Timetable Mobile Application

Oskar Ciebien

B.Sc.(Hons) in Computing in Software Development

April 6, 2022

Final Year Project

Advised by: Mr. Martin Hynes

Department of Computer Science and Applied Physics
Atlantic Technological University (ATU)
Formerly Galway-Mayo Institute of Technology (GMIT)

Contents

List of Figures

About this project

Abstract This project is a Timetable Application developed for Mobile Devices. It differs from some of the other timetable applications currently available on the App Store and Google Play Store, because it has a common and a reliable security feature. This feature is the Google Two Factor Authentication. The user can disable or enable this feature at their own discretion. This Timetable Application also has a customisation feature which is the dark mode. This dark mode allows the user to switch between dark and light modes in the app. It changes the application theme colours from dark to light and vice versa. This feature will provide ease of use at any time of the day. Most importantly this mobile application will have easy to use Front-End interface. With a menu at the bottom of the screen with options to be picked from. In the application the user will have many screens to choose from. The Timetable screen, which displays the weekly schedule. The settings screen which offers settings in relation to the securities and the choice of theme within the application as well as account options. There is also the home page screen which displays the information about the application and the author of the application. I have used React Native to develop this application and Firebase for the Back-End of this Timetable Application, which will store all of the user information, settings, and timetable data. The database merges whenever an internet connection is established with the device. The database that I have used is also real-time, which allows for instant display of data when it has been modified.

Authors Oskar Ciebien is a final year student at Atlantic Technological University (ATU) formely known as Galway-Mayo Institute of Technology (GMIT). Currently pursuing a Bachelor of Science (Honours) in Computing in Software Development.

Chapter 1

Introduction

This is the introduction chapter of this dissertation. It outlines the layout, the objectives, the redirection to the source code, the project timeline and the project difficulties as well as the learning outcomes of the Applied Project and Minor Dissertation module.

1.1 Layout of the Dissertation

The following chapters will cover different aspects of the dissertation:

- **Chapter 1 - Introduction -** Chapter 1 which is the current chapter, will describe the project objectives, the redirection to the project's source code and documentation, the projects timeline and difficulties and finally the learning outcomes of the Applied Project and Minor Dissertation module.
- **Chapter 2 - Context -** Chapter 2 will explain the overall planning of the project.
- **Chapter 3 - Methodology -** This chapter will discuss the approach on the project, the tools used while developing the project, source control, the meetings conducted with the supervisor and the research which was done throughout the process of creating this project.
- **Chapter 4 - Technology Review -** Chapter 4 will provide a review on all the technologies and the languages that have been used.
- **Chapter 5 - System Design -** Will describe the project as a whole, the design and applications developed.

- **Chapter 6 - System Evaluation -** Will discuss the types of testing that have been carried out while developing the application.
- **Chapter 7 - Conclusion -** The final chapter will conclude the project and discuss the evaluation, objectives, downfalls and the improvements that could have been done on the project.

1.2 Project Source Code and Documentation

The project source code is available at a GitHub repository, under the following link `https://github.com/Oskar-Ciebien/Timetable_Application`.

A brief description of the project, the project requirements and instructions are available at the repository in the README file.

All the documentation and videos are available at the repository in the folder called Documents.

1.3 Goal of the Project

The goal of this project was to create a Timetable application, which would differ from most of the timetable applications currently available on Google Play Store or Apple App Store.

The project will also allow the user to register and log in using email and password. Thanks to this authentication individual information from that account will be saved to a database and could be accessed on any device after logging into the account.

The goal of the timetable was to be editable with or without an internet connection, but to sync whenever an internet connection would be established.

At the start the initial idea was to develop a mobile timetable application and to publish it on Google Play Store, as the application is mainly targeted at Android Devices. This idea will hopefully come true, as the application will most likely stay updated and worked on after submission.

1.4 Project Timeline

1.5 Project Difficulties

As I was developing the timetable application I have faced many difficulties. The difficulties that I have faced are as follows:

- Get familiar with the technologies that I have chosen - React Native, Firebase and Android Studio. I never used any of them before. I sometimes found it difficult to understand the logic of those, but thanks to this course I was able to adapt to new technologies quickly.
- My first project idea. I decided to restart the project completely after the Christmas review. It was all done to make the project seem and work better.
- Time and priorities were definitely large enemies while developing this project. I sometimes was forced to decide what to focus on first, or how much time I should spend developing a feature and if I was not able to finish it in the given time, then I should focus on something else. This solution worked very well for me and enabled me to be more efficient.

1.6 Learning Outcomes

Applied Project and Minor Dissertation module offers many learning outcomes. Which are very important when it comes to future job search or further studies in the field of computer science. The learning outcomes of this module are as follows:

- Demonstration of the application with the use of appropriate research methodologies and techniques which are all related to software development.
- Demonstration of the awareness at the current state of the art of computing.
- Application of critical thinking during challenging computer and software based problems.
- Design and implementation of a computing solution project which requires research.

