastination. This could be due to some participants feeling like using an app would increase their procrastination as they would be on their phone for longer, or being on their phone could lead them to go to other more distracting apps. Furthermore, there was an outlier in the results. One participant (s11) rated the app as very unhelpful in all categories, but gave it a punctuation of 95 in the SUS. This could either be because they found the usability of the app was good but they did not think it would be useful for them or maybe they meant to give it a different rating and made a mistake.

6.7.6 SUS results

This section covers Research Questions 3 and 4: how the usability of the app is perceived by students of Informatics as well as students outside of the school of Informatics.

Participant	SUS score	Informatics student
s1	92.5	Yes
s2	62.5	No
s3	97.5	Yes
s4	95	Yes
s5	90	Yes
s6	92.5	No
s7	85	Yes
s8	90	Yes
s9	85	Yes
s10	97.5	No
s11	95	Yes
s12	90	Yes
s13	92.5	Yes
s14	95	Yes
s15	95	Yes
s16	97.5	Yes
s17	95	No
s18	87.5	Yes
s19	90	Yes
s20	92.5	Yes
s21	87.5	Yes
s22	95	Yes
Mean	**90.90**	
Median	**92.5**	

Table 6.4: SUS results from the evaluation

The results of the SUS questionnaire can be seen in table 6.4. The score from most of the participants is above 90, with only five participants having a result below 90. The mean is 90.90 and the median is 92.5, with a maximum value of 97.5 and a minimum value of 62.5. This score rates the usability of the app as acceptable, according to the rating scale in table 6.1, and graded with an A, according to the grading scale in table 6.2. These results are much better compared to the results in the last evaluation last year, which were a mean of 75.59 and a median of 80. The reason for this could be that the usability of the app has improved after taking into consideration more of the requirements gathered, not only from last year, but also from the participatory design workshops conducted this year.

I also separated the results by participants who were part of the school of Informatics and those who were not. The mean for the Informatics students is 92.21, and the mean for student outside of Informatics is 86.5, below the overall mean. This could mean that there could be additional work done to improve the app for students outside of the school of Informatics. The median for both is 92.5, which is the same as the combined median.

6.8 Conclusion

In this chapter, I have described how I conducted the evaluation studies and I have presented the results of these studies.

In conclusion, the app has received good feedback and good SUS results, however, there are still some ways in which the app could improve. From this evaluation, I have managed to gather further requirements as well as new suggested changes, that should be considered for the further development of the app, which are presented in table 6.5.

The following are the requisites for the three different levels of priority:

- High: A requirement or suggested change will be prioritised as high if it is supported supported by at least six participants.
- Medium: A requirement will be prioritised as medium if it is supported by at least three participants.
- Low: supported by at least one participant.

#	Requirement/Suggestion	Priority	Reason For Priority
1	Being able to edit how the home screen is organised	Low	Supported by 2/22 participants.
2	Have the *Add to today* option not look like a button	High	Supported 6/22 participants.
3	*Add to today*: have other day options as well	Medium	Supported by 4/22 participants.
4	Make clearer what the difference is between a timer and stopwatch	Low	Supported by 2/22 participants.

Table 6.5: Changes or requirements suggested by the participants

Chapter 7

Discussion, future work and conclusion

7.1 Discussion

Overall, I'm satisfied with the methodology and tools I used throughout this project, especially for the design workshops I conducted at the beginning. I thought it was very useful and insightful to have the participants draw their own mock-ups of what they thought the timer and time-tracking functionality could look like. I think some of the participants were worried their drawings would not be very good or that their ideas were not good, which was a drawback of using this method, but they all provided me with very useful information.

Regarding the development of the app, even though I had issues with React Native [32] and Firebase [11], I think it was a good decision choosing them, since I thought the process of developing using React Native was more flexible than using Android Studio [3] in my experience, and I already had experience using Firebase, which helped. Not to mention all the documentation and resources that were available for these tools. Furthermore, this project gave me the opportunity to learn a lot about React Native [32] that I would not have otherwise.

During the evaluation, I was happy that we found so many students who agreed to participate, especially the students who were not part of the School of Informatics. I got better feedback and better SUS results than last year, which could mean that I am going in the right direction with the app. I also really enjoyed hearing the opinion of the students about the app and their ideas, which I thought were very interesting. It was encouraging seeing some of the participants excited to use this app.

Furthermore, due to the Covid-19 pandemic, I could not conduct my studies face-to-face, which made it a bit more difficult to do, but I do not think they limited me much in terms of getting the data I needed.

7.1.1 Limitations

The biggest limitations I encountered during this project were the issues I ran into while developing the app, which you can read about in section 5.3. They hindered the process extensively and kept me from implementing as many features as I intended and there is still work to be done to have a fully functioning app. However, even without these issues, I think I would not have had enough time to implement everything I wanted, not only because of my lack of experience implementing apps using React Native [32], but also because of the complexity of the app, which I really realised while I was developing it. I think this is something I should have paid more attention to last year. As I was designing the app, I should have kept in mind that I would have to implement this app, and that since I do not have much experience developing apps, I should have kept the app a bit more simple.

7.2 Future work

Due to the aforementioned limitations, there is still work to be done on this project. In the following sections I will describe how this project could continue.

7.2.1 Implementation

I think the first step moving forward would be to finish implementing the app with the functionalities that were already started: the calendar, task manager and timer/time-tracker, including as many of the requirements as possible. The next step would be to try to address the requirements that have been unaddressed during this part of the project, such as a reward system (requirement #22 in table A.1), help distributing work (requirement #21 in table A.1) or the gamification of the timer and time-tracker (requirement #1 in table 4.1), as well as some of the ideas that were included into the design last year that were not included in the implementation yet, such as the course-work manager.

