"Rememberry: A memory education and rehabilitation app"

Nyasha Gumbo
15726701

Final Year Project – **2019**
B.Sc. Single/Double Honours in
Computer Science/Computer Science and Software Engineering

Department of Computer Science

Maynooth University

Maynooth, Co. Kildare

Ireland

A thesis submitted in partial fulfilment of the requirements for the B.Sc. Single/Double Honours in Computer Science/Computer Science and Software Engineering.

Supervisor: **Joseph Duffin**.

Contents

Declaration

I hereby certify that this material, which I now submit for assessment on the program of study as part of the award of M.Sc. in Computer Science (Software Engineering) qualification, is *entirely* my own work and has not been taken from the work of others - save and to the extent that such work has been cited and acknowledged within the text of my work.

I hereby acknowledge and accept that this thesis may be distributed to future final year students, as an example of the standard expected of final year projects.

Signed: Nyasha Gumbo Date: 28/03/2019

Acknowledgements

I would like to thank my supervisor, Joseph Duffin, for all the help and support he provided over the duration of this project.

I would like to thank Peter Mooney and the team of demonstrators, who lead CS385 in NUIM, for allowing me to take this module and assisting me where needed.

I would like to thank Senna Howley, who branded and designed all Rememberry logos and symbols used for the project.

I would like to thank Dr. Sean Commins and Dr. Richard Roche who showed a genuine interest for the project and provided insight into what the industry is looking for.

Abstract

The goal of this thesis is to investigate the viability of using mobile application technology as a means of providing a memory intervention course to users who suffer with age related memory problems and dementia.

The topics covered in the thesis include the approach and preparation taken, research done on the topic material, an analysis of the problem and a breakdown of the solution and an evaluation of the effectiveness of the solution.

The main conclusion that was arrived at was that the technology used is fully capable of providing the memory intervention service described and only further testing in a clinical setting will be able to show how effective the application can be.

List of Figures

List of Tables

Chapter one: Introduction

The purpose of this chapter is to present the reader with a high level overview of the project. It is the aim to provide a brief description of different aspects of the project within this chapter, before covering the topics mentioned in greater depth in later chapters.

1.1 Topic addressed in this project

The basis of this project stems from numerous memory intervention courses that are conducted on patients suffering from age related memory problems and dementia. The courses are currently run by occupational therapists or nurses over 6 weeks with each session lasting 1-2 hours. During these courses patients are educated on the human mind and memory. Awareness is also raised on the effects age can have on memory. Exercises are done to help with memory skills and techniques and strategies are discussed to help them cope with memory problems (Gross AL1, 2012).

The aim of this current research and development is of a neuro-education and memory rehabilitation application based loosely on the courses described. This application will be targeted mainly at people suffering from cognitive decline and their carers/family. The aim is to implement an inexpensive, easily-accessible intervention in the form of a neuro-education application for people dealing with different stages of cognitive decline, in order to provide them with an improved understanding of what is happening to them on a neuroanatomical, physical and cognitive level as well as aiding them with the help of both internal and external techniques that have scientific backing behind them (Cahill, Macijauskiene, Nygård, Faulkner, & Hagen, 2008).

1.2 Motivation

With the constant improvements in health care and medicine, the average human lifespan is increasing and with that the need for effective treatment for reducing cognitive decline in an aging population is required more than ever. Subjects suffering from diseases such as Alzheimer's and dementia lose efficiency in cognitive areas such as long and short term memory, prospective memory ,immediate memory and general attention span. This mobile application will help to improve or assist in these areas. In the field of memory rehabilitation there is a distinct lack of effective pharmacological treatments for these cognitive impairments. Cholinesterase inhibitors such as donepezil and rivastigmine are only proven to show time limited benefits for individuals with Alzheimer's disease (George Savulich, 2017). By creating an accessible mobile application it could help improve memory functions and general life of people suffering from these diseases as well as their family members and carers.

While the memory intervention course has been proven a general success by studies(Martin, 2016) the course also comes with some constraints and if these constraints could be lifted the course would be able to help a lot more people than it currently is. These constraints are mainly centred on the logistics of attending a physical face-to-face intervention course such as time-related, accessibility related and personnel related problems. Does a certain time suit everyone who wishes to attend the course? Can everyone make their way to the location the course is being held at? Are there enough occupational therapists or nurses available to make sure everyone is attended too? The motivation behind this application is to try our best to eradicate these types of problems.

1.3 Problem statement

The technical problem faced with this project was designing an android based mobile application that is easy to use and easily accessible that carried functionalities and utilities based mainly on the memory intervention course mentioned above along with added functionalities to accompany this.

The app will contain interactive displays that educate the users on the fundamentals of memory. The application will be constructed with built in support to the patient on how to deal with the issues they are encountering with their memory as well as multiple surveys to determine the performance of any individuals memory performance in multiple areas. This data will be stored by the app and used to help track the progress of the users.

It will also carry other utilities such as a built in notes utility to record and keep any notes and a calendar to assist the users with remembering events and details.

1.4 Approach and Preparation

I began my approach to designing this application by first conducting research in the market on applications that are out there. I assessed 5 applications in the general domain of memory rehabilitation and noted all the things they did well, as well as what they lacked.

Survey of Apps designed to support and improve memory and Cognitive ability	
Mind Mate	Pros: Easy to use (Good UI), Contains videos, nutritional tips etc, games adjust based on proficiency Cons: Only 3 games directly focused on memory, doesn't track progress
Lumosity	Pros: Good UI, records in depth statistics compares you against others, wide expanse of games, starts with baseline tests. Cons: Subscription for premium version is $15 a month.
MemBrain	Cons: UI is quite basic, not free, not a lot of memory based games, unable to choose specific game.
CogniFIt	Pros: Sectors based for different people eg.ADHD,Depression etc, Good UI, Tracks results Cons: Games not scientifically tailored, full version not free,
Elevate	Pros: Starts with quick quiz to tailor to your needs and skill level, Good UI,tracks progress, Cons: Full version not free, no memory section

Most of these apps were very helpful with interesting brain exercises and justified tracking of progress of the users however, what they all seemed to lack was any education on understanding memory and how memory problems can begin to occur. As this is one of the main facets covered in the memory intervention course I wanted to make sure this was integrated into the app.

In terms of content I had a solid idea of what information and utilities I wanted the application to have. However I was less certain on the design of the user interface and general flow of the app. The two pillars I centred my design ethos on where "simplicity" and "ease of use". To get a rough idea of the layout of my app I started by designing some wireframe frameworks. I did this by using Balsamiq mock-ups. I designed a home screen and the general navigation and flow of the app. This gave me a good base to start developing my application from.

I decided to use the ionic framework to develop my hybrid mobile application. I found out that ionic is built on top of the Angular framework from Google and aims at reducing manual labour for you. It can save you a significant percentage of your overall development time and help you get your product out in the market very fast. As I mentioned earlier time was a very precious resource this was one of the main reasons that the ionic framework was chosen. The open source nature of the framework ensures you will always be able to find and use third-party solutions when developing most of the common functionalities for your app. You can even extend the functionality of those pre-existing solutions quickly. Another reason this framework was chosen was the fact I was able to enrol myself in a module being taught at my university NUIM, "Mobile Application Development", which covered the development of hybrid applications using ionic framework. This was a big motivating factor for choosing ionic as I had a resource I could go to with any questions or queries. This course took a modern approach to mobile app development by working with angular and ionic to develop hybrid cross platform applications. Essentially this means using HTML5 and JavaScript to create mobile first web applications which are then converted into native apps using Apache Cordova. I attended the course for 8 weeks attending both lectures and labs, where lab demonstrators offered assistance, and this was hugely beneficial in my understanding of application development.

Despite how convenient hybrid apps are their performance levels are significantly lower than that of native apps which are perfectly compatible with the host system's architecture and components, thus making sure the performance always remains above the bar. I took this on board while deciding the route I would take but still felt the advantages the ionic hybrid framework offered outweighed its slightly lower performance levels.

I also decided to use Googles firebase to manage my users and user information. One of the best features of Firebase is the Analytics dashboard that it comes equipped with. It is free and has the capacity to report 500 event types, each with up to 25 attributes. The dashboard is optimised to view user behaviour and measure various user attributions. Ultimately it helps you understand how people use your app so you can better optimize it in the future. As user tracking will be a large part of this project I decided firebase was the best option. Firebase is also quite scalable which was a deciding factor also.

1.5 Metrics

In terms of evaluating my work I plan to allow a controlled number of my piers to use the application for a determined amount of time (5-7) days. I will then ask them to rate the application under certain categories such as ease of use, how useful they found it, how stimulating it was etc. I feel like this would give me a good idea of how successful the development of my application was. A more comprehensive and scientifically based metric of the application would have been more suitable but due to time constraints and difficulty in finding suitable test subjects we could not go in this direction just yet. An example of such a metric can be found in George Savulich’s paper where a randomized controlled trial was carried out on 42 patients with a diagnosis of amnestic mild cognitive impairment to determine the effectiveness of a mobile memory game. Half the patients played the game (8 hours a week) over 4 weeks with the other half assigned to the control, in this case clinic visits as usual (George Savulich, 2017). A comprehensive study like this would have been beneficial in determining the effectiveness of the app and may still be carried out at a further date.

Chapter two: Technical Background

The aim of this section is to present the reader with an overview which will explain some of the existing issues and research topics related to the project at hand - the design and creation of an application that will be used to assist people suffering with memory issues. We will introduce the reader to the problem domain - human memory, memory rehabilitation techniques, software usability issues - related to the developed application. We will also provide summaries of the attempts of other authors to provide assistance to people with disabilities through the use of software and modern technology.

2.1 Topic material

2.1.1 What is memory?

Memory is the process of storing and retrieving information and experiences. Information received makes its way into our memory through our senses. The formation of memories begins with our senses, if our senses such as smell, sight and sound are not functioning properly the task of forming accurate memories becomes difficult. Memories are processed by several systems throughout the brain and kept for use at a later stage (Mason & Kohn, 2001).

Memory can be separated into multiple unique sub-categories where each one has its own specific functionality. All these subcategories link up to help the memory function appropriately. It can generally be broken down into sensory, short-term and long-term memory. When a memory is first generated it is immediately stored in in the short-term memory and then based on its use eventually stored in the long-term memory. The memory acts as a sort of filter analysing snippets of information and past experiences to determine if it is useful or not.

2.1.2 Types of memory

Short-term memory or working memory is capable of keeping the information in our brain for short periods of time and using this current information for the task at hand. Short-term memory is supported by the regions of the brain called frontal and parietals lobes.

In terms of sensory memory the input received from the sense organs such as eyes, ears etc. remains in the sensory memory for a very short time like a matter of seconds. For example, the sound from a radio may appear to be in your ears for a fraction of time even after the radio has been turned off or the image of a TV may linger in our eyes even after the television has been turned off (Baddeley, 2009).

Long-term memory refers to the continuous storage of information taken in. Usually this information is far detached from us but can be summoned into the short-term memory to be utilised when needed. While some of this information is easy to recall some long-term memories are much more difficult to access. This ability to recall memories obviously differs from person to person (Terry, 2018).