- Integration of many different technologies in order to develop and deliver a working solution project.
- Being critical in regards to the work and research.
- Reflection on the strengths, weaknesses and any future potential of the project.

1.7 Chapter Conclusion

This chapter has described what each chapter is about, it provided the project's source code and documentation link. The goal, timeline and difficulties have also been set out. Finally the learning outcomes of the module have been written out.

Chapter 2

Context

2.1 Overview

2.2 Chapter Conclusion

Chapter 3

Methodology

3.1 Overview

This methodology chapter will describe the various methodologies that have been used in developing this project. It goes over the Approach, Meetings the Source Control and the Tools that have also been used. Additionally it will go over some Research methods that were taken into account before making decisions.

3.2 Research Methodology

There are different approaches to research. These approaches enhance the process and proves the way the research has been conducted [5].

The Research Methodology used for this project was the Qualitative Research Methodology.

Qualitative Research relies on information that is text based. This research is done from first-hand. Information can be obtained through observations, documents and artifacts [6].

This type of Research Methodology was a good choice for this project, as it allowed the author to research information through documentations and articles in order to be able to fix issues or to know more on the field of technology, such as a programming language or framework.

3.3 Software Methodology

Software Methodologies allow us to divide up the development process into smaller steps which in the end can help with design, development and management of the project [7].

This project has been developed using the **Rapid Application Development (RAD)** or sometimes also referred to as **Rapid Application Building (RAB)**.

This Software Methodology suits very well with this type of project as it is developed by only one person, it allows for developing many features and then deciding which of the few options to keep [8]. This way the development of the application or project is simpler as decisions can be made at nearly any time.

RAD is made up of many phases. These include the following [8]:

- **Requirements Planning Phase** - Which includes the scope of the project, requirements and design.
- **User Design Phase** - This phase goes through how users would like to see the project. What kind of inputs, outputs, design they would like to see.
- **Construction Phase** - Building of the project. But as this is RAD, any changes or improvements can still happen. This phase also includes testing of the project or application.
- **Cutover Phase** - Final phase. It includes final testing and user training in some situations.

3.4 Meetings

The project meetings were scheduled weekly. They were attended by both the supervisor and myself. They were all carried on-line as it was more comfortable. The meetings were under fifteen minutes long.

3.4.1 Meetings at the Beginning

The early meetings were based on the planning and possible ideas for the project. The project timeline and the project design brief has been done during those meetings. These early stage meetings took a couple of weeks.

3.4.2 Regular Meetings

Regular meetings consisted of:

- Current progress of the project.
- The problems encountered.
- Feedback from the supervisor.
- Plans for the following week.
- Changes in the project brief or weekly project timeline.

3.5 Source Control

The source control used during the development of this project was **Git** with the hosting provider **GitHub**.

3.5.1 Why was it picked?

This was the preferred source control as it is the most preferred and used source control in the industry. As well as all the student's project are based on GitHub which serve as the student's portfolio for the future.

3.6 Tools

In order for this project to be developed in the most convenient way. The most popular and best tools had to be used.

3.6.1 Visual Studio Code

Visual Studio Code has been used in development of this project as it a very popular, easy to use and widely modifiable piece of software, thanks to many developers working on extensions in order to improve it.

This Integrated Development Environment (IDE) has been used throughout the years of the Software Development course. Thanks to that, it made the ease of use better, as the student was already used to this tool.

3.6.2 Android Studio - Emulator

Since this project is aimed at Android devices. An emulator has been used to test the application in the quickest way possible.

The use of an emulator while developing an application is also convenient as there is no delay while using. The developer does not have to wait for the application to install on an android device etc. it is readily available straight away with no cables in the way. With the use of the same device that is used to code.

3.7 Chapter Conclusion

This Methodology chapter has went through the research methodology and software methodology that has been done in the process of making this project. It also discusses the meetings that have been conducted with the supervisor. Git and GitHub have been explained as the source control of this project, and finally the two main tools that were used by the student accordingly.

Chapter 4

Technology Review

4.1 Overview

In this chapter, the different types of technologies that have been used while developing this project will be discussed.

The Timetable Application has been developed, by using different technologies to work together. The following are the technologies that have been used to develop this project.

4.2 Git

Git is an open source distributed version control system. It is designed in such a way to be able to manage small and large projects with ease, while being fast and efficient [9].

Git is released under the GNU General Public License version 2.0, which means it is an open source project. Thanks to this license, all developers are free to share, change and use it for free [9].

This distributed version control system, is very easy to learn, fast and efficient, hence why it is so popular and used by many of the biggest companies in the world such as Google, Facebook, LinkedIn and Microsoft [9].

What are the Advantages?

One of Git's advantages is how distributed it is. Instead of using one repository, each of the developers has their own local repository and a history of

commits [10]. This makes it fast, scalable and easier to work with.