7.2.2 Extracting course data from DRPS [9]

The Degree Regulations & Programmes of Study (DRPS) [9] provides information about the programmes of study and courses at the University of Edinburgh, such as the timetable for the courses. Since one of the first requirements was to have the contact hours automatically uploaded for an easy set up (requirement #2 in table A.1), my idea is that the user could select the courses they take from a list pulled from the DRPS and the app could pull the timetable data and upload it to the users' calendar automatically.

7.2.3 Adapting the app for any university students

As seen during the evaluation study, the SUS results for non-Informatics students were not as a high as the results for Informatics questions. This could have been affected by the lower amount of non-Informatics students compared to Informatics students, or

some other factors, but it would be worth finding out how this app could be adapted to be used by any university student.

7.3 Conclusion

During this study, I designed, implemented and evaluated a time management app for Informatics students. I started by researching development tools, database systems, and design methods. I conducted participatory design workshops with six Informatics students using mock-ups to design the timer and time-tracking functionality. Then, after choosing the tools for the development of the app, I implemented the functionalities described in 5.3. Lastly, I conducted evaluation studies with 17 Informatics students as well as five non-Informatics students and gathered new requirements. The overall feedback was positive and some students showed interest in using the app once it is available to the public.

7.3.1 RQ1: How can we design a timer functionality and enhance the time-tracking functionality design for the app, which had inconclusive outcomes last year, so that it suits Informatics students at the University of Edinburgh?

The answer to this question is described in chapter 4. In order to design a timer and time-tracking functionality that suits Informatics students, one should follow the requirements gathered on table 4.1.

7.3.2 RQ2: What are the technologies that we can use to implement this app?

The answer to this question is described in sections 2.5 and 2.6. For this app I considered using Android Studio [3] and React Native [32], and Cloud Firestore [5] and MongoDB [25] for the database. I chose React Native, since I was already familiar with JavaScript [21], the base for React Native, and because it provided a single path of development to have the app available for both Android and iOS. I also chose Cloud Firestore because I had already had experience with it in the past. More details are available in section 3.3.

7.3.3 RQ3: How is the usability of the app perceived by Informatics students?

The answer to this question is described in section 6.7.6. The usability was rated with a mean of 92.21 by Informatics students, which corresponds to a rating of Acceptable in the acceptability scale (see table 6.1) and a rating of A in the grading scale (see table 6.2).

7.3.4 RQ4: How is the usability of this app perceived by student of other disciplines at the University of Edinburgh?

The answer to this question is described in section 6.7.6. The usability was rated with a mean of 86.5 by students outside of the School of Informatics, which corresponds to a rating of Acceptable in the acceptability scale (see table 6.1) and a rating of B in the grading scale (see table 6.2). This is lower than the rating provided by the Informatics students. This could be due to the difference in number of participants, or other reasons.

7.3.5 RQ5: What is the potential impact of the app for students?

The answer to this question is described in section 6.7.5. Participants were asked to rate the app from very helpful (5) to very unhelpful (1) in terms of 4 different aspects: keeping up with courses, meeting deadlines, reducing procrastination and managing personal life.

The highest rated aspect was meeting deadlines (mean of 4.7 and median of 5), and the lowest, reducing procrastination (mean of 3.8 and median of 4). The reason for this could be than by using an app to manage their time, some students think they will spend a lot of time on the app, and will end up using it as a means of procrastination.

The other aspects, keeping up with courses and managing personal life, had means of 4.5 and 4.05 and medians of 5 and 4, respectively.

Bibliography

[1] Agenda - React Native Calendar. https://github.com/wix/react-native-calendars#agenda. Accessed: 25-03-2021.

[2] Android. https://www.android.com/intl/en_uk/. Accessed: 31-03-2021.

[3] Android Studio. https://developer.android.com/studio. Accessed: 31-03-2021.

[4] CheckBox - React Native Elements. https://reactnativeelements.com/docs/checkbox. Accessed: 25-03-2021.

[5] Cloud Firestore. https://firebase.google.com/docs/firestore. Accessed: 31-03-2021.

[6] Core Components and Native Components. https://reactnative.dev/docs/intro-react-native-components. Accessed: 23-03-2021.

[7] CSS (Cascading Style Sheets). https://www.w3.org/Style/CSS/Overview.en.html. Accessed: 11-04-2021.

[8] Datetimepicker. https://github.com/react-native-datetimepicker/datetimepicker. Accessed: 25-03-2021.

[9] Degree Regulations & Programmes Of Study - The University of Edinburgh. http://www.drps.ed.ac.uk/. Accessed: 04-04-2021.

[10] Figma. http://www.figma.com.

[11] Firebase. https://firebase.google.com/. Accessed: 31-03-2021.

[12] Flutter. https://flutter.dev/. Accessed: 11-04-2021.

[13] Forest. https://www.forestapp.cc/. Accessed: 16-02-2021.

[14] Forest - Stay Focused on the App Store. https://apps.apple.com/gb/app/forest-stay-focused/id866450515. Accessed: 07-04-2021.

[15] Formik. https://formik.org/. Accessed: 25-03-2021.

[16] A guide to mobile app development: Web vs. native vs. hybrid. https://clearbridgemobile.com/mobile-app-development-native-vs-web-vs-hybrid/. Accessed: 23-10-2020.

[17] HTML (HyperText Markup Language). https://html.spec.whatwg.org/. Accessed: 11-04-2021.

[18] Institute for Academic Development. https://www.ed.ac.uk/institute-academic-development.