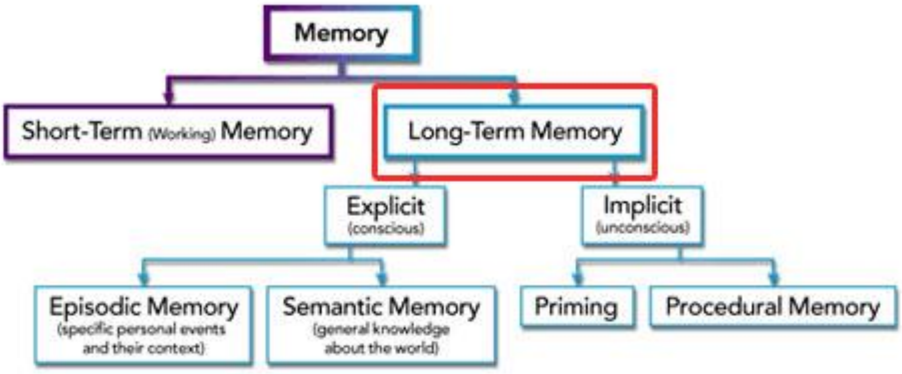

Memory
Short-Term (Working) Memory
Long-Term Memory
Explicit (conscious)
Implicit (unconscious)
Episodic Memory (specific personal events and their context)
Semantic Memory (general knowledge about the world)
Priming
Procedural Memory

Figure 1.1 Breakdown of the human memory

2.1.3 Issues with memory

Mild cognitive impairment represents a more severe form of memory loss and is the stage between the expected cognitive decline of normal aging and the more serious decline of dementia. It can involve problems with memory, language, thinking and judgment that are greater than normal age-related changes. Although patients with mild cognitive impairment are able to continue to live independently, they show objective memory impairments similar to those seen in people with early stage Alzheimer's disease (Selam Negash, 2008). About 10% of people aged 65 years or older have mild cognitive impairment, and nearly 15% of them develop Alzheimer's disease each year.

As people live longer, the risk for developing Alzheimer's disease increases dramatically. Although it is the most common cause of late life dementia, other causes, particularly vascular disease, contribute to the occurrence of dementia, often defined as impairment in several cognitive domains including memory to the extent that it interferes with daily life (Anon., 2018).

2.1.4 Evaluating memory

Evaluating memory is quite a complicated process as so many variables have to be taken into account. However there are a few standardised tests in the memory field that are widely accepted such as The Rivermeade Behavioural Memory Test and the Hopkins prospective and Verbal memory test (Cour, 1990).

2.1.5 Memory rehabilitation techniques

Cognition-focused interventions are interventions that directly or indirectly target cognitive functioning. These include interventions such as the memory intervention courses I have studied as opposed to interventions that focus primarily on behavioural, emotional, or physical function. Interventions of this type usually involve interacting with the subject in a number of general activities and discussions. They are commonly conducted in group scenarios and are focused on the general improvement of social and cognitive functioning. A recent study done that focused on cognitive-focused interventions of a similar type to that described above came to the conclusion that this general stimulation of the mind and real-life based situational approach consistently produced improvements in cognition and as a consequence improved well-being and quality of life (Woods B, 2012).

Cognitive training takes a more focused and analytical approach on memory rehabilitation and typically involves guided practice on a set of standardized tasks designed to show particular cognitive functions, such as memory, attention, or problem solving. Tasks may be presented in paper-and-pencil (P. QUAYHAGEN, 1995) or computerized (R. Mielke, 1996) form and may even involve analogues of activities of daily living (Zanetti, 2010). Tailoring of task difficulty on the basis of the individual performance level and adaptive training is becoming increasingly available through computerized packages (Owen AM, 2010). This means tests are becoming more flexible and easier to tailor to every subject's specific requirements. One assumption underlying cognitive training is that practice has the potential to improve or at least maintain functioning in the given domain. An additional assumption is that any effects of practice will generalize beyond the immediate training context.

2.2 Technical material

Ionic 2.2.1

The app would be built using the ionic framework. Ionic is built on AngularJs which is a Javascript framework that extends the vocabulary of HTML. The application created lies in the inner square of Ionic/Angular as the app is built with these two frameworks. NgCordova is used underneath to build and emulate the platform app. Apache Cordova is a set of APIs that assist in accessing the native functions of android and IOS platforms. This is all wrapped in node.js which must be installed initially (Mittal, 2017).

Firebase 2.2.2

Firebase is a mobile platform provided by Google. It provides numerous features to web and app developers. It is centred on cloud services and allows users to push and pull data from any device or browser. This is very useful for services our application would provide such as analytics and user authentication. Firebase handles the backend online component of the app. It is done through an SDK with easy to use API's. (Sinicki, 2017)

Chapter three: The Problem

The problem faced in this project is how to incorporate a memory intervention course into a mobile application which still provides all the support the original pen and paper intervention course offered. The application will have to be functional and simple to use.

3.1 Project UML documentation

As shown in the UML diagram below the application is targeted for use by both users and their helpers. The users will be able to learn about memory along with their helpers as well as partake in surveys and receive feedback from these surveys, use both the calendar and notes services provided and receive strategies and tips to help with any memory problems they are facing.

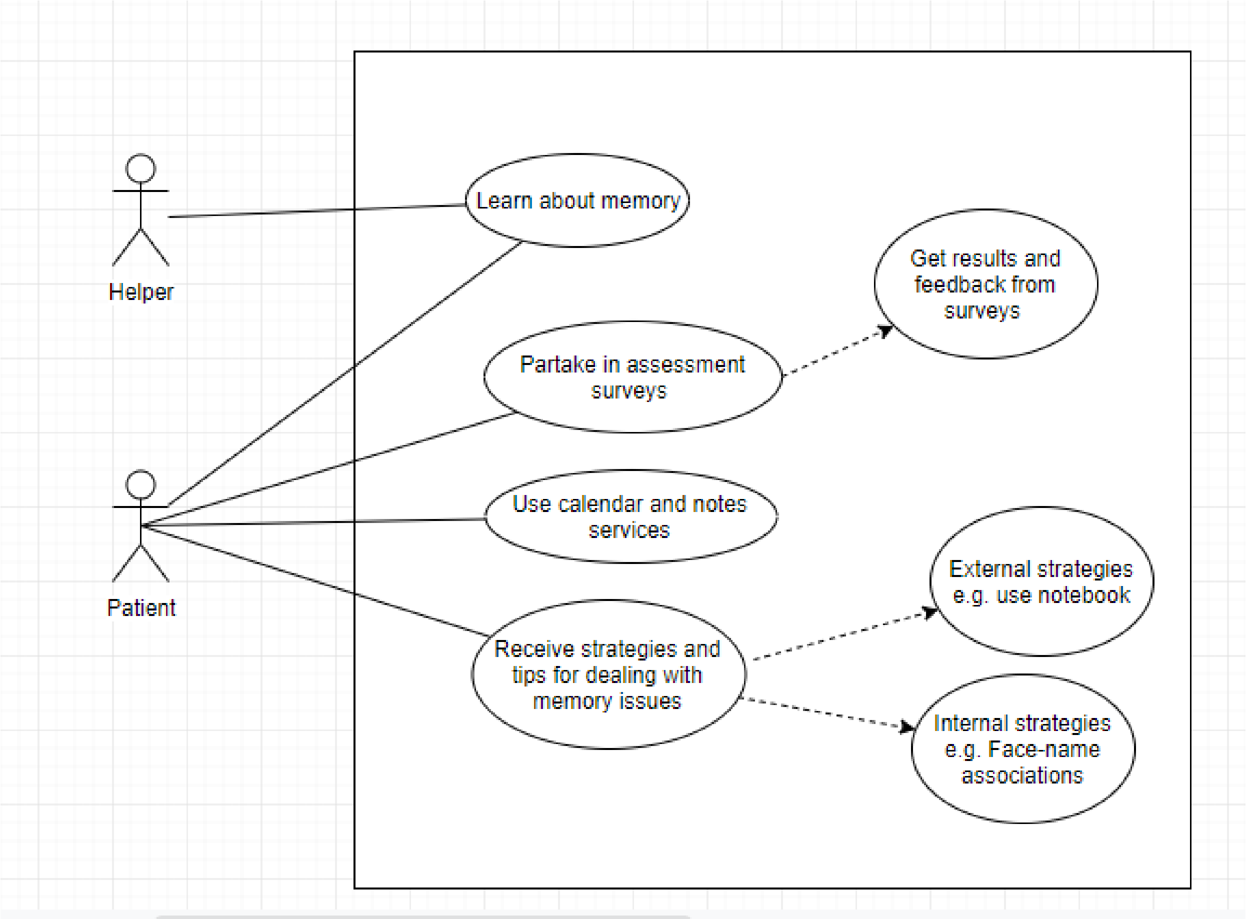

Helper
Patient
Learn about memory
Partake in assessment surveys
Get results and feedback from surveys
Use calendar and notes services
Receive strategies and tips for dealing with memory issues
External strategies e.g. use notebook
Internal strategies e.g. Face-name associations

Figure 3.1 UML Activity Diagram

3.2 Class Diagram

The class diagram below shows the basic structure the application will look to follow. It shows the applications classes, attributes and operations of the classes. It also depicts the relationships between the classes. The user contains the UserID and password attributes. They have the ability to login or commence the registration process. The registration process, for first time users, collects additional attributes from the users and then updates the database for future authentication when logging in. Every user can partake in assessment surveys and view their user profile which shows survey results and user information. They can also use the notes and calendar utilities.