Git is one of the most popular version controls. Thanks to that, nearly all developers have had experience with Git and will not have to learn it from the beginning [10].

What are the Disadvantages?

Git is an excellent tool, but it also has some disadvantages.

One disadvantage of this piece of software is that it is slower on Windows machines [11]. Which could be a big downside for power users of Windows machines when using Git for development.

Git also does not support binary files [11], which could be important for some developers.

How was it used in this project?

In this project Git has been used as the version control system. It was used to connect to GitHub and push commits from the local repository to the hosted repository up in the cloud in the GitHub Servers.

4.3 GitHub

GitHub is a hosting service for version control, with the use of the above mentioned Git [12].

It offers source code, project management and many more useful collaboration features. [12].

What are the Advantages?

The Pages that it offers allows developers to showcase their projects and portfolios on a static page for free. They can be customised as the developers would like it with the paid themes [13].

GitHub can be accessed anywhere. Whether it is from a phone, windows PC or a Mac. Windows and Mac code sharing is simple and they can both do the same actions. While users can only access it on the phone only to view the repositories and projects [13].

What are the Disadvantages?

With Git having some disadvantages, and GitHub uses Git, this also shows that it does have some bad sides as well.

As any other service provider websites, GitHub also provides services, but not all are free. For users that do require a lot of repositories, teams etc. this could mean that they will be forced to buy a plan [14].

Since GitHub is a website. Any websites sometimes might encounter an error or update where the website will have to be put on a downtime for some time. This could create some problems and inconvenience when trying to push code up onto a repository online. Especially when working in a team.

How was it used in this project?

GitHub acted as the hosting provider of the project's repository. It worked with Git to host the repository with all the needed files, including Documentations, the Project and the Read Me.

4.4 Node.js

Node.js, is an open-source piece of software that works on many platforms. It has a backend JavaScript runtime environment, which runs on a so called V8 engine [15].

Node allows the creation of web servers and networking tools with the use of JavaScript. It has a collection of "module" that are responsible for various core functionalities [15].

The only language that Node supports natively is JavaScript as the name suggests .js. But there are many languages that can compile to JavaScript, some of these include Dart and TypeScript [15].

What are the Advantages?

Node.js is open source. This means that it is completely free. All modules, libraries and code samples are free to use [16].

Since it is open source. Developers try to help each other out. It is community driven. There are many developers who contribute to Node.js, and they are always willing to help out or to share their knowledge with somebody

else [16].

What are the Disadvantages?
Too many modules. There is a lot of unnecessary modules, which could over complicate things [16] [17].

Most JavaScript developers do not have a lot of experience with Node.js. Even experienced Software Developers tend to find themselves struggling with it sometimes [17].

How was it used in this project?
Node.js was downloaded specifically for this project in order to get the Node Package Manager. As it is essential for this project to be built and ran as expected.

4.5 Node Package Manager (NPM)

Node Package Manager or NPM is a package manager for Node.js. It is an open source project. It's goal is to help JavaScript developers share packaged modules of code [18].

The registry is a collection of packages which are open source and readily available to everyone to use for front-end, mobile applications and more. Developers can easily install any of the packages as well as publish packages [18].

What are the Advantages?
As mentioned above, the Node Package Manager is an open source project. Which is always a great thing when it comes to software development. It will most likely get updates for a very long time as individual developers will always try to improve on something.

What are the Disadvantages?
NPM might be the preferred and most popular package manager that is used currently, but it definitely has its disadvantages as well. NPM can sometimes take its time to install or reinstall a package, when compared to other package managers. This shows that there is definitely a problem with the performance of this package manager [19].

How was it used in this project?

The Node Package Manager was a very important technology in this project. With the help of npm, the other technologies were possible to be installed as packages and ran for the project to be hosted as needed for development and testing.

4.6 React Native

React Native is a JavaScript based framework. Facebook has released React Native in 2015 and it is still maintaining it. React Native combines native development with React, hence the name React Native. React Native code is written in JavaScript but it is then rendered with native code for a given device [20].

What are the Advantages?

Most important feature of this framework is the ability to develop cross-platform application with one code [20]. Which then transforms it into native code for other applications like Android and iOS. This feature is a great advantage, which decreases the cost and time it takes to develop applications. It is well suited for start-ups, as it does not require separate developers to develop working versions for iOS and Android.

React Native also offers a great feature called ”Fast Refresh”, this feature allows for as the name suggests a fast refresh while developing an application. The application refreshes on saving a file, without the need of restarting the application [20].

What are the Disadvantages?

One major disadvantage that have been encountered in this project is the lack of information in the error messages. There have been times where the error messages seemed the same while debugging, therefore it was very difficult to debug the application. There was no specific ”hint” as to what causes the problem in the code.