[19] iStudiezPro. https://istudentpro.com/. Accessed: 07-04-2021.

[20] Java. https://www.oracle.com/java/technologies/. Accessed: 11-04-2021.

[21] Javascript. https://en.wikipedia.org/wiki/JavaScript. Accessed: 23-10-2020.

[22] Kotlin. https://kotlinlang.org/. Accessed: 11-04-2021.

[23] Microsoft Forms. https://forms.office.com. Accessed: 26-03-2021.

[24] Modal - React Native. https://reactnative.dev/docs/modal. Accessed: 25-03-2021.

[25] MongoDB. https://www.mongodb.com/. Accessed: 31-03-2021.

[26] MongoDB Realm. https://www.mongodb.com/realm. Accessed: 31-03-2021.

[27] myHomework. https://myhomeworkapp.com/. Accessed: 07-04-2021.

[28] MyStudyLife. https://www.mystudylife.com/. Accessed: 07-04-2021.

[29] Nvivo. https://www.qsrinternational.com/nvivo-qualitative-data-analysis-software/about/nvivo. Accessed: 16-02-2021.

[30] Python. https://www.python.org/. Accessed: 11-04-2021.

[31] React. https://reactjs.org/. Accessed: 23-10-2020.

[32] React Native. https://reactnative.dev/. Accessed: 23-10-2020.

[33] React Native Animated Progress Circle. https://github.com/simonsteer/rn-animated-progress-circle. Accessed: 25-03-2021.

[34] React Native Numeric Input. https://github.com/himelbrand/react-native-numeric-input. Accessed: 25-03-2021.

[35] React Native Progress Circle. https://github.com/MrToph/react-native-progress-circle. Accessed: 25-03-2021.

[36] React Native Progress Wheel. https://github.com/RobertFOConnor/react-native-progress-wheel. Accessed: 25-03-2021.

[37] React Native Stopwatch Timer. https://github.com/michaeljstevens/react-native-stopwatch-timer. Accessed: 25-03-2021.

[38] Ruby. https://www.ruby-lang.org/en/. Accessed: 11-04-2021.

[39] Storyboarding — Open Innovation Toolkit. https://toolkit.mozilla.org/method/storyboarding/.

[40] Student Counselling.

[41] study hub - institute for academic development.

[42] Swift. https://developer.apple.com/swift/. Accessed: 11-04-2021.

[43] XCode. https://developer.apple.com/xcode/. Accessed: 11-04-2021.

[44] Yup. https://github.com/jquense/yup. Accessed: 01-04-2021.

[45] Zoom. https://zoom.us/. Accessed: 16-02-2021.

[46] Oscar Axelsson and Fredrik Carlström. Evaluation targeting react native in comparison to native mobile development. 2016.

[47] Aaron Bangor, Philip Kortum, and James Miller. Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of usability studies*, 4(3):114–123, 2009.

[48] Aaron Bangor, Philip T Kortum, and James T Miller. An empirical evaluation of the system usability scale. *Intl. Journal of Human–Computer Interaction*, 24(6):574–594, 2008.

[49] John Brooke. Sus: a "quick and dirty'usability. *Usability evaluation in industry*, page 189, 1996.

[50] Francesco Cirillo. *The pomodoro technique*. Lulu. com, 2009.

[51] Brigitte Claessens, Wendelien Eerde, Christel Rutte, and Robert Roe. A review of time management literature. *Personnel Review*, 36, 02 2007.

[52] Wallace Clark. *The Gantt chart: A working tool of management*. Ronald Press Company, 1922.

[53] Pelle Ehn and Morten Kyng. *Cardboard Computers: Mocking-It-up or Hands-on the Future*, page 169–196. L. Erlbaum Associates Inc., USA, 1992.

[54] Chung-Shun Feng, Tsu-Wu Hu, Yi-Ru Chen, and Chu-Yin Tsai. Research on improving the user experience and usability evaluation of tomato work method app-using forest app. In *Proceedings of the 2019 7th International Conference on Information Technology: IoT and Smart City*, ICIT 2019, page 147–151, New York, NY, USA, 2019. Association for Computing Machinery.

[55] Greg Guest, Kathleen M MacQueen, and Emily E Namey. *Applied thematic analysis*. sage publications, 2011.

[56] Kim Halskov and Peter Aiken. Experiences using cooperative interactive storyboard prototyping. *Commun. ACM*, 36:57–64, 06 1993.

[57] Rex Hartson and Pardha Pyla. *The UX Book: Process and Guidelines for Ensuring a Quality User Experience*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition, 2012.

[58] K. Maike Hecht and Susanne Maass. Teaching participatory design. In *Proceedings of the Tenth Anniversary Conference on Participatory Design 2008*, PDC '08, page 166–169, USA, 2008. Indiana University.

[59] Takashi Kato. What "question-asking protocols" can say about the user interface. *International Journal of Man-Machine Studies*, 25(6):659–673, 1986.

[60] Simon Larose, Annie Bernier, and George M Tarabulsy. Attachment state of mind, learning dispositions, and academic performance during the college transition. *Developmental Psychology*, 41(1):281, 2005.

[61] Therese Macan, Comila Shahani, Robert Dipboye, and Amanda Phillips. College students' time management: Correlations with academic performance and stress. *Journal of Educational Psychology*, 82:760–768, 12 1990.

[62] Elisa Miller. Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions, 2012.

[63] Ranjita Misra and Michelle McKean. College students' academic stress and its relation to their anxiety, time management, and leisure satisfaction. *American journal of Health studies*, 16(1):41, 2000.

[64] Celeste Lyn Paul. A modified delphi approach to a new card sorting methodology. *J. Usability Studies*, 4(1):7–30, November 2008.