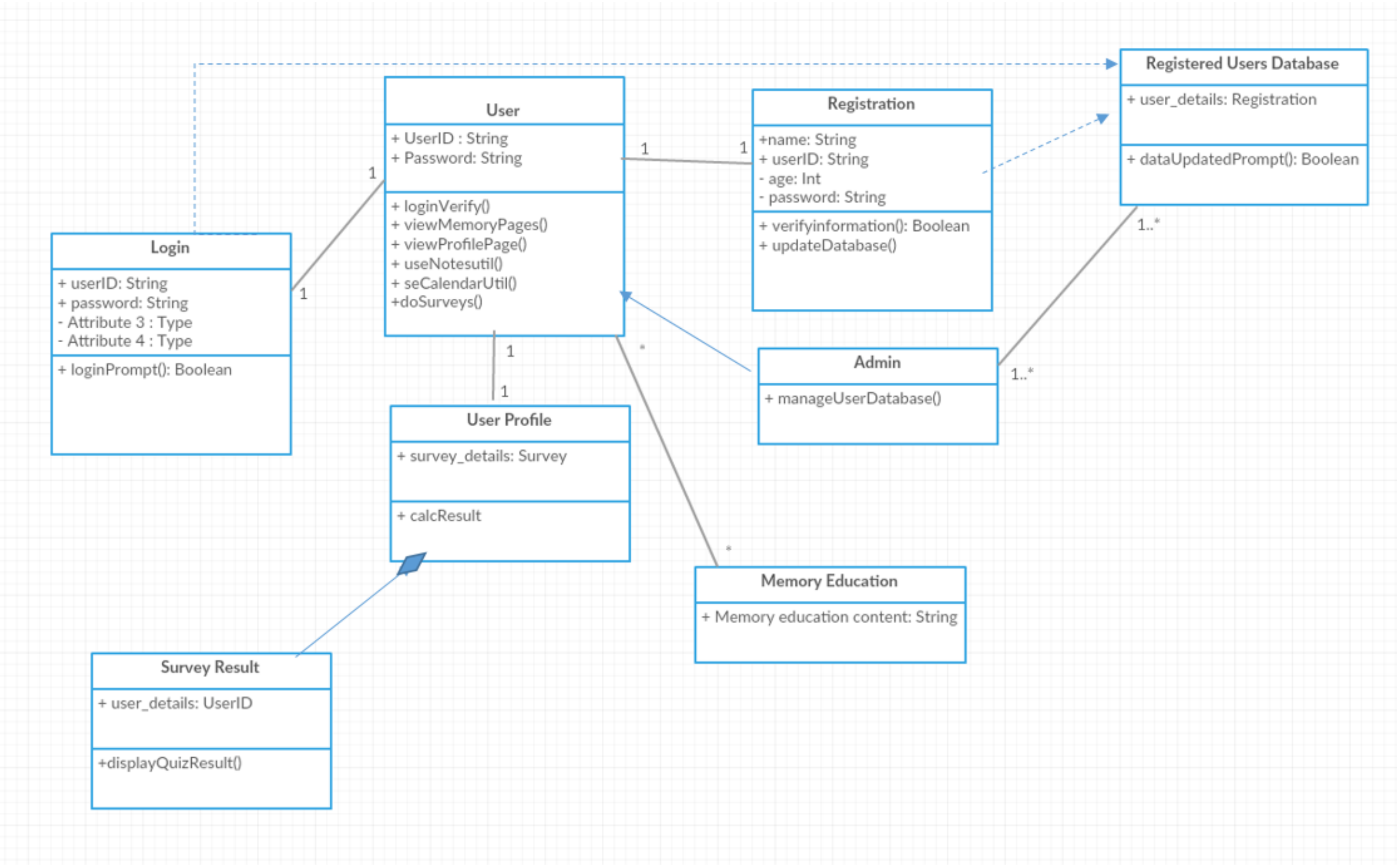


Figure 3.2 UML Class diagram

Chapter four: The Solution

This section describes how a number of the main components and modules of the application were built in order to create the final application. In the following pages, a number of written explanations in conjunction with a number of useful code snippets have been provided. These explain how certain goals were achieved while developing the application.

4.1.1 Application Structure

Ionic applications are built with Cordova. Cordova is basically a way of packaging the HTML, JS and CSS files into apps that can be run on desktop and mobile devices and also provides a plugin architecture for getting access to native functionality that is not available for JS to run from a web browser.

Ionic is built on AngularJs which is a Javascript framework that extends the vocabulary of HTML. The application created lies in the inner square of Ionic/Angular as the app is built with these two frameworks. NgCordova is used underneath to build and emulate the platform app. Apache Cordova is a set of APIs that assist in accessing the native functions of android and IOS platforms. As shown in appendix 4.1 this is all wrapped in node.js which must be installed initially.

The agile software development approach was followed while developing the application. The motivating factor for following this approach was the that the requirements and solutions evolved throughout the project and though the memory intervention course was a solid base to build around there was a lot of flexibility in what could be done with this brief. I had also previously been on a team where we developed a mobile application and following the agile software approach was very beneficial to us on that occasion. The diagram in appendix 4.2 shows the general cycle followed during development.

This cycle was started off by gathering the primary requirements from the memory intervention course as well as investigating and researching the project background. Implementations of technologies were also investigated. Development then started based on this investigation and the cycle continued henceforth.

4.1.2 Setting Up application

To initialise the new application the following line was run from the terminal:

```
$ ionic start MyIonicProject blank
```

This line initialized a blank Ionic project which was the most suitable to start for this task. The CLI will generate a project with name “MyIonicProject”, based on the blank template which has basic scaffolding for an Ionic project with one example Home page. After finishing the initial setup, the project directory was able to be accessed and the application could be run with the following line.

```
$ ionic serve
```

Development could now begin building out from this initial template.

4.1.3 Page creation

To generate new pages for the ionic app the following line is run:

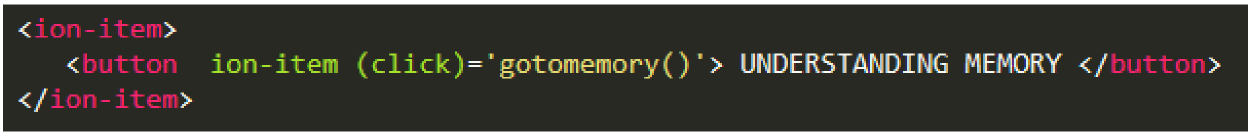
```
$ ionic generate page pageName
```

This line creates a new page in your ionic app. It will create a HTML, TypeScript and CSS file for this page. Though all pages of the app were not instantiated at the beginning phase of development, key pages such as the Home Page, Strategies and Tips Pages were with other pages being added as needed.

4.1.4 Page Navigation

Page navigation was a crucial element in development as we wanted users to be able to navigate through the app easily and intuitively. A side menu was used to provide quick access to essential pages. To create the side menu the following code was placed in the app.html file(See appendix 2.3).

This code creates a menu button accessible on the left side of the screen. Buttons that can be used to navigate to the other pages where then placed in the ion-content field of the same file.



```
<ion-item>
   <button  ion-item (click)='gotomemory()'> UNDERSTANDING MEMORY </button>
</ion-item>
```

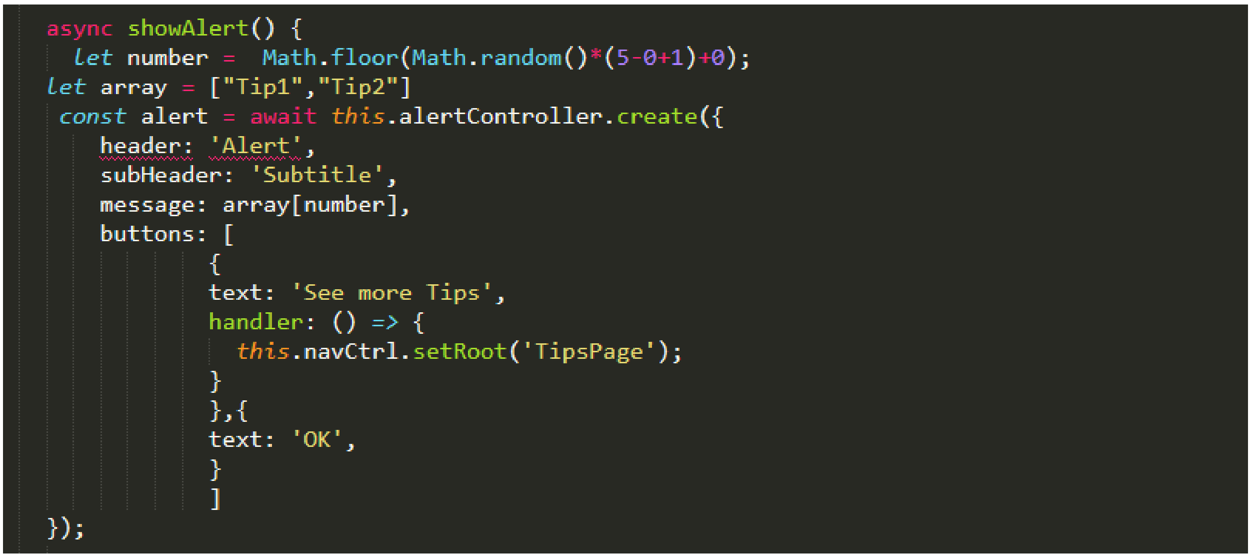
As seen above an "on click" function is then attached to the button which calls a function that navigates to the specified page. This function was defined in the app.component.ts file as follows:

```
gotomemory() {
    this.nav.setRoot('MemoryPage');
  }
```

This navigation technique was used throughout the project whenever navigating to a different page at the click of a button. See appendix 5.2.

4.1.5 Presenting dialog boxes to the user

One functionality the app offered was presenting the user with a daily memory tip such as advice on eating healthy or to use a calendar daily. The user would get one random tip every time they logged onto the application. This was done using an alert controller which would display every time the user successfully logged in. The alert controller was instantiated in the constructor of the login and then the showAlert() function that produces the daily alert was placed inside login() function so that whenever the user logged in successfully an alert would show with a daily tip.



```
async showAlert() {
  let number =  Math.floor(Math.random()*(5-0+1)+0);
let array = ["Tip1","Tip2"]
 const alert = await this.alertController.create({
    header: 'Alert',
    subHeader: 'Subtitle',
    message: array[number],
    buttons: [
            {
            text: 'See more Tips',
            handler: () => {
              this.navCtrl.setRoot('TipsPage');
            }
            },{
            text: 'OK',
            }
            ]
});
```

To ensure the alerts were random an array of all the possible tips the alert could suggest was created. A random number generator was instantiated and this number was then used to pick an array element and display it as shown. Two buttons were also attached to the alert. The “Ok” button simply causes the alert to subside while the “See more tips” button navigates the user to the Tips Page. See appendix 5

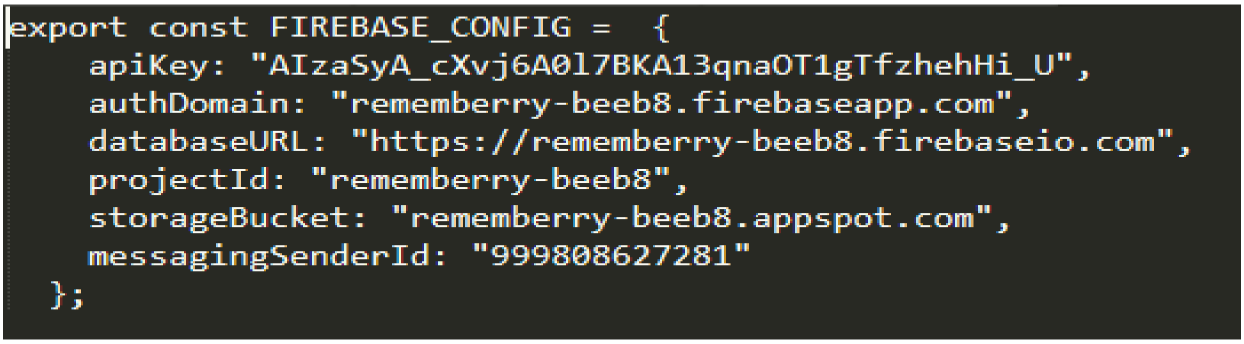
4.1.6 Setting up database

The database was set up to be used for user authentication and storing user information such as user name and age as well as survey results. These were stored in data objects.

Before using Firebase with the app Angular Fire 2 (The official angular library for firebase) had to be installed with the following command in the command prompt.