Another disadvantage that could cause a bigger issue for some users is the loading time of the application. While developing this application. React Native takes a longer amount of time to start than if it was developed natively through Android Studio for example.

How was it used in this project?

In this project React Native was used to build the Front-End of the application, as well as to connect with other components in order to have the project working as planned.

4.7 Firebase

Firebase was founded in 2011. From 2014 it has been acquired by Google. It is rapidly gaining respect and fame amongst developers. Large multinational companies are trusting Firebase services.

This back-end service provider offers many services such as, databases, authentication, storage and hosting. With that it also offers analytics to see how the application is evolving. Firebase also offers many paid plans as well as a limited free plan.[21].

What are the Advantages?

One of the advantages that Firebase offers is that it is very easy to integrate on iOS, Android and Web [21].

Firebase offers many services, all in one package. They offer databases, analytics, storage and hosting. This is a great service to consider, as it allows for quick scaling and having many services in one offer great integration.

The documentation is well written and very helpful. They offer an example code snippet to each feature of their service in the documentation. This does not mean that it always help, as the code may differ a lot, depending on how the source code of a project is written [21].

What are the Disadvantages?

iOS devices do not seem to be favoured by Firebase. Which makes sense as Android is sponsored and focused by Google more than other systems. This is a disadvantage as Firebase claims it is a cross-platform service [22].

Firebase has a major limitation, which could be a large factor to consider when deciding if this service is right for a project. This limitation is availability. Firebase will not work in countries where Google is banned or not allowed [22].

There are two available plans to pick from when choosing this service. The

"Spark Plan" is completely free, but has it's limitations, it is great for very small applications, it also depends which services the applications will use, and it is great for development and testing. There is also the "Blaze Plan" which is a Pay As You Go plan. This is not necessarily a good idea, as we can never be sure how viral the application will be in the next month or so. This could increase the costs drastically without making enough profits from the application itself [22].

How was it used in this project?

Firebase managed all the back-end of the project. The database and the authentication features of this service were used, to store the user's information as well as the timetable information.

4.8 Android Studio

Android Studio is a great piece of technology which provides tools for building applications for any Android Device [23]. It also offers many emulators which are based on real devices such as Google Pixel phones and different versions of Android such as KitKat, Oreo etc.

What are the Advantages?

Android Studio has a built in editor. It is called the Visual Layout Editor, it allows for quick and easy manipulation of GUI elements for the application, without touching the code [23].

One of the most important advantages of it is the emulator. There are multiple options of devices and versions of android to pick from. This emulator is fast, it is quick and very simple to install. It has complete functionality of a standard phone. There is access to the camera, volumes, tilting of the screen and many others [23].

The Integrated Development Environment (IDE) inside of Android Studio is based on the IntelliJ IDE. Which is one of the most popular IDE's. Additionally Android Studio carries some other functionality which enhances the productivity [24].

What are the Disadvantages?

Emulators are big in size. They take a lot of memory and storage. This could be an issue on devices used for development with smaller storage capacity

like some laptops. Certain laptops or PCs can get warm when running an emulator in the background, as they use a lot of memory to run.

How was it used in this project?

The only way this piece of software was used in this project was the Android emulator. Which helped to develop, test, visualise and connect everything together.

4.9 Expo

"Expo is a framework and a platform for universal React applications". It is a great collection of tools and services which can be used with React Native in order to develop, build and deploy cross-platform applications using only one language which is JavaScript or TypeScript [25].

What are the Advantages?

The Metro Bundler or the Expo Developer Tools, makes the whole development process a whole lot easier and faster. With just one click of a button or option picked in the development console or in the web browser, the developer is able to run the application on their iOS or Android emulator as well as a website.

Thanks to Expo-Cli the creation of a project is much quicker and it provides templates to use from, to give developers a little bit of a boost at the start if they decide to use one of the templates [26].

What are the Disadvantages?

Expo App sometimes disconnects with the Expo Developer Tools app or the Metro Bundler. It can be frustrating and make the developer loose concentration while developing.

How was it used in this project?

Thanks to Expo, this project was hosted and was able to connect with the emulator. Without Expo, this project would not be possible. The Expo-CLI was also used to initialise the project in the beginning.

4.10 LaTeX

LaTex is a typesetting system. It is commonly most used for scientific and technical documentation. It is a free piece of software [27].

What are the Advantages?

One of the huge advantages of LaTeX is efficiency. With the use of table of contents, or bibliography or references and table of figures etc. the authors do not have to insert anything or refresh it manually. LaTeX adds everything to it automatically after it gets compiled.

What are the Disadvantages?

The appearance of LaTeX may not suit everyone. It still allows the writers to have some customisation in their documents through the use of packages, but in the end the document will look plain as it was built for scientific and technical documentation which do not need a lot of customisation and pretty looks.

Writing in LaTeX is completely different than in other text editors or text processing programs like the very popular **Word**. In LaTeX we need to use commands, packages etc. It is not as simple as a drag and drop to insert an image to a document.