[65] Mark Snyder. Self-monitoring of expressive behavior. *Journal of personality and social psychology*, 30(4):526, 1974.

[66] Clay Spinuzzi. The methodology of participatory design. *Technical Communication*, 52:163–174, 05 2005.

[67] Xiaofeng Wang, Federico Gobbo, and Michael Lane. *Turning Time from Enemy into an Ally Using the Pomodoro Technique*, pages 149–166. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

Appendices

Appendix A

Table of requirements obtained in part 1 of the project

#	Requirement	Priority	Reason For Priority
1	Calendar system	High	Supported by 13/15 students, 2/2 staff members and research.
2	Have contact hours automatically uploaded	High	Supported by 5/15 students. It will make it easier to set up.
3	Have assignment deadlines automatically uploaded	Medium	Supported by 2/15 students and 1/2 staff members. It would make it easier to set up.
4	Use a Gantt Chart [52] to display assignment deadlines	Low	Supported by 1/15 students.
5	Reminders and notifications for events	High	Supported by 6/15 students and 1/2 staff members.
6	Colour-coding	High	Supported by 7/15 students.
7	Recurring events	Medium	Useful for events that repeat, like lectures, and very common in calendar apps. Supported by 2/15 students.
8	Different calendar viewing options (month, week, 3-days, day)	Medium	Supported by 2/15 students and very common in calendar apps.
9	Synchronise calendars from other accounts	Medium	Supported by 1/15 students and 1/17 students.
10	To-do lists	High	Supported by 10/15 students and by research.
11	Add deadlines to tasks	High	Supported by 2/15 students. Useful for students to stay on top of their tasks.
12	Tick off tasks to mark them as completed	High	Supported by 2/15 students. Included in all task management apps, very basic feature.
13	Prioritise tasks	High	Supported used by 10/15 students, 1/2 staff members and research.
14	Group tasks	High	Supported by 2/15 students and research.
15	Task progress bar	Medium	Supported by 2/15 students.
16	Quickly adding tasks	Medium	Supported by 2/15 students.
17	Have tasks linked to calendar events	High	Supported by 7/15 students.
18	Ask the user after the event if the tasks were completed	Low	Supported by 2/15 students.
19	Tracking time	High	Supported by 3/15 students, 2/2 staff member and research.
20	Feedback time tracking information to course organisers	Low	Mentioned by 1/2 staff.
21	Help distribute work	Medium	Supported by 3/15 students and 1/2 staff members
22	Reward System	Medium	Supported by 3/15 students and 1/2 staff members.
23	Easy to use and set-up	High	Supported by 7/15 students.

Table A.1: New list of requirements

#	Requirement	Priority	Reason For Priority
24	Have everything in one place	Medium	Supported by 3/15 students.
25	Monitor how much time the user spends on something and notify if behind schedule	Low	Supported by 1/2 staff members
26	Life management	Medium	Supported by 1/15 students and 1/2 staff members.
27	Feedback	Low	Supported by 1/2 staff members
28	Have visualization of time tracker information	Low	Supported by 1/6 participants.
29	Customisable colours	Low	Supported by 2/17 students.
30	Confirmation windows after creating or deleting an element	High	Supported by 1/17 students, 2/2 HCI experts and would cover one of Nielson's heuristics, "error prevention".
31	Users should be able to share tasks and events with other users	Low	Supported by 2/17 students.
32	Course overview	Medium	Supported by 1/2 HCI expert.
33	Suggestions of the app on when to work	Low	Supported by 1/2 staff members.
34	Study timer	High	Supported by 1/17 students and research.
35	Edit/delete events or tasks from dashboard	Low	Supported by 1/17 students.
36	Help choosing today's tasks	Medium	Supported by 1/2 HCI experts.
37	Notifications about tasks left to complete	Low	Supported by 1/17 students.
38	Word auto-complete	Low	Supported by 1/17 students.
39	Have multiple collaborators in one task	Low	Supported by 1/17 students.
40	Start time tracker from event	Medium	Supported by 2/17 students and 1/2 staff members.
41	Send notifications to help the user not forget to stop the time tracker	Medium	Supported by 2/17 students.
42	Filter and group trackers	Low	Supported by 1/17 students.
43	Start time tracker without specifying label	Low	Supported by 1/17 students.
44	Display how much time is split between courses	Low	Supported by 1/17 students.
45	Help managing revision	Low	Supported by 1/17 students.
46	Suggestions from the app on when to work	Low	Supported by 1/2 staff members.
47	Progress bar for achievements	Medium	Supported by 2/17 students and 1/2 staff members.

Table A.2: New list of requirements (cont.)

Appendix B

Table of suggested changes obtained in part 1 of the project

#	Suggestion	Priority	Reason For Priority
1	Move Delete button from the screen to edit events and tasks	High	Supported by 5/6 participants.
2	Remove class schedule from menu to make it less crowded	Medium	Supported 2/6 participants.
3	Add option to add tasks or events from the dashboard	Medium	Supported by 1/6 participants, and I think it would be useful for adding things quicker to the app.
4	Add other options when sliding a task to the sides	Low	Supported by 1/6 participants.
5	Font should be smaller	Medium	Supported by 1/17 students and 1/2 HCI experts.
6	Make scrolling more visible	Medium	Supported by 2/17 students and 1/2 HCI experts.
7	Create new tasks when linking tasks to event	Medium	Supported by 2/17 students and 1/2 staff members.
8	Show the time when the event ends	Low	Supported by 1/17 students.
9	Menu should not change	Medium	Supported by 2/17 students.
10	Add delete icon to subtasks	Low	Supported by 1/17 students.
11	Change colour of selected tab to white	Medium	Supported by 1/2 staff members and 1/2 HCI experts.
12	Make start tracker button smaller	Medium	Supported by 2/17 students.
13	Include legend in bar chart	Medium	Supported by 1/17 students and 1/2 HCI experts.
14	Extending the Gantt chart [**?**]	Medium	Supported by 1/2 HCI experts.
15	Renaming the feature of Coursework	Low	Supported by 1/17 students.
16	Change designs of incomplete badges	Low	Supported by 1/17 students.