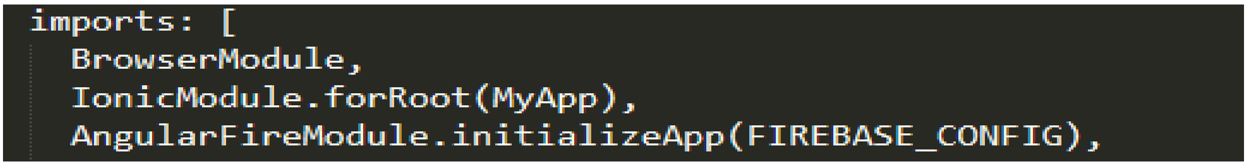
```
npm install @angular/fire firebase –save
```

After this a new project was created on Firebase. The firebase credentials were retrieved from firebase and then added to the ionic application by adding the following code to the firebase.credentials.ts file.



```
export const FIREBASE_CONFIG =  {
    apiKey: "AIzaSyA_cXvj6A0l7BKA13qnaOT1gTfzhehHi_U",
    authDomain: "rememberry-beeb8.firebaseapp.com",
    databaseURL: "https://rememberry-beeb8.firebaseio.com",
    projectId: "rememberry-beeb8",
    storageBucket: "rememberry-beeb8.appspot.com",
    messagingSenderId: "999808627281"
  };
```

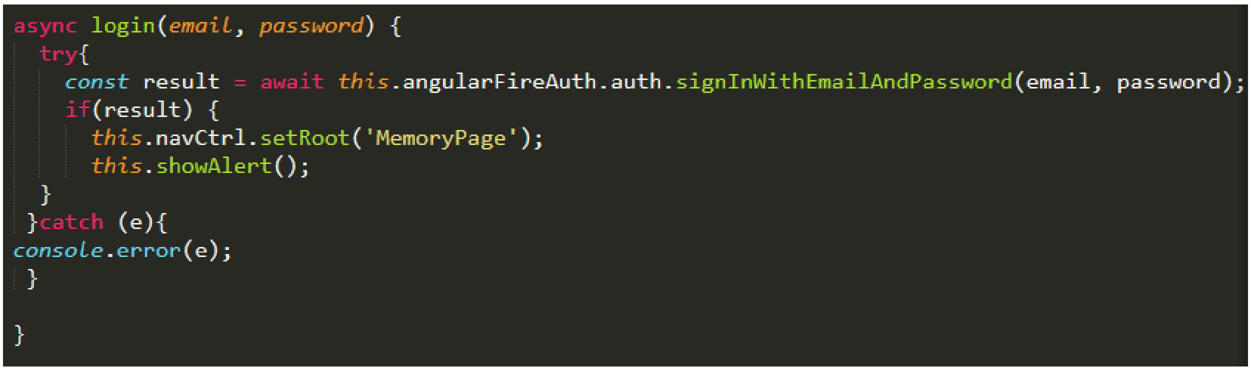
This is then initialized in the app.module.ts file as follows to connect firebase to the application.



```
 imports: [
   BrowserModule,
   IonicModule.forRoot(MyApp),
   AngularFireModule.initializeApp(FIREBASE_CONFIG),
```

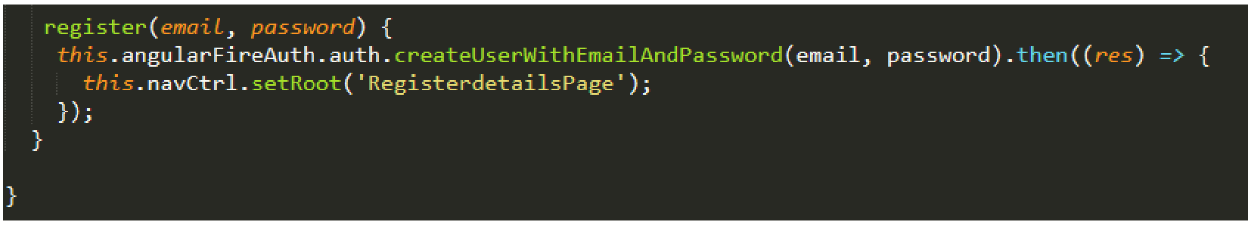
Firebase authentication was chosen to manage the user’s registrations and login capabilities. Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to the app. It supports authentication using passwords, phone numbers, popular federated identity providers like Google, Facebook and Twitter, and more. For this app simple email and password was used to authenticate users. This provides us with functions to register new users and sign in and out existing users. All these functions trigger the “onAuthStateChanged” observer and so successful auth changes are handled in the handler instead of with promises that these functions return.

To create the functions for registering and signing in a user we call the “createUserWithEmailandPassword” and “signInwithEmailandPassword” methods. Some basic HTML is also written to allow the user to input their email and password, along with a few other details when registering.



```
async login(email, password) {
  try{
    const result = await this.angularFireAuth.auth.signInWithEmailAndPassword(email, password);
    if(result) {
      this.navCtrl.setRoot('MemoryPage');
      this.showAlert();
    }
 }catch (e){
console.error(e);
 }

}
```



```
  register(email, password) {
   this.angularFireAuth.auth.createUserWithEmailAndPassword(email, password).then((res) => {
     this.navCtrl.setRoot('RegisterdetailsPage');
   });
  }

}
```

4.1.7 Creating the notes page

The notes page would be used by the user to write and store notes within the app and acts as an external memory aid for the user.

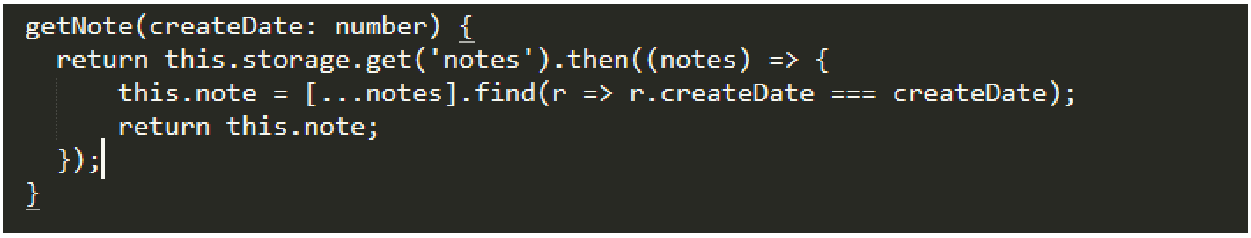
Before starting the creation of the notes page the ionic storage module was imported as it would be the key component used to store the note data and ensure data continuity. Ionic Storage is an easy way to store key/value pairs and JSON objects. Storage uses a variety of storage engines underneath, picking the best one available depending on the platform. After this was imported the necessary pages for the notes function were created. These included the ‘notes page’, ‘add-note page’ and ‘view-note page’. A note was then explicitly defined by creating our own custom type with an interface as shown in appendix 2.2.

This will allow us to import a Note type elsewhere in our application, which will allow us to force our notes to follow the format defined above. Each note must have a title that is a string, content that is a string, a date that is in the date format and a create date that is a number which will be used later to identify specific notes.

While the pages created beforehand are responsible for displaying the data on the screen the majority of the logic implemented will be carried out in a service created ‘notes-service’. This service does not display anything but is used by the pages in the application to perform data/logic operations. This keeps the code on the pages to a minimal and promotes the reuse of code. The notes service will implement methods which will allow us to do the following:

- Create a note and add it to an array in the data service
- Delete notes from that array
- Find and return a specific note by its create date
- Save the array of notes to storage
- Load the array of notes from storage

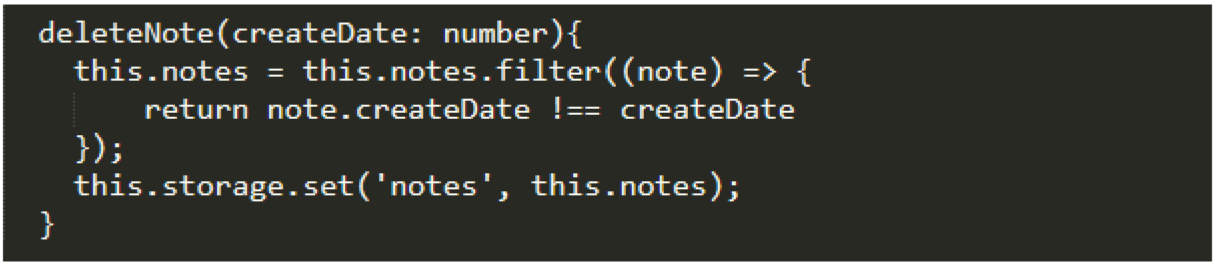
A “saveNote()” function is created to push a new note made on to the notes array from the Add Notes page and a “getAllNotes()” function is used to display a list of all the notes on the Notes page. To retrieve a specific note a “getNote(note.createDate)” function was created in as follows.



```
getNote(createDate: number) {
  return this.storage.get('notes').then((notes) => {
      this.note = [...notes].find(r => r.createDate === createDate);
      return this.note;
  });
}
```

This function takes in the notes create date, which is the number of milliseconds elapsed since January 1, 1970, and is a suitable unique identifier for this purpose. I uses this create date to find the note in the array with the same date and then returns it.

A deleteNote() function was created to delete specific notes. Again the create date was used as the unique identifier.



```
deleteNote(createDate: number){
  this.notes = this.notes.filter((note) => {
      return note.createDate !== createDate
  });
  this.storage.set('notes', this.notes);
}
```

Instead of explicitly deleting the identified note the filter function was used to create a new array with the unwanted note filtered out and then this new filtered array was set in storage as the new notes array.

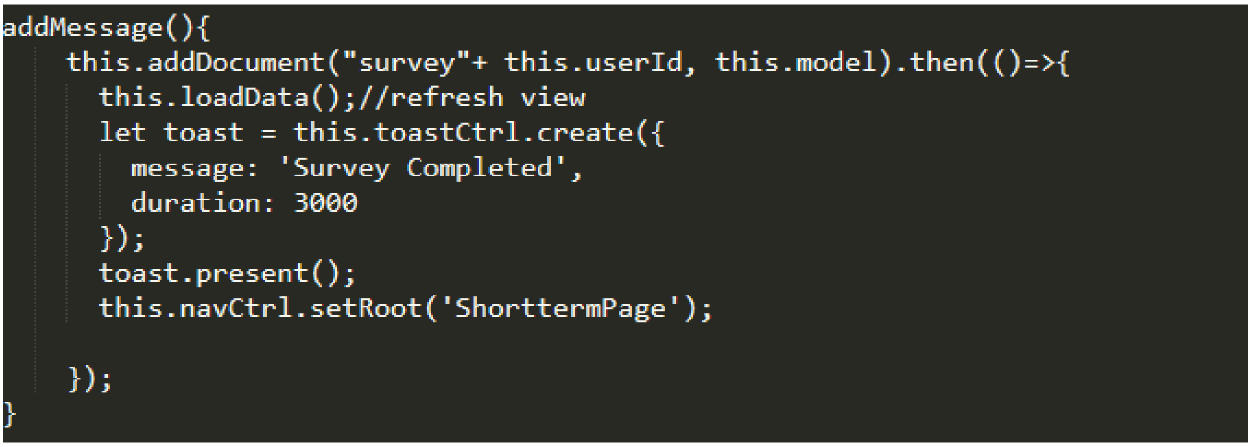
After the notes service was finished the functions were implemented in the suitable pages and the accompanying HTML files were written to display the notes correctly. See appendix 5.3

4.1.8 Creating memory surveys

The memory surveys are provided as a way for users to track their own progress in terms of their memory improvement. They are Likert scale based surveys which ask the users to rate how much they agree with given statements on a scale of one to five.

To create the memory assessment surveys ionic form items were used to push data to a firestore database, which is a type of database available to use on Firebase. Ion-items were created as shown in appendix 2.1.

The input value chosen by the user is linked to a model component with [(ngModel)] = “model.Q1” and this is done for all the questions on the survey. In the corresponding .ts file functions were written to push the model components recorded up onto the firestore database. To ensure that data from these surveys stayed unique to each user the user’s identification number, that is automatically created when a new user registers, was used.