How was it used in this project?

Without LaTeX this dissertation would not be as comfortable to write as it was. This tool was first used by the students of this course at the beginning of fourth year. It is always great to learn new technologies which could make the students switch from the traditional text processing softwares to LaTeX.

4.11 Chapter Conclusion

The Technology Review chapter provided descriptions of the technologies used in this project, with the most important advantages and disadvantages of them. This project would not have been possible without the use of these technologies.

Chapter 5

System Design

5.1 Overview

This chapter goes over the system design for the project. Since React Native allows for developing many applications with one code, this chapter will also go over the two types of applications that have been developed.

5.2 Android Application

The Android application was the main focus of this project. As the targeted devices of this project are Android devices.

Why Android Devices?

The reasons why Android devices have been picked as the main target of this application are as follows:

- Android devices are the most popular devices used currently worldwide. They take over 70% of global market share, while over 20% of the global market share are iOS devices [28].
- It is easier to develop Android applications than iOS applications.
- With the picked technologies for this project, there are more limitations for iOS devices than the Android devices.

5.2.1 Login and Register

At first, once the user starts up the timetable application. The Login and Register screen is displayed, where the user is presented with two input boxes and three buttons. There is an email and password text input. Then followed by the two main buttons, Login and Register and a text button which is responsible for recovering a forgotten password.

Login Functionality

When a user decides to login. They are required to input their email and password. They can then press on the Login button.

After pressing the login button the application checks with the authentication system from firebase, if the email and password that the user has entered match.

If they do, the user is then authenticated and will be transferred to the Home Page.

If the details entered by the user do not match, then an alert box will be displayed on the screen, to inform the user about the error.

Register Functionality

In order to register in the app. The user has to give their email and password in the input boxes provided.

Next up the user can click on the Register button, which will prompt firebase authentication system to register a new user with the given email and password.

On a successful register, the user details are saved in the firebase authentication system and then the user is moved to the home page of the application.

The system also checks if the email is already taken and if the password is too short. This will prevent the user from registering and prompt an alert box to let the user know of the situation.

Forgotten Password Functionality

When the user finds themselves in a situation where they cannot remember their password for their account. They can use the Forgot Password feature in the application.

The forgot password buttons can be found on the Login and Register screen of the application and the Settings screen. The reason for it to be on the Login and Register screen is that if it was not there the user's account would be completely lost when the password has been forgotten.

The forgot password button is also on the settings screen. There are two reasons for this:

- The application uses asynchronous storage and therefore keeps the user logged in. They might forget their password after some time.
- Or the user might want to change their password every so often, because of security purposes and they might end up with the password being forgotten as they did not have to log in often, thanks to the application keeping the user logged in.

This asynchronous storage feature has it's advantages and disadvantages. But thanks to that it should increase user satisfaction, as they do not have to waste a few seconds to log in every time they want to access their timetable, which most likely will be looked at many times a day.

5.2.2 Home Page

Once the login or register process has been passed through successfully. The user is then transported to the home page screen. On this screen the user is greeted with information about the application and a brief tutorial on how to begin using the application.

5.2.3 Timetable

When a user clicks or taps on the Timetable icon from the tabs menu. They will be moved to the timetable screen where they will be able to add new timetable using the icon on the bottom right hand corner, which looks like a plus sign. They will also be able to see their timetable for the week.

5.2.4 Settings

Once the settings icon has been pressed on the tabs menu at the bottom of the screen, the setting screen will be displayed in front of the user. This screen displays some important functions of the application, which any application should consider having when an email authentication has been added to it.

Dark Mode

Change Password

The Change Password screen is responsible for changing the password whenever the user sees it as a necessity.

The user is asked to enter their email, current password and the new password. With this information, the application is able to verify if the email and current password are valid and match the account details. Therefore the password can be changed in the Firebase Authentication service.

```
<View style={styles.inputContainer}>
  <TextInput
    placeholder="Enter Your Email"
    value={email}
    onChangeText={(text) => setEmail(text)}
    style={styles.input}
  />
</View>
```

Figure 5.1: Forgot Password Text Input from Project Source Code.

The above figure describes the code structure for each of the inputs in this given screen.

There was one issue that was faced when this feature was developed. The changing of password was not allowed, after some time. As the user gets deauthenticated after some time and therefore will not be allowed to change the password. Therefore firstly the application reauthenticates the user with

firebase. That is why the current password is needed in this case, as the application passes on the password to reauthenticate and then changes the password.

Change Email

The screen to change the user's email, is responsible for allowing the user to change their email. It takes the very same approach as the previous change password screen. It firstly needs to authenticate the user once again, before proceeding to changing the email.

The design of this screen looks very much the same, except there is one less input box. The input boxes on this screen are for inputting the new email and for the current password. The current password is there to confirm the user's choice and to reauthenticate the user with the firebase authentication service.