Table B.1: Changes suggested by the participants

Appendix C

Participants of first study

Participant #	Year of study	Had participated in previous studies?
s1	Year 4	No
s2	Year 4	No
s3	Year 4	No
s4	Year 5	Yes
s5	Year 4	Yes
s6	Year 5	Yes

Table C.1: Table showing the participants who took part in the design workshops

Appendix D

Participant Information Sheet for first study

Participant Information Sheet

Project title:	Designing a Time Management app for and with Informatics students
Principal investigator:	Cristina Alexandru
Researcher collecting data:	Julia Castillo Trujillo
Funder (if applicable):	No

This study was certified according to the Informatics Research Ethics Process, RT number 2019/94240.Please take time to read the following information carefully. You should keep this page for your records.

Who are the researchers?

The researchers of the study are Julia Castillo Trujillo who is an undergraduate student in the University of Edinburgh School of Informatics and Cristina Alexandru who is her supervisor.

What is the purpose of the study?

This study is part of an undergraduate project, which aims to design and develop a time management app that will address Informatics students' need for better managing their time at university. The purpose of this study is to gain a better understanding of how users would see a timer and time-tracking functionality work on the app.

Why have I been asked to take part?

The aim of this project is to design an app to help Informatics students at the University of Edinburgh. As a potential end user for such an app, you can help us improve our design to create something which will address your needs and, ultimately, help you with your studies and your life at university.

Do I have to take part?

No – participation in this study is entirely up to you. You can withdraw from the study at any time, without giving a reason. Your rights will not be affected. If you wish to withdraw, contact the PI. We will stop using your data in any publications or presentations submitted after you have withdrawn consent. However, we will keep copies of your original consent, and of your withdrawal request.

What will happen if I decide to take part?

This will be conducted online and in a group. First we will explain what the timer and time-tracking functionality is intended to be according to user requirements gathered in the previous part of the project, and we will ask you to draw a mock-up of how you think this functionality should be presented to the user and once you are finished you will send it to me. After every participant is done, every participant will explain their own mock-up, talking about the different elements they decided to add and why, and in the meantime the other participants will write down what elements they like the most of the other participants' mock-ups. Once all mock-ups have been reviewed, we will have a discussion together on what elements have been the most liked and what combination of elements should be present in the design. This process shouldn't last longer than one hour and a half, and it will be video and audio recorded with your consent.

Are there any risks associated with taking part?

There are no significant risks associated with participation and it will not affect your studies in any way.

Are there any benefits associated with taking part?

The app is to be used by students of the School of Informatics like you, so by participating in this study you'll have the chance to influence the design of the app.

What will happen to the results of this study?

The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will be anonymized: We will remove any information that could, in our assessment, allow anyone to identify you. With your

consent, information can also be used for future research. Your data will be archived for a minimum of 2 years.

Data protection and confidentiality.

Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a unique participant number rather than by name. Your data will only be viewed by the researcher/research team: Julia Castillo Trujillo and Cristina Alexandru.

All electronic data will be stored on a password-protected encrypted computer, on the School of Informatics' secure file servers. Your consent information will be kept separately from your responses in order to minimise risk.

What are my data protection rights?

The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance to the Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to Julia at s165774@sms.ed.ac.uk.

Who can I contact?

If you have any further questions about the study, please contact the principal investigator, Cristina (calexan4@staffmail.ed.ac.uk). If you wish to make a complaint about the study, please contact inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint.

Updated information.

If the research project changes in any way, an updated Participant Information Sheet will be sent by email to you by Julia.

Alternative formats.

To request this document in an alternative format, such as large print or on coloured paper, please contact Julia (s1651774@sms.ed.ac.uk).

General information.

For general information about how we use your data, go to: edin.ac/privacy-research

Appendix E

Consent form

Participant number:_____

Participant Consent Form

Project title:	Designing a Time Management App for and with Informatics students
Principal investigator (PI):	Cristina Alexandru
Researcher:	Julia Castillo Trujillo
PI contact details:	calexan4@staffmail.ed.ac.uk

Please tick yes or no for each of these statements.

		Yes	No
1.	I confirm that I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.		
2.	I understand that my participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.		
3.	I agree to being audio and video recorded.		
4.	I consent to my anonymised data being used in academic publications and presentations.		
5.	I understand that my anonymised data can be stored for a minimum of two years		
6.	I allow my data to be used in future ethically approved research.		
7.	I agree to take part in this study.		

Name of person giving consent	Date dd/mm/yy	Signature

Name of person taking consent	Date dd/mm/yy	Signature

THE UNIVERSITY of EDINBURGH
informatics

Appendix F

Script for first study

Hello everyone. My name is Julia and I will be running today's workshop.

First, I would like to ask you to fill in the consent form and send it over to me, if you haven't done so yet. If everyone has done so, I will now start recording this session.

First, let me tell you what is going to happen today.

First I'm going to explain to you a little bit about my project and about a specific functionality of the app that I want to create. After I explain all of this, I would like you to draw on a piece of paper or a tablet (whatever you prefer) how this functionality would look like to you. Then you will send me your drawings and I will let all of you explain your design a little bit, and lastly we will discuss the different elements of the design to see which elements I should be including in my app.