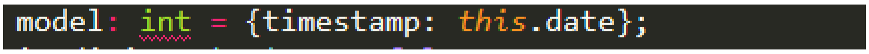
```
addMessage(){
    this.addDocument("survey"+ this.userId, this.model).then(()=>{
      this.loadData();//refresh view
      let toast = this.toastCtrl.create({
        message: 'Survey Completed',
        duration: 3000
      });
      toast.present();
      this.navCtrl.setRoot('ShorttermPage');

    });
}
```

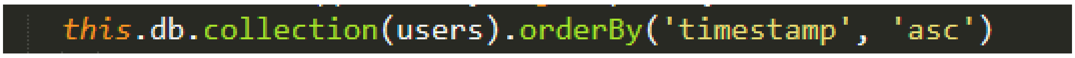
The addDocument() fuction shown above will automatically create a document in the database using the first object its given as the name and the second object its given will be the data it records in the database. So each user will have their own database for each survey they participate in which will be named using their unique ID number. See appendix 5.6

4.1.9 Presenting database data

The data collected from the memory surveys is presented to the user on the users profile page. To present this data it needs to be pulled from the fire-store database it is held in. A bar chart was chosen as the main way to show the users survey scores as it gives them a clear indication of improvement or decline in a visually intuitive way. The first problem faced in employing this was that data pushed to fire-store databases is not ordered in any way so when the data is pulled from the database the survey results were displayed in a random fashion. The survey results would best be displayed according to date and to do this timestamps were added to the data model when it was initially uploaded as a field and then this field was used to order the data from earliest date to latest date.

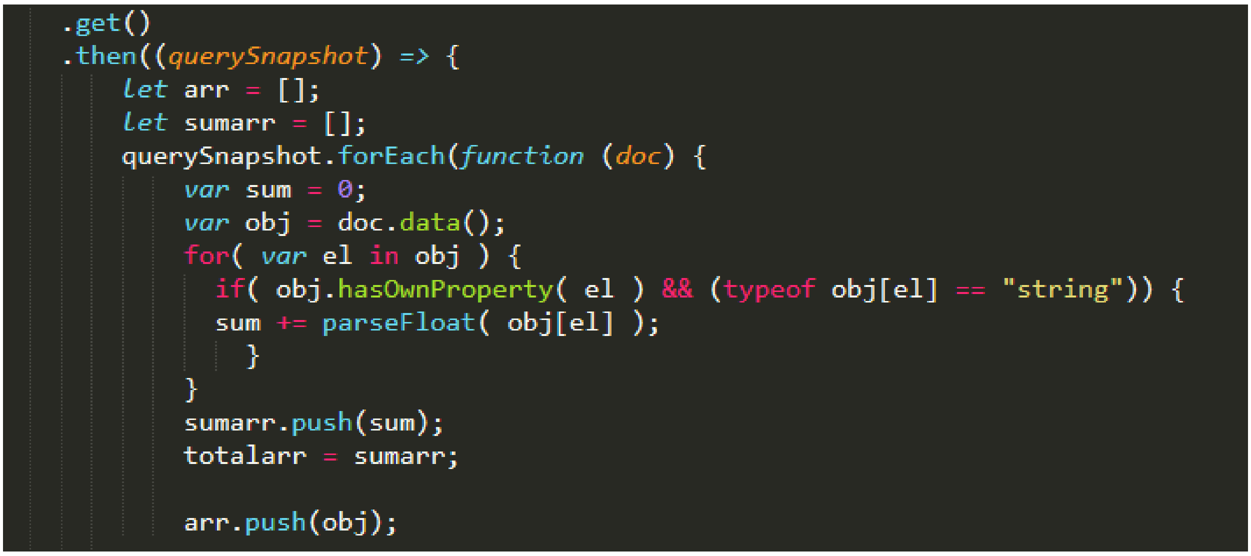


```
model: int = {timestamp: this.date};
```



```
  this.db.collection(users).orderBy('timestamp', 'asc')
```

When the data is initially pulled from fire-store it is in the form of a data object as such {(Q1: 3), (Q2:4)..., (Q6:1).We wanted the bar chart to show the total score of each survey which would be a mark out of 30. So firstly we had to take each data object and extract the mark for each question and then sum all these marks up and push this sum into an array which would be represented by a bar in the bar chart. This was done as follows.



```
    .get()
    .then((querySnapshot) => {
        let arr = [];
        let sumarr = [];
        querySnapshot.forEach(function (doc) {
            var sum = 0;
            var obj = doc.data();
            for( var el in obj ) {
              if( obj.hasOwnProperty( el ) && (typeof obj[el] == "string")) {
              sum += parseFloat( obj[el] );
                }
            }
            sumarr.push(sum);
            totalarr = sumarr;

            arr.push(obj);
```

A for loop is initiated that will run through all elements of the data object. If the element has its own property and is a string, as we do not want to add the timestamp value, the element would be parsed into a float and added to the sum variable. The final sum value is then pushed onto the sum array. The end result is a bar chart that looks as follows. See appendix 5.4

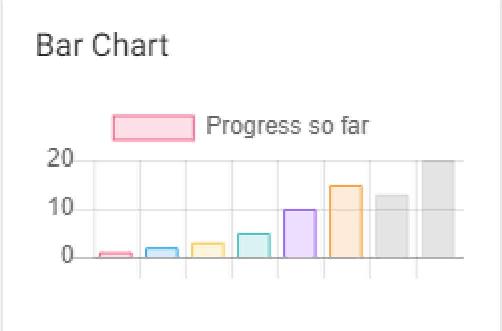


Figure 4.1 Survey results bar chart

4.2 Version control

For version control GitHub was used. Firstly a new repository was created on GitHub and then this was linked to my ionic project via node.js. Files from the ionic project are then added to the local repository with the command ‘*$ git add .*’ . Files are then committed to your local repository along with a message or tag as such ‘*$ git commit -m "First commit"*’. This commits the changes and prepares them to be pushed to a remote repository. Then the URL for the remote repository where your local repository will be pushed is added with ‘*$ git remote add origin remote repository URL*’. Finally to push the changes in your local repository to GitHub you run ‘*$ git push origin master*’, which pushes the changes up to the remote repository you specified as the origin. An effort was made to push any new successful changes up to GitHub on completion.

4.3 Problems encountered

There were a number of key problems faced whilst developing this application. The major problem faced was the ionic framework updates. Ionic had recently updated to ionic 4 from ionic 3. A lot of the ionic plugins and dependencies that had once worked on ionic 3 now gave errors with ionic 4 and this halted progress in the project numerous times. A lot more time was allocated to developing work-arounds to these issues than was originally estimated.

Another problem faced was creating the fire-store database to use to store survey data. As this was the first time using fire-store a lot of background research had to go into how to implement it properly, how to push data onto the database and what form the data had to be in. Extracting the data was also problematic as it was extracted as a JavaScript data object but for the purposes of the project it needed to be extracted as an array. As fire-store is a fairly new addition to the firebase services, the documentation didn't go in depth and there were only a few online tutorials and walkthroughs to reference so a lot of trial and error had to be employed.

Chapter five: Evaluation

This chapter presents an evaluation of the solution. As this was a mobile application developed for actual users the most suitable and prevalent means of testing would be through usability testing. It was expected that this method would provide useful feedback from a range of different potential users. Given more time to conduct testing tests would be done to determine whether the app helps to improve different aspects of memory. Tests would also be done to determine whether it improves users understanding of memory and how and why memory deteriorates.

5.1 Software Testing

Software testing was an essential part of the evaluation process. The main method used to test the software was unit testing. This was particularly suitable for this project as most of the application pages functioned independently from other pages bar a few exceptions such as the profile page. Unit tests were performed on code modules at particular times during the development process. After a page is initially finalised a unit test is carried out on that page independently and also after any significant code changes are made to the page.

5.1.1 Unit test examples

An important unit test carried out was for the functionality of the login and registration pages. Test accounts were created and then logins were performed with the test accounts. These tested the login() and register() functions. Particularly if they accepted different types of email address and passwords. The formats of the objects pushed to firebase by these functions was also observed to make sure it was suitable for the purposes intended such as displaying user information on the profile page.

Unit testing was also performed on the notes utility. The createNote(), editNote() and deleteNote() functions were all tested to see if they performed as intended. Tests were also carried out to determine if the note stored by the app would still be present after the application was restarted.

5.1.2 Stress Testing

Stress tests were also performed. These included a stress test to assess the apps ability to hold survey information. 50 surveys were done to see if the app could hold all the results without any issues. 50 user accounts were also created to see if the database servers were able to hold them with no issues. The application passed both of these stress tests with no issues.

5.2.1 Usability Testing Procedure

Usability testing in a way is like black box testing of an application to deduce if the product built is easy to use and easy to learn how to use(M, Nagaraj, Gattu, & Shetty, 2014). The goal of these tests was to quickly learn any design flaws that had not been considered prior to the testing. To test the overall usability of the application various usability tests were conducted. The main aim of the sessions being to provide qualitative feedback that would then be critically analysed and used to make improvements to the design of the application.

Five candidates were recruited to participate in the tests. They were specifically chosen to cover a varying range of demographics. All participants were asked to rate how familiar they are with mobile technology and technology in general as this would be an important aspect to keep in mind when analysing the test data. During their time with the app they were also asked to complete a memory and general cognition survey which was built into the app. The survey used a 5 point Likert scale for them to choose their answers from with 1 – Strongly disagree and 5 – Strongly agree. This data would also be taken into account when assessing the test results. The subject's data is shown in the table below.

Participant Data table				
Participant	**Gender**	**Age**	**Cognition Assessment ***	**Experience with mobile devices**
1	Male	19	26	Extremely Familiar
2	Male	36	27	Quite familiar
3	Female	42	24	Reasonably familiar
4	Male	65	21	Little experience
5	Female	22	28	Extremely Familiar

*((**0 – Poor Memory and cognition, 30 – Excellent Memory and Cognition))**

The participants would be measured on two different metrics. These would be the objective and subjective metrics. The objective metrics were gathered by recording if subjects could complete tasks assigned to them regarding the app, how long it took to complete given tasks and any errors made on the user's side or the applications side while performing the tasks. For the subjective metrics the test moderator monitored the participant's interactions with the app. Each participant was encouraged to "think out loud" while interacting with the app following the Concurrent Think aloud moderation technique (Van Den Haak, De Jong, & Schellens, 2003). This technique helps vocalize any issues the user might have with the app and elicits real-time feedback and emotional responses while using the app. A retrospective approach was also taken in terms of the subjective metric where the users were asked to rate the app under certain criteria and also asked the following questions.

- Where you comfortable navigating through the apps many pages?
- Is there anything you felt was not intuitive or difficult to figure out?
- Do you have any recommendations for future iterations of the app?

The participants were given a phone that was linked up to display the app using a software called Ionic DevApp. The Ionic DevApp is a free app that makes it easy to run your Ionic app directly on your iOS or Android device without having to deal with native SDK installation. It was very important that the test subjects interacted with the application in the manner that it would finally be deployed at a later date and not on a computer screen as to identify any issues they might find in terms display.

Tasks given and descriptions	
Tasks	**Description**
Task 1	Create an account and sign in
Task 2	Navigate through pages and learn about memory
Task 3	Use the notes and calendar functions
Task 4	Complete a memory survey
Task 5	Visit profile page and look at progress

5.2.2 Findings from Evaluation

In this section the findings from the usability test sessions are presented and reviewed. We discuss the feedbacks we obtained on each of the tasks which were presented to the test participants.

Task 1:

The first task was to create a new account on the app and then sign in. Participants 1-4 navigated through this process with no issues. The process was designed to be simple and straight forward with users just asked to provide an email address and password along with their name and age. Participant 5 had entered their chosen password incorrectly during registration therefore was unable to sign in after registration. **The participant recommend double entry of the password could be added to the registration page to ensure passwords are entered correctly initially** and this was recorded by the moderator to consider for future iterations.