Forgot Password

Delete Account

Log Out

The log out button is situated on the settings screen. Once the log out button is pressed the application logs the user out. The user of the application is then navigated back to the login page. Where they can log in again, with a different account or register with a new one.

5.2.5 Account Management

All of the user accounts are managed through Firebase. Each user gets a unique identification number (UID) through which they can be identified. This UID can also be used for other services, like the databases which will match individual information for each user.

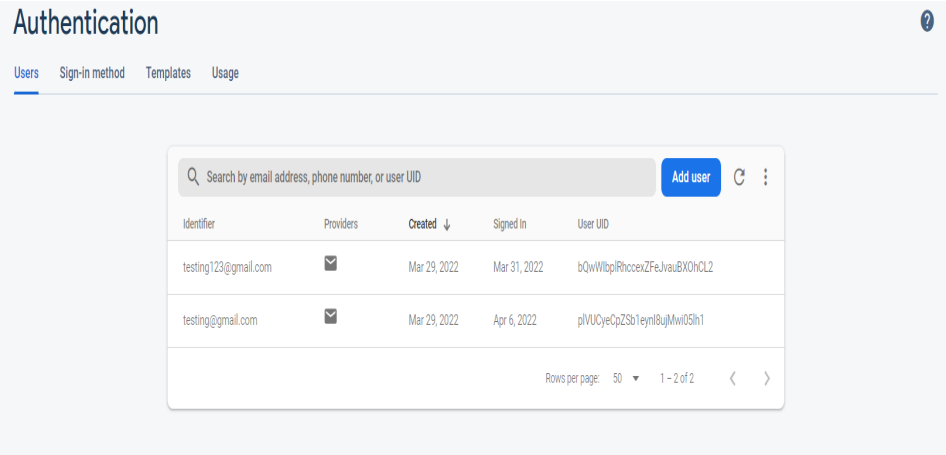

Figure 5.2: Firebase Authentication Project Page.

As shown on the figure above, the administrator of the firebase project can search for a user, delete a user and also add a new user. They are also able to see when the user has created their account and when was their last login.

The administrators can see what kind of account the user has created. In this case the accounts have been created with an email and password.

5.2.6 Database Management

Firebase provides two types of Databases. The firestore database and the realtime database. The database that has been picked for this project is the realtime database.

This database has been picked, because receiving information instantly on any change is an important factor for this type of application. The data has to be returned instantly.

A timetable application is a type of application, where a user most likely will check on many times a day. A refresh button on the screen would not be as good for user experience as a realtime database which would update whenever there is a change detected.

5.2.7 In-App Navigation

React Native provides many types of navigation. All with their pros and cons, different animations and functions. React Native offers the ability to

nest navigations on top of one another. This is especially used when the application is made in such a way, that there are drawers which have buttons which then move the user to a different screen etc.

In this project the nested Stack and Tab navigation have been used. Both of them play an important role in the application and are responsible for different screens and have connections in between them.

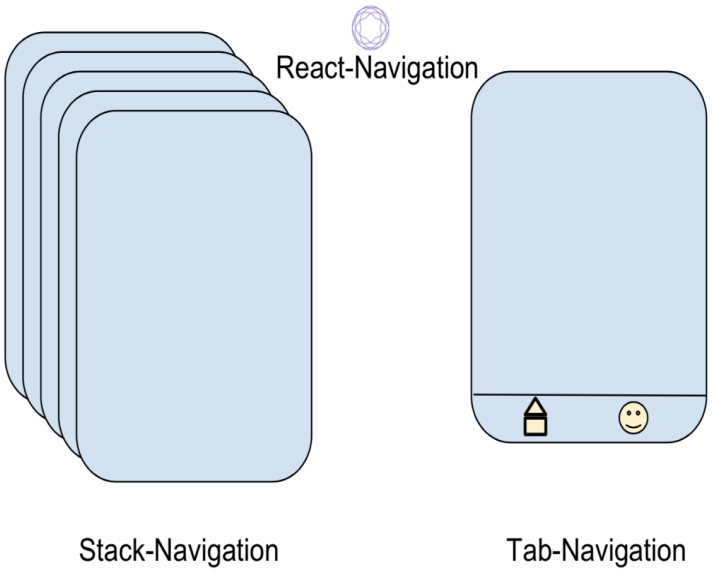

Figure 5.3: Tab and Stack navigation. Adapted from [1].

The Tab navigation is used for three main screens of the applications which are:

- Home Page
- Timetable
- Settings

These three screens can be picked from the tab navigator at the bottom of the screen. This tab navigator is available only on those three screens.

```
<Tab.Screen
  name="Timetable"
  component={TimetableScreen}
  options={{
    tabBarIcon: ({ focused }) => (
      <View
        style={{
          alignItems: "center",
          justifyContent: "center",
          marginTop: 0,
        }}
      >
        <Image
          source={require("./assets/icons/timetable.png")}
          resizeMode="contain"
          style={{
            width: 28,
            height: 28,
            tintColor: focused ? "#2b7eba" : "grey",
          }}
        />
        <Text
          style={{ color: focused ? "#2b7eba" : "grey", fontSize: 14 }}
        >
          Timetable
        </Text>
      </View>
    ),
  }}
/>
```

Figure 5.4: Tabs navigation in the project.