So before I start, could you all grab some paper, a notebook or a tablet where you can draw?

My project consists of designing a time management app for Informatics students like you. Last year I conducted a few studies for requirement gathering and to evaluate the design I created. However, there is a part of the app where people's opinion would sometimes contradict themselves, so I thought I would run this workshop to have a clearer idea of what the design of this functionality would look like.

This functionality consists of a time tracker combined with a study timer. The time tracker would simply work like a stopwatch, it would just tell you how much time you spent on a certain activity. This activity can be a coursework, doing chores or going for a run. This would hopefully make the student aware of how much time they spend doing certain things and that way they can plan their time better. For example, a student might allocate 1 hour in their schedule to complete tutorial exercises, but then they would see on the timer that it actually took 2 hours and 30 min, so next time they can allocate the right time in their schedule. Do you have any questions?

The study timer would follow the Pomodoro technique, which states that the best way to work is working in 25 minute intervals with a 5 minute break in between. In this case, the student would be able to change the time interval to whatever works better

for them. This functionality will hopefully help the student to focus and to take more breaks. Do you have any questions?

Lastly, this would also include a ”Statistics” section where the user would be able to see how much time they spend on different activities, either as part of a time tracker or a timer.

I will also let you know that this functionality would be accessible by a drawer menu, i.e. a menu that slides in from the left that can be accessed by a burger icon (3 horizontal lines on top of each other) which is present in top left corner of most screens of the app.

Do you have any questions? Okay, now I will ask you some questions to help you start thinking about the design. I will now give you around 10min to draw it, but I can give you more time if you need it.

Okay, now that everyone has finished drawing, please send it to me. You can take a picture and send it to me over email or upload it to Files in Teams.

Thank you. Now that I have all the drawings I would like to ask each one of you to explain what you added to the design and why. Everyone else, please make a note of any element that you like that you think would be useful in an app like this one, or any elements that you don’t think would be useful and why.

Okay, now that everyone has explained their own design, I would like to discuss the different elements that each has included in their design.

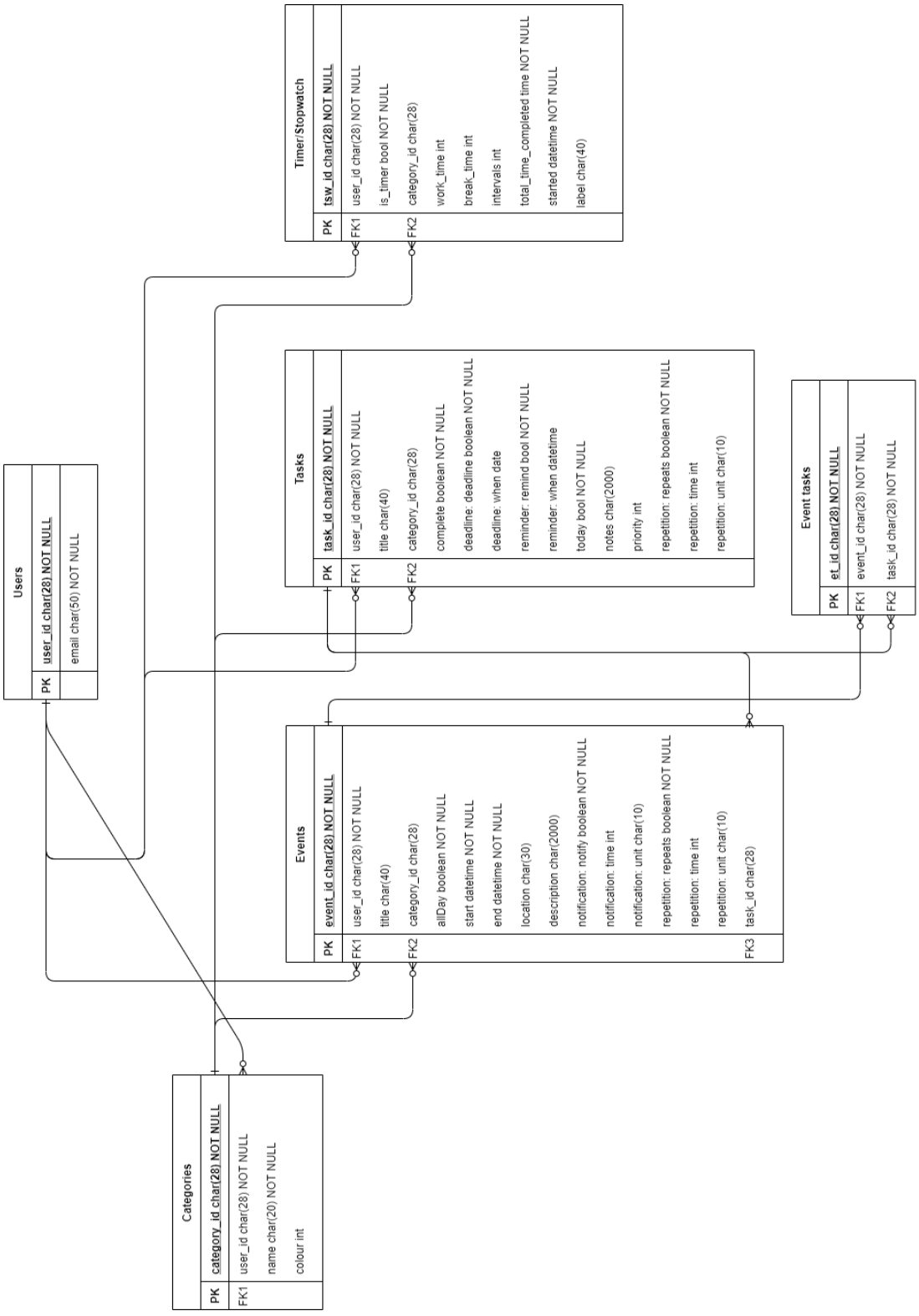
- Are there any elements that you thought were really interesting? Why?
- Is there anything you disagreed with? Why?