Task 2:

The second task asked the participants to learn about memory on the various memory education pages. After signing in to the app a user will be brought *to the initial memory education page* and from there they can navigate to similar pages e.g. *Short term memory, long term memory and attention pages*. All participants navigated through these various pages with ease and never got lost or confused. **However participant 3 expressed that on arriving at the initial page she was not sure what to do next and said that a message of some sort explaining the next steps to take in using the app would be useful.**

Task 3:

The third task asked the participants to use the *notes* and *calendar* functions of the application. These were accessed using the *side menu* and all participants navigated to them with ease. The *notes utility* allows users to write notes that save in the app for further reference. Notes can also be edited and deleted. All the participants were able to create, delete and edit notes without any problems. They then used the *calendar* utility. This allows users to view a month by month calendar

Task 4:

The participants were then directed to a *memory survey page* and asked to complete the survey. While this survey was used to gather information about the participants for further analysis after testing it was also the general template for all the surveys included in the app and so would provide us with useful information as to how users interact with this page. All participants conducted the survey with no issues. **The participants enjoyed how the survey was simple and not too tedious with none of them losing interest.**

Task 5:

After the survey the participants were directed to the *profile page*. Here the users can *view their progress* in the multiple surveys they have completed. All participants were able to navigate to the survey they had completed and view their progress in a text format and also in a *graphical format* in the means of a bar chart. Gathering from the thinking aloud technique all participants fully understood what all the figures on the page meant and were not confused.

See appendix for all pages referenced.

5.3 External Evaluations

Though tests were not performed on the exact target user group this application was meant to be used by, one of the intervention programs that the application was based on did perform such tests. The study they carried out examined the effectiveness of the program on the subject's day to day memory functions and use of memory strategies and tips.

A one group pre-test and post-test prospective clinical study was utilised. Clients attended a group session one hour a week for six weeks and tests were performed on the clients at the beginning, two weeks after they completed the course and finally three months later.

Results of the study were positive and provide strong evidence to support the role that occupational therapy can provide in terms of memory rehabilitation (Martin, 2016).

5.4 Other recommendations

After the testing was completed we asked the participants if they had any final recommendations or remarks. Participants 1 and 2 said there was no major changes they would make to the application and thought it was an overall good experience. Participant 3 said they would have appreciated more feedback throughout the application on completion of tasks and on navigating to new pages etc. Participant 4 expressed he would of liked the text on some of the memory education pages to be slightly bigger. Participant 5 felt that the memory education section was definitely beneficial but some more tasks to exercise the techniques learnt in this section would improve the app drastically.

Chapter six: Conclusion

6.1 Project Summary

In this chapter we discuss the overall approach taken for this project as well as our findings and conclusions. We will first summarise the work that has been completed and conclude by identifying future improvements that could be made to the application.

This project involved designing an android based mobile application that is easy to use and easily accessible that carried functionalities and utilities based mainly on the memory intervention course mentioned previously along with added functionalities to accompany this.

Before undertaking this project it was important to understand the background of the key components involved. In this case those components were the human memory and how it works as well as how technology can be used to aid in memory support and rehabilitation. Comprehensive research was done in both these areas. After conducting this research the task of creating the application described above felt more achievable and manageable.

There was a substantial learning curve to overcome when development of this application began. It was the first time using most of the software needed to create the application and so a lot of time was set aside to becoming familiar with the likes of the ionic framework, node,je, firebase and Cordova. An agile software development approach was followed during development. The benefits of this were its flexibility especially when it came to adding new unplanned features into the application later on in the development process.

A series of usability testing sessions were used to test the initial version of the app. The software development efforts leading up to this testing phase were focused on delivering an application that was stable, dependable and met the functional requirements. A number of users were allowed to use the app and their feedback was taken into account to implement into further iterations. The feedback was very positive and all the constructive criticism given was taken into account and implemented back into the app where it could be.

6.2 Future improvements

In this section a number of possible future changes and improvements to the application are presented. During the development of the app some potential changes became apparent. While the agile software development approach was used and this allowed changes to the initial design and functionality to be accommodated the changes discussed below could not be implemented due mostly to time constraints. Adjustments and improves were also identified in the evaluation section (See 5 above).

The first improvement we would recommend is adding more accompanying quizzes to the application. Self-assessment surveys are provided to the users to track how they feel their cognitive functions are improving in different areas of their memory. However adding more quiz based utilities to the application would provide the user with a more objectively based assessment of their memory. It would also allow the user to test out the multiple memory tips and strategies the app provides them with. The app provides examples of how these tips and strategies should be used but memory quizzes within the app would allow them to apply what they have learnt in a tested environment.

Another improvement we would like to add to the app is a way to push content onto the app easily and quickly. The content covered in this app is part of a field that is constantly growing and new information such as test study's and papers surface every week. Ideally the app would come with a default set of information that is accessible to the user with or without a network connection. Then if the user had a network connection the app could check to see if there is any additional information available for download and display this to the user.

More sophisticated tracking is another improvement we would like to make. At the moment the app tracks the user's survey results and provides them with feedback as to how they are progressing. However using more fields to assess the user's results would improve the tracking. Incorporating the user's age into the analysis along with how frequently they have been using the app would provide a better insight into how effective the application is.

6.3 Conclusion

A mobile application has been developed which provides a user with an easy to use and accessible memory intervention course. The application educates users on the human memory as well as providing them with techniques and strategies to help with memory. This project has shown how the standard pen and paper intervention courses used today can be adapted into mobile applications which can hopefully reach a wider audience. User feedback on the application, though limited, was positive but future testing on real world patients would be required to objectively verify the usefulness of this application going forward. Working on this project and in this field has been extremely rewarding and future development on this application is a very exciting prospect.

References

Anon., 2018. *Toronto Memory Program.* [Online]
Available at: https://www.torontomemoryprogram.com/alzheimers-disease-risk-factors/
[Accessed March 2019].

Baddeley, A. D. E. M. W. &. A. M. C., 2009. *Memory.* 6th ed. Hove: England: Psychology Press..

Cahill, S., Macijauskiene, J., Nygård, A., Faulkner, J., & Hagen, I. (2008). Technology in dementia care. **久留米大学大学院ニュースレター**, *47*, 55–60. Retrieved from http://www.tara.tcd.ie/xmlui/bitstream/handle/2262/49835/Technology in dementia care.pdf?sequence=1&isAllowed=y

Cour, P. &. G. K., 1990. Rivermead Behavioural Memory Test. pp. 130-141..

George Savulich, T. p., 2017. Cognotive Training using a Novel memory game on an ipad in patinets with mild amnestic cognotive impairment.

Gross AL1, P. J. S. A. K. A. K. J. S. J. S. Q. R. G., 2012. Aging Mental Health. *Memory training*

M, N., Nagaraj, A., Gattu, H., & Shetty, P. K. (2014). Research Study on Importance of Usability Testing/ User Experience (UX) Testing. International Journal of Computer Science and Mobile Computing, 3(10), 78–85. Retrieved from https://doaj.org/article/cb68895b7a684ba29df022a15512242a

Mittal, A., 2017. What is ionic?.

Martin, Á. C. T. S. M. (2016). OUTCOMES OF A MEMORY STRATEGY EDUCATION GROUP INTERVENTION FOR CLIENTS WITH MILD COGNITIVE IMPAIRMENT AND EARLY DEMENTIA. Retrieved from https://academic.oup.com/ageing/article-pdf/45/suppl_2/ii1/23570791/afw159.06.pdf

Mason, D. J., & Kohn, M. L. (2001). *The memory workbook: Breakthrough techniques to exercise your brain and improve your memory. The memory workbook: Breakthrough techniques to exercise your brain and improve your memory.* Oakland, CA, US: New Harbinger Publications.

Owen AM, H. A. G. J., 2010. Putting brain training to the test. Volume 465, pp. 775-8.

P. QUAYHAGEN, M. &. R. C. R. &. A. R. P. &. A. R. J., 1995. A Dyadic Remediation Program for Care Recipients with Dementia.. pp. 153-159.

R. Mielke, M. G. B. K. J. K. K. H. W.-D. H., 1996. Propentofylline enhances brain activation during memory stimulation in Alzheimer's disease. *Neurobiology of Aging,,* Volume 17(4).

Selam Negash, Y. E. G. R. C. P., 2008. *Handbook of Clinical Neurology(Chapter 26 Neuropsychological characterization of mild cognitive impairment).* Volume 88 ed. s.l.:s.n.

Sinicki, A., 2017. An introduction to Firebase – the easiest way to build powerful, cloud-enabled Android apps.

Terry, W., 2018. *Learning and Memory..* 4th ed. New york: Routledge.

Van Den Haak, M. J., De Jong, M. D. T., & Schellens, P. J. (2003). Retrospective vs. concurrent think-aloud protocols: Testing the usability of an online library catalogue. *Behaviour and Information Technology*, *22*(5), 339–351. https://doi.org/10.1080/0044929031000

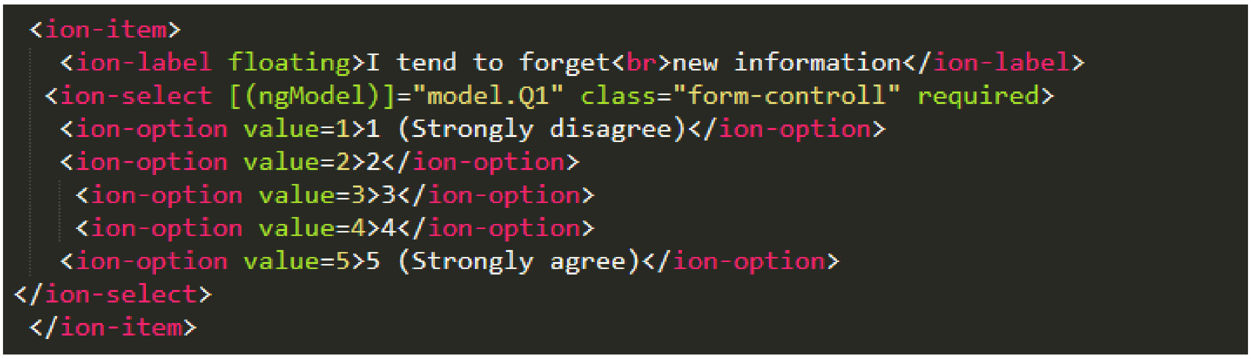
Woods B, A. E. S. A. O. M., 2012. Cognitive stimulation to improve cognitive functioning in people with dementia.. Issue Issue 2.

Zanetti, O. &. Z. G. &. D. G. G. &. D. V. L. &. P. A. &. M. T. &. T. M., 2010. Effectiveness of procedural memory stimulation in mild Alzheimer's disease patients: A controlled study.. *Neuropsychological Rehabilitation: An International Journal*, pp. 263-272.

Appendices

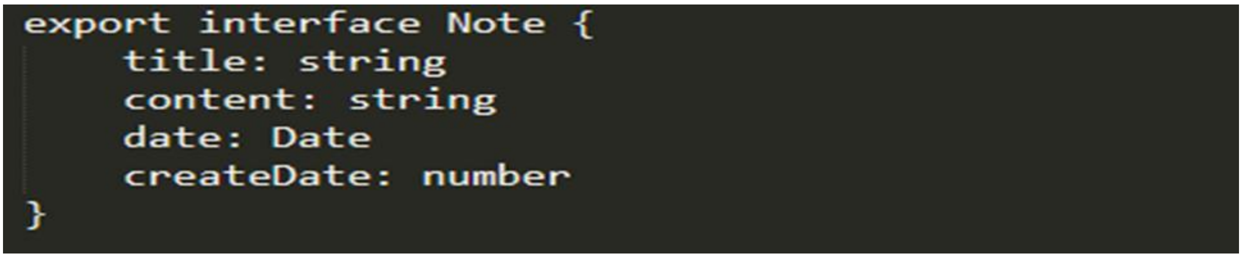
Appendix 1 Code developed for this project.