As shown on the figure above, that is how each tab is implemented in the code. Each tab has it's own icon, which is an image, but at a smaller size. All tabs have the same colours. They change colour when they are the active screen.

The Stack navigation is used for all other screens in the application. This includes the Login and Register, all the options from the settings screen.

```
<Stack.Navigator>
  <Stack.Screen
    name="Login"
    options={{ headerShown: false }}
    component={LoginScreen}
  />
  <Stack.Screen
    name="HomeTabs"
    options={{ headerShown: false }}
    component={Tabs}
  />
  <Stack.Screen
    name="ChangePassword"
    options={{ headerShown: false }}
    component={ChangePasswordScreen}
  />
  <Stack.Screen
    name="ForgotPassword"
    options={{ headerShown: false }}
    component={ForgotPasswordScreen}
  />
  <Stack.Screen
    name="ChangeEmail"
    options={{ headerShown: false }}
    component={ChangeEmailScreen}
  />
  <Stack.Screen
    name="DeleteAccount"
    options={{ headerShown: false }}
    component={DeleteAccountScreen}
  />
  <Stack.Screen
    name="AddTimetable"
    options={{ headerShown: false }}
    component={AddTimetableScreen}
  />
</Stack.Navigator>
```

Figure 5.5: Stack navigation in the project.

As it is shown in the figure just above. The stack navigation is a very simple type of navigation compared to the tab navigation. Every screen of the application except the main three is a stack screen. Each stack screen in this project has their header disabled. This header shows a title of the screen on each screen if it is enabled. It has been disabled in this project as it does not look appealing.

These two types of navigation are connected together, by firstly loading the Login screen. Once a user is logged in, the stack navigation is calling the Tabs component, which actually calls a function called tabs. This functions opens the home screen with the tabs menu at the bottom of the screen. From there, those screens from the tab menu have different screens that can be directed to. For example on the Settings screen from the tab navigator, the user can switch to change email screen which is part of the stack navigation. This is the same logic as on the figure below.

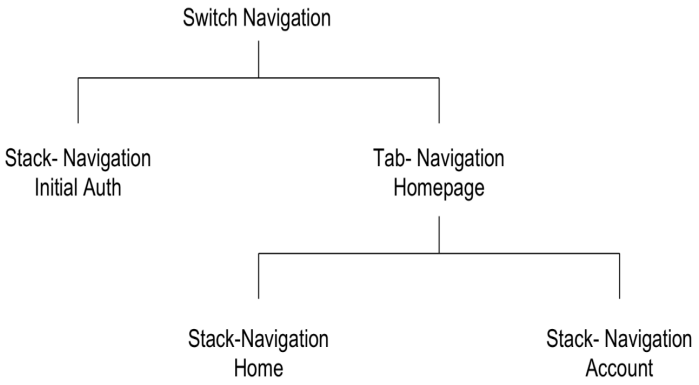

Figure 5.6: Connected Tab and Stack Navigation. Adapted from [1].

5.2.8 Two-Factor Authentication

5.2.9 Types of Views

5.3 Web Application

Thanks to the fact that this application is built as a cross-platform, and with the use of Expo and React Native, the Web Application was also possible to build at the same time.

The Web Application of this project looked completely the same as the Android version of this project. It had some minor differences in the feedback animations when buttons were pressed, as well as the style of the application looked more stretched as the project was mainly aimed at Android devices, the web application was not taken into consideration. Nonetheless it was still looked at from time to time in order to test the functionality.

5.4 Chapter Conclusion

This chapter has went over the two types of applications of this project. With the main focused application being the Android application. It described the many different features of the application, such as the navigation, the screens and the database.

Chapter 6

System Evaluation

6.1 Overview

This chapter will discuss the types of testings that were involved while developing the application. This way it will analyse various aspects of the project.