Do you have any other questions or comments before we finish?

That is the end of the session. Thank you all for participating.

Appendix G

Entity-Relationship Diagram


Users
PK user_id char(28) NOT NULL
email char(50) NOT NULL
Timer/Stopwatch
PK tsw_id char(28) NOT NULL
FK1 user_id char(28) NOT NULL
is_timer bool NOT NULL
FK2 category_id char(28)
work_time int
break_time int
intervals int
total_time_completed time NOT NULL
started datetime NOT NULL
label char(40)
Tasks
PK task_id char(28) NOT NULL
FK1 user_id char(28) NOT NULL
title char(40)
FK2 category_id char(28)
complete boolean NOT NULL
deadline: deadline boolean NOT NULL
deadline: when date
reminder: remind bool NOT NULL
reminder: when datetime
today bool NOT NULL
notes char(2000)
priority int
repetition: repeats boolean NOT NULL
repetition: time int
repetition: unit char(10)
Event tasks
PK et_id char(28) NOT NULL
FK1 event_id char(28) NOT NULL
FK2 task_id char(28) NOT NULL
Events
PK event_id char(28) NOT NULL
FK1 user_id char(28) NOT NULL
title char(40)
FK2 category_id char(28)
allDay boolean NOT NULL
start datetime NOT NULL
end datetime NOT NULL
location char(30)
description char(2000)
notification: notify boolean NOT NULL
notification: time int
notification: unit char(10)
repetition: repeats boolean NOT NULL
repetition: time int
repetition: unit char(10)
FK3 task_id char(28)
Categories
PK category_id char(28) NOT NULL
FK1 user_id char(28) NOT NULL
name char(20) NOT NULL
colour int

Appendix H

Participants of the evaluation of the app

#	Year of study	School of Informatics?	Student Representative?	Participated before?
s1	Year 2	Yes	No	No
s2	Year 2	No	No	No
s3	Year 2	Yes	No	No
s4	Year 2	Yes	No	No
s5	Year 2	Yes	Yes	No
s6	Year 5	No	No	No
s7	Year 3	No	No	No
s8	Year 2	Yes	Yes	No
s9	Year 2	Yes	Yes	No
s10	Year 3	No	No	No
s11	Year 1	Yes	Yes	No
s12	Year 1	Yes	Yes	No
s13	Year 1	Yes	Yes	No
s14	Year 5	Yes	No	Yes
s15	Year 4	Yes	No	Yes
s16	Year 5	Yes	No	Yes
s17	Year 4	No	No	No
s18	Year 1	Yes	No	No
s19	Year 1	Yes	No	No
s20	Year 1	Yes	No	No
s21	Year 1	Yes	No	No
s22	Year 1	Yes	No	No

Table H.1: Table showing the participants who took part in the evaluation study for the app

Appendix I

Participant Information Sheet - Question Asking Protocol

Participant Information Sheet

Project title:	Designing and Developing a Time Management app for and with Informatics students
Principal investigator:	Cristina Alexandru
Researcher collecting data:	Julia Castillo Trujillo
Funder (if applicable):	No

This study was certified according to the Informatics Research Ethics Process, RT number 2019/18995.Please take time to read the following information carefully. You should keep this page for your records.

Who are the researchers?

The researchers of the study are Julia Castillo Trujillo who is an undergraduate student in the University of Edinburgh School of Informatics and Cristina Alexandru who is her supervisor.

What is the purpose of the study?

This study is part of an undergraduate project, which aims to design and develop a time management app that will address students' need for better managing their time at university. The purpose of this study is to evaluate the app, in order to gather feedback and suggestions regarding the usability of the app, and to learn about its potential impact on students.

Why have I been asked to take part?

The aim of this project is to design an app to help students at the University of Edinburgh. As a potential end user for such an app, you can help us improve our design to create something which will address your needs and, ultimately, help you with your studies and your life at university.

Do I have to take part?

No – participation in this study is entirely up to you. You can withdraw from the study at any time, without giving a reason. Your rights will not be affected. If you wish to

withdraw, contact the PI. We will stop using your data in any publications or presentations submitted after you have withdrawn consent. However, we will keep copies of your original consent, and of your withdrawal request.

What will happen if I decide to take part?

This study will be conducted online, on Zoom via a university account. We will ask you to complete certain tasks on the app. While you complete these tasks, you should say out loud what you are thinking and doing. I also might ask you a few questions during the tasks. Once you have completed the tasks, I will ask you a few more questions about the app, its usability and its potential impact, and I will ask you to fill in a short questionnaire about the usability of the system. This process shouldn't last longer than half an hour, and it will be video and audio recorded with your consent.

Are there any risks associated with taking part?

There are no significant risks associated with participation and it will not affect your studies in any way.

Are there any benefits associated with taking part?

The app is to be used by students of the School of Informatics like you, so by participating in this study you'll have the chance to influence the design of the app.

What will happen to the results of this study?

The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will be anonymized: We will remove any information that could, in our assessment, allow anyone to identify you. With your consent, information can also be used for future research. Your data will be archived for a minimum of 2 years.

Data protection and confidentiality.

Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a unique participant number rather than by name. Your data will only be viewed by the researcher/research team: Julia Castillo Trujillo and Cristina Alexandru.