1.1



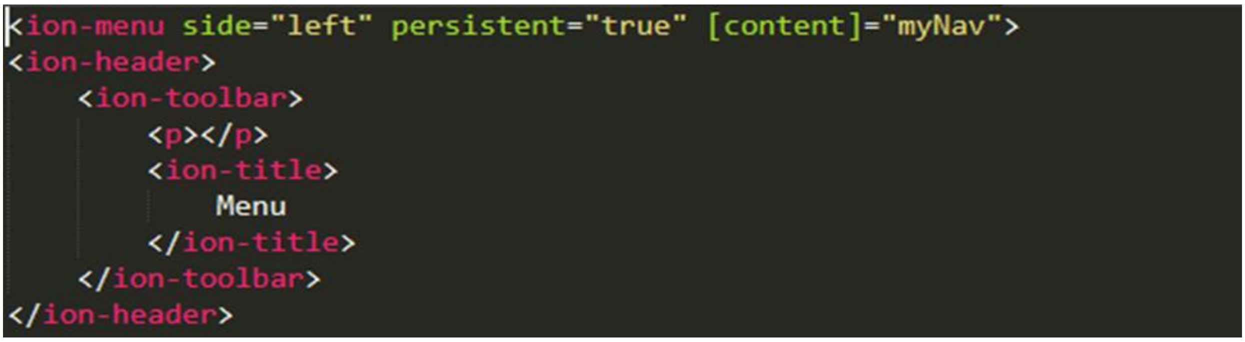
```
 <ion-item>
   <ion-label floating>I tend to forget<br>new information</ion-label>
  <ion-select [(ngModel)]="model.Q1" class="form-controll" required>
   <ion-option value=1>1 (Strongly disagree)</ion-option>
   <ion-option value=2>2</ion-option>
    <ion-option value=3>3</ion-option>
    <ion-option value=4>4</ion-option>
   <ion-option value=5>5 (Strongly agree)</ion-option>
</ion-select>
 </ion-item>
```

1.2



```
export interface Note {
    title: string
    content: string
    date: Date
    createDate: number
}
```

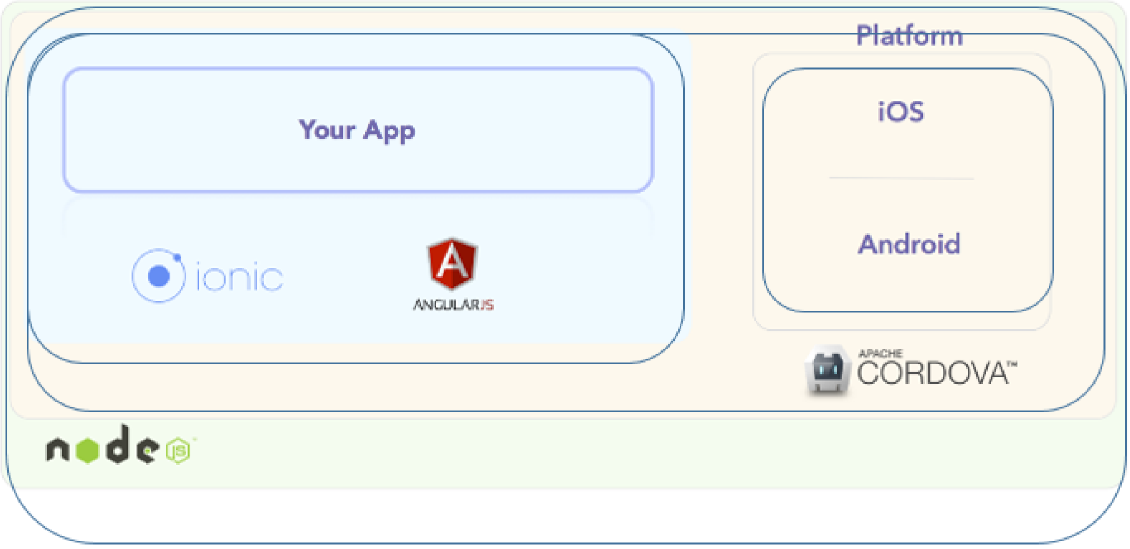
1.3



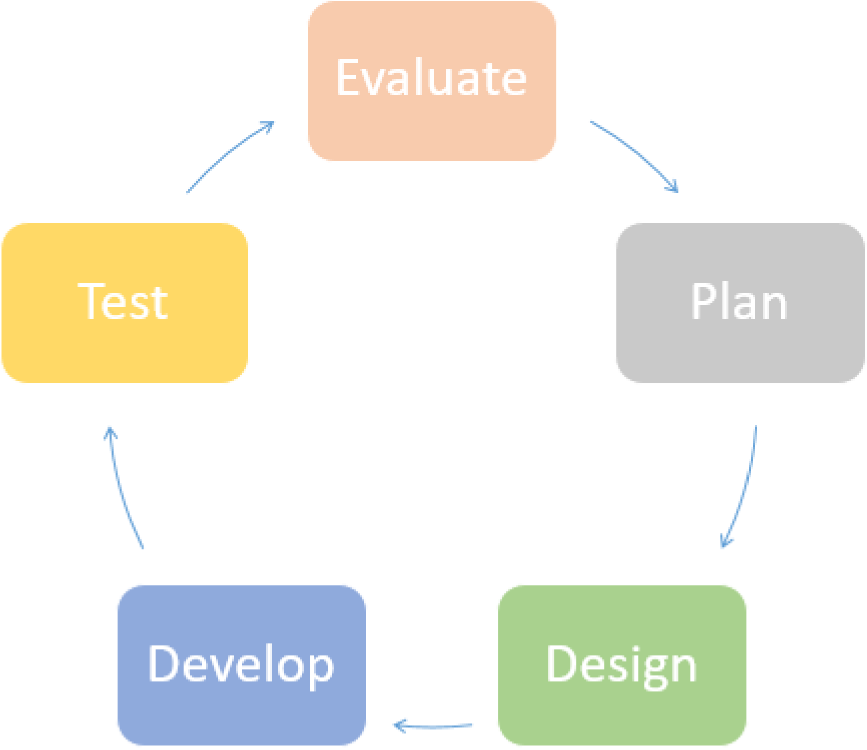
```
<ion-menu side="left" persistent="true" [content]="myNav">
<ion-header>
    <ion-toolbar>
        <p></p>
        <ion-title>
            Menu
        </ion-title>
    </ion-toolbar>
</ion-header>
```

Appendix 2 Framework and Development process diagrams.

2.1

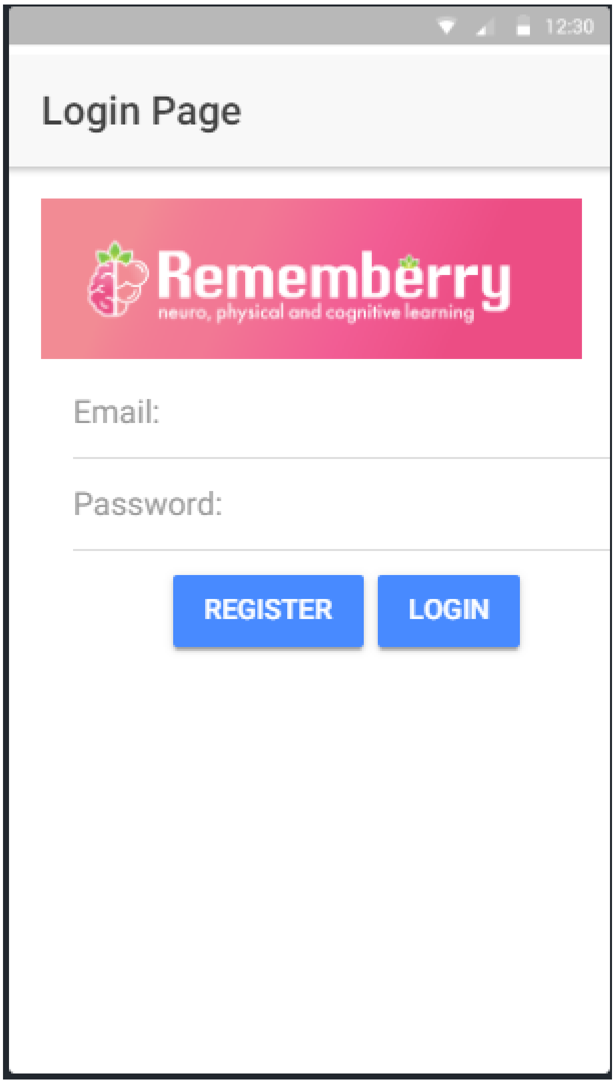


2.2



Appendix 3 Screen shots of the project implementation

3.1 Login page



12:30

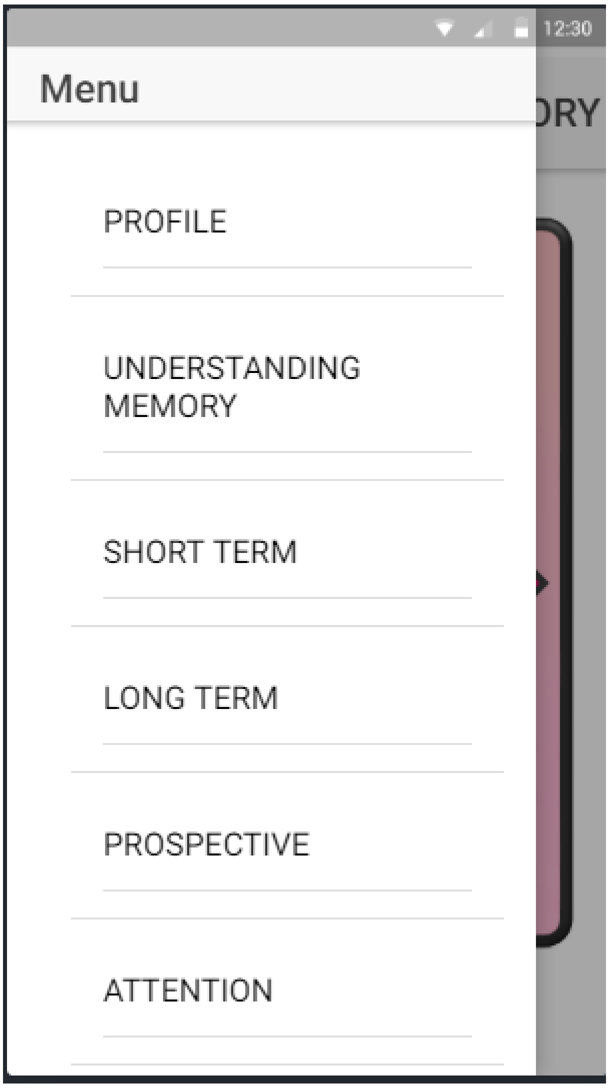
Login Page

Email:

Password:

REGISTER LOGIN

3.2 Registration page



12:30

Menu

PROFILE

UNDERSTANDING MEMORY

SHORT TERM

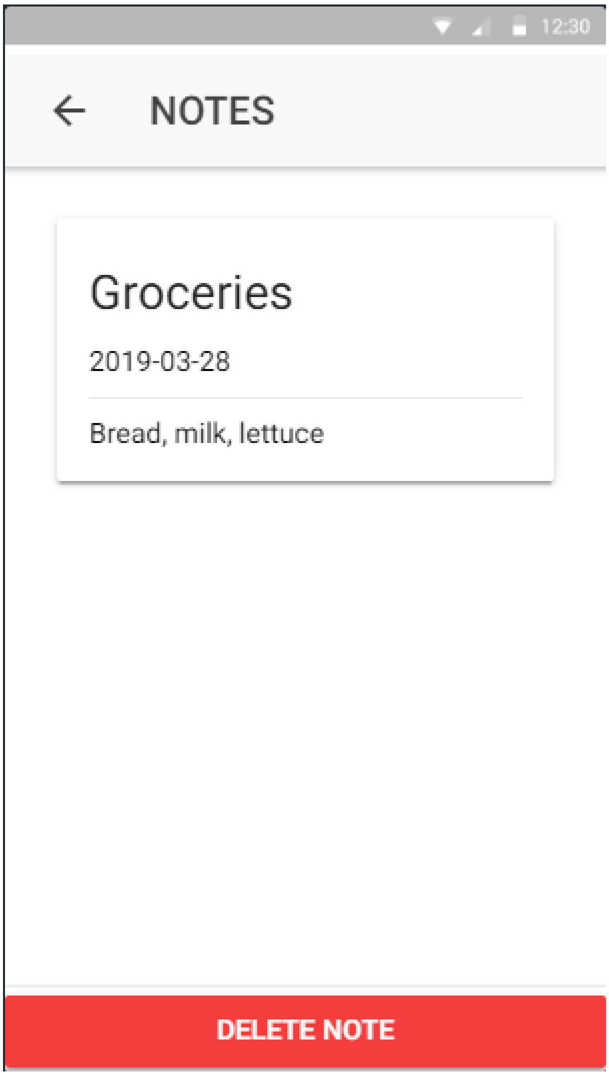
LONG TERM

PROSPECTIVE

ATTENTION

3.3 Notes Page

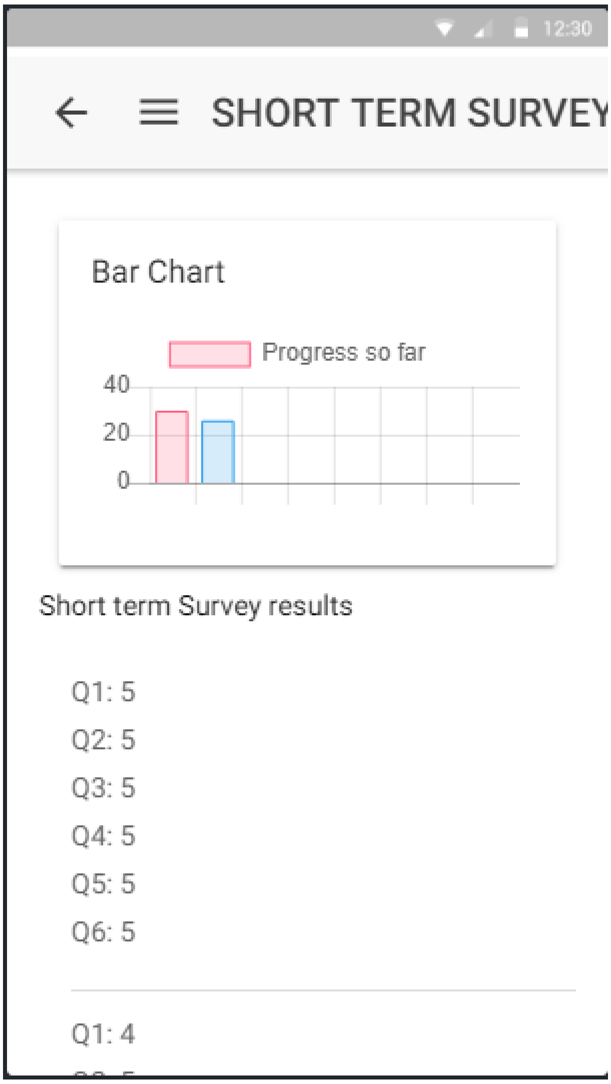
3.4 Profile page



Groceries

2019-03-28

Bread, milk, lettuce

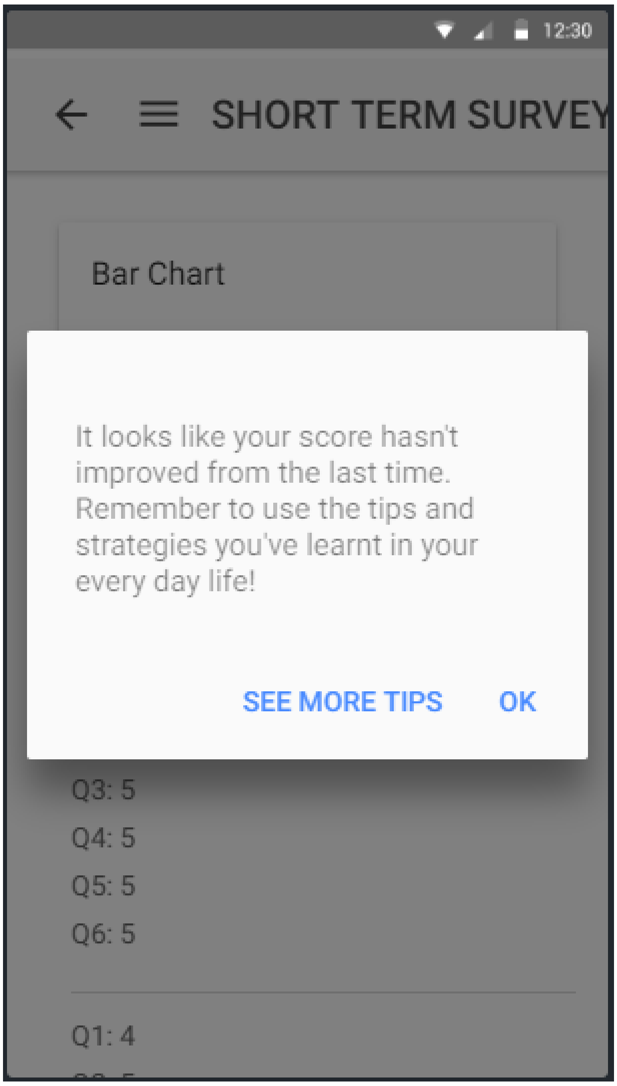


Short term Survey results

Q1: 5
Q2: 5
Q3: 5
Q4: 5
Q5: 5
Q6: 5

Q1: 4

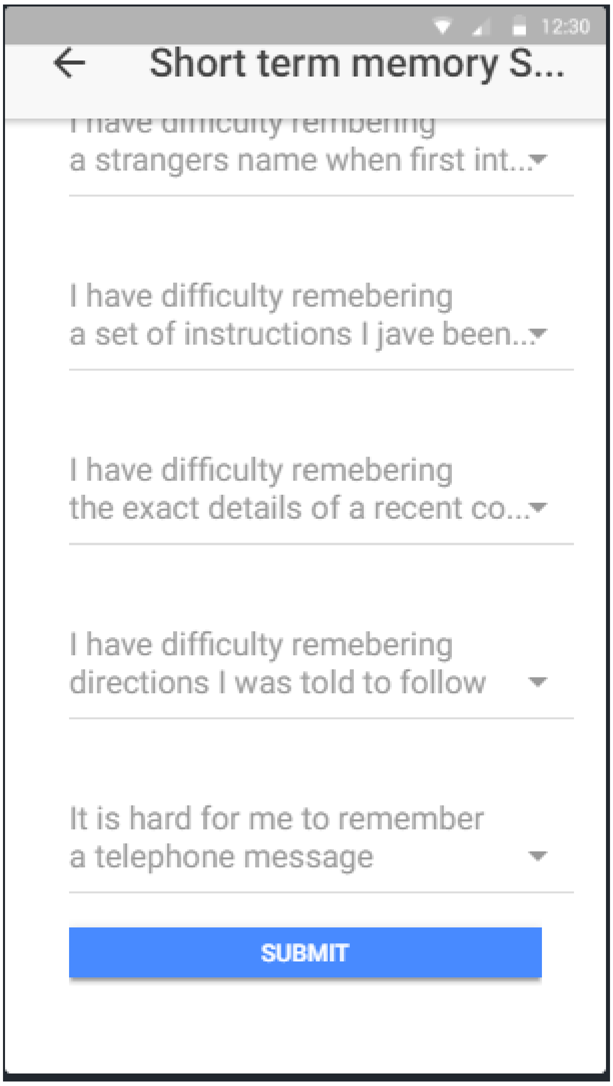
3.5 3.6 Memory survey page



It looks like your score hasn't improved from the last time. Remember to use the tips and strategies you've learnt in your every day life!

Q3: 5
Q4: 5
Q5: 5
Q6: 5

Q1: 4



I have difficulty rembering
a strangers name when first int...

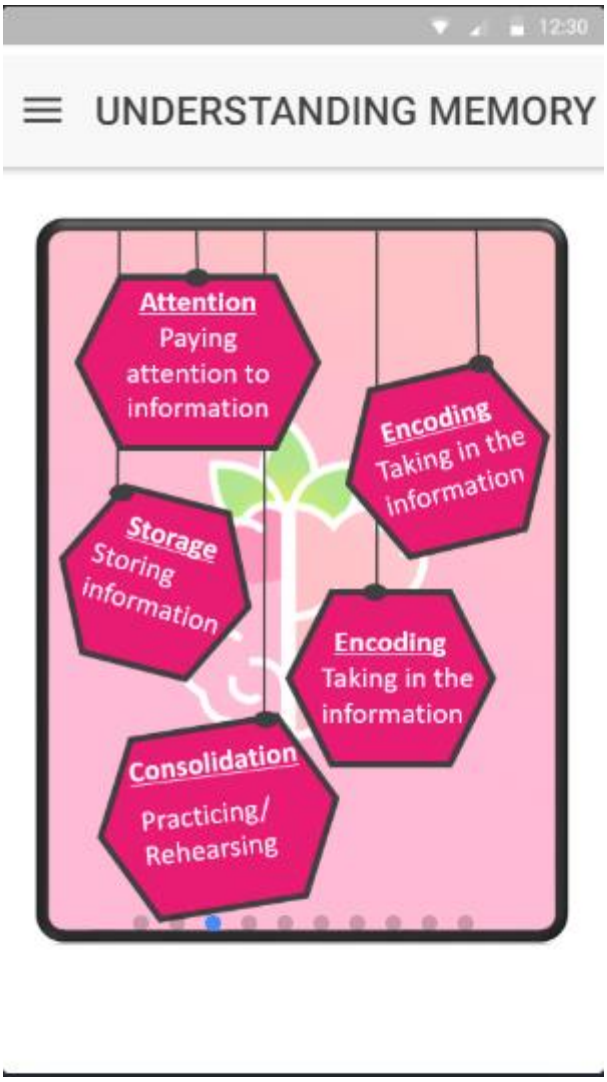
I have difficulty remebering
a set of instructions I jave been...

I have difficulty remebering
the exact details of a recent co...

I have difficulty remebering
directions I was told to follow