6.2 Graphical User Interface Testing

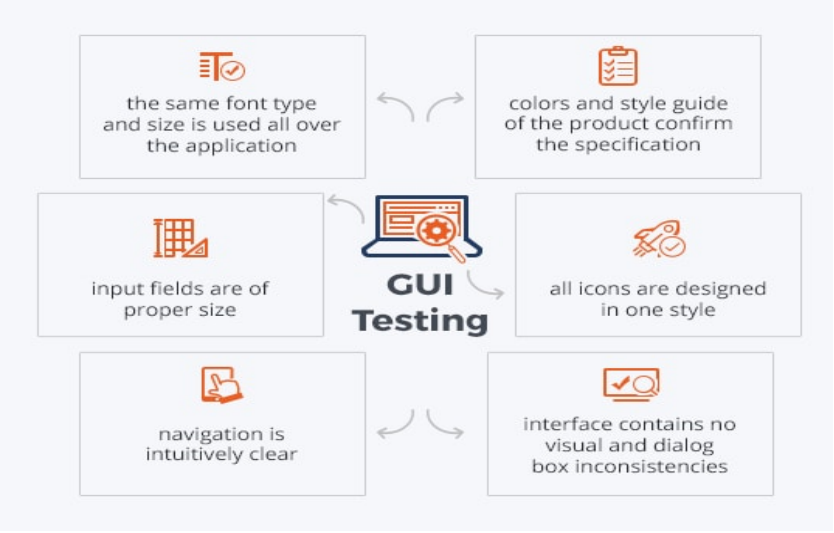

Figure 6.1: Example of GUI Testing. Adapted from [2].

Graphical User Interface Testing tests a piece of software with a Graphical User Interface (GUI). The piece of software that is being tested can have buttons, areas of text input, menus, text etc [29]. These tests can be performed using certain software tools or by developers and designers opinion. It depends what is being developed.

In this project I have used Graphical User Interface Testing to test all of the components at the Front-End of my project. I have tested to see if the buttons are of the right size and at the correct position. I had to make sure that all Front-End components match the style of my application and do not overlap each other. I have completed this type of testing by going using my opinion on the elements of the Front-End.

6.3 End to End Testing

Figure 6.2: Example of End to End Testing. Adapted from [3].

End to End Testing is a very important part of testing. It is used in order to test the functionality and performance of a piece of software . The idea of this test is to pretend what a real user experience would look like while using the application [30].

In my project I have put myself into the mindset of a user and tested the application as I would like it to work as a user of the application. During

testing, I have checked if I was given the right feedback to my actions. I have tested if I could retrieve the right information as well as send the right information to the other systems of the projects which for example were the Authentication and the Real-time Database.

6.4 Functional Testing

Figure 6.3: Example of Functional Testing. Adapted from [4].

Functional Testing is a type of black-box testing. It tests the functions of a software project, by giving the software inputs and testing the outputs. The tests are strictly focused on the functionality and not the software used to develop the piece of software, hence it is a type of black-box testing [31].

I have used Functional Testing in this project by testing each function one by one. This was a lengthy process, but thanks to that I was able to see what I have done wrong or not yet implemented so that a specific function could work as expected.

6.5 Chapter Conclusion

Chapter 7

Conclusion

7.1 Overview

This is the last chapter of this dissertation it will conclude all the other chapters. Discuss the objectives and analysis of the project and also mention any downfalls or problems encountered as well as the improvements that could be made in the future.

7.2 Context and Objectives

7.3 Evaluation

7.3.1 Improvements

7.3.2 Downfalls

7.3.3 Overall

7.4 Chapter Conclusion

Bibliography

[1] P. Doshi, “React navigation: Stack-navigation with tab-navigation.”

[2] U-Tor, “Gui testing: What, why, how?.”

[3] A. R. Chowdhury, “All you need to know about end to end testing.”

[4] C. Technologies, “Functional testing.”

[5] Wikipedia, “Methodology.”

[6] Wikipedia, “Qualitative research.”

[7] Wikipedia, “Software development process.”

[8] Wikipedia, “Rapid application development.”

[9] Git, “Git’s official website.”

[10] Atlassian, “Why git for your organization.”

[11] P. Pedamkar, “Introduction to git.”

[12] Wikipedia, “Github.”

[13] CodeInstitute, “What is github?.”

[14] J. Clancy, “The advantages and disadvantages of using github.”

[15] Wikipedia, “Node.js.”

[16] O. Romanyuk, “Node.js is a great runtime environment - and here’s why you should use it.”

[17] AltexSoft, “What is node.js and its pros and cons.”

[18] NPM, “Npm official website.”

[19] A. Opidi, "Npm vs. yarn: Which package manager should you choose?."

[20] R. Native, "React native's official website."

[21] Firebase, "Firebase's official website."

[22] Moqod, "Downsides of firebase: limitations to be aware of."

[23] A. Studio, "Android studio."

[24] A. Studio, "Meet android studio."

[25] Expo, "Introduction to expo."

[26] Expo, "Expo cli."

[27] LaTeX, "Latex's official website."

[28] J. Wallen, "Why is android more popular globally, while ios rules the us?."

[29] I. Banerjee, B. Nguyen, V. Garousi, and A. Memon, "Graphical user interface (gui) testing: Systematic mapping and repository," *Information and Software Technology*, vol. 55, no. 10, pp. 1679–1694, 2013.

[30] SmartBear, "Combine api and ui testing for confidence at every layer of your application."

[31] Wikipedia, "Functional testing."