All electronic data will be stored on a password-protected encrypted computer, on the School of Informatics' secure file servers. Your consent information will be kept separately from your responses in order to minimise risk.

What are my data protection rights?

The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance to the Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to Julia at s165774@sms.ed.ac.uk.

Who can I contact?

If you have any further questions about the study, please contact the principal investigator, Cristina (calexan4@staffmail.ed.ac.uk). If you wish to make a complaint about the study, please contact inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint.

Updated information.

If the research project changes in any way, an updated Participant Information Sheet will be sent by email to you by Julia.

Alternative formats.

To request this document in an alternative format, such as large print or on coloured paper, please contact Julia (s1651774@sms.ed.ac.uk).

General information.

For general information about how we use your data, go to: edin.ac/privacy-research

Appendix J

Participant Information Sheet - Stakeholder Walkthrough

Participant Information Sheet

Project title:	Designing and Developing a Time Management app for and with Informatics students
Principal investigator:	Cristina Alexandru
Researcher collecting data:	Julia Castillo Trujillo
Funder (if applicable):	No

This study was certified according to the Informatics Research Ethics Process, RT number 2019/18995.Please take time to read the following information carefully. You should keep this page for your records.

Who are the researchers?

The researchers of the study are Julia Castillo Trujillo who is an undergraduate student in the University of Edinburgh School of Informatics and Cristina Alexandru who is her supervisor.

What is the purpose of the study?

This study is part of an undergraduate project, which aims to design and develop a time management app that will address students' need for better managing their time at university. The purpose of this study is to evaluate the app, in order to gather feedback and suggestions regarding the usability of the app, and to learn about its potential impact on students.

Why have I been asked to take part?

The aim of this project is to design an app to help students at the University of Edinburgh. As a potential end user for such an app, you can help us improve our design to create something which will address your needs and, ultimately, help you with your studies and your life at university.

Do I have to take part?

No – participation in this study is entirely up to you. You can withdraw from the study at any time, without giving a reason. Your rights will not be affected. If you wish to

withdraw, contact the PI. We will stop using your data in any publications or presentations submitted after you have withdrawn consent. However, we will keep copies of your original consent, and of your withdrawal request.

What will happen if I decide to take part?

This study will be conducted online, on Zoom via a university account. This will be conducted in a group. We will ask you to perform some tasks on the prototype and write down every decision you're making and any problems you run into. After every participant is done, we will ask that all of you verbalize your written responses and discuss potential usability problems. After that, we will ask you to fill in a short survey regarding the general usability of the design and the potential impact of the system, and we will ask you some questions regarding any difficulties you may have encountered while making use of the prototype and your opinion on the prototype. This process shouldn't last longer than one hour and it will be audio-recorded with your consent.

Are there any risks associated with taking part?

There are no significant risks associated with participation and it will not affect your studies in any way.

Are there any benefits associated with taking part?

The app is to be used by students of the School of Informatics like you, so by participating in this study you'll have the chance to influence the design of the app.

What will happen to the results of this study?

The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will be anonymized: We will remove any information that could, in our assessment, allow anyone to identify you. With your consent, information can also be used for future research. Your data will be archived for a minimum of 2 years.

Data protection and confidentiality.

Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a

unique participant number rather than by name. Your data will only be viewed by the researcher/research team: Julia Castillo Trujillo and Cristina Alexandru.

All electronic data will be stored on a password-protected encrypted computer, on the School of Informatics' secure file servers. Your consent information will be kept separately from your responses in order to minimise risk.

What are my data protection rights?

The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance to the Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to Julia at s165774@sms.ed.ac.uk.

Who can I contact?

If you have any further questions about the study, please contact the principal investigator, Cristina (calexan4@staffmail.ed.ac.uk). If you wish to make a complaint about the study, please contact inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint.

Updated information.

If the research project changes in any way, an updated Participant Information Sheet will be sent by email to you by Julia.

Alternative formats.

To request this document in an alternative format, such as large print or on coloured paper, please contact Julia (s1651774@sms.ed.ac.uk).

General information.

For general information about how we use your data, go to: edin.ac/privacy-research

Appendix K

Tasks and Questions for the evaluation study

K.1 Tasks

1. You have a meeting with your dissertation supervisor today at 4pm and you would like to have it in your calendar and would also like to get a reminder 10 minutes before the meeting. Add it to your calendar and set the reminder.
2. Your supervisor has told you to finish the background chapter for your dissertation by the end of the week, but you want to do it today. Add it to your tasks for today, but set the deadline for Sunday.
3. You want to spend 1 hour working on this chapter but would like to take a 5 min break after 30 min of working. Set a timer for it.

K.2 Questions after each task

1. Can you shortly explain what you did to complete the task?
2. How did you find the task?
3. Is there anything that did not make sense to you?

K.3 Questions after tasks are completed

1. What did you like about the app?
2. What did you dislike?
3. Do you have any suggestions on how the app could improve?
4. What is your opinion of the ”Add to Today” button?

5. One of the functionalities that is unique to this app is that you can add tasks to an event as well as add a task to the calendar (still working on implementing it), how useful would you find this in your day to day or for university?
6. Do you think you would use this app?
7. If you were to use this app every day, how much would you envisage it would impact you in terms of (rate it from very helpful to very unhelpful):
    - Keeping up with your courses
    - Meeting deadlines
    - Reducing procrastination
    - Managing your personal life
8. Would you use this app to keep a schedule?
9. Would you use this app to manage your tasks?
10. Would you use the timer and stopwatch functionalities to manage your studying and work?
11. Would you recommend it to a friend?
12. Are you from the School of Informatics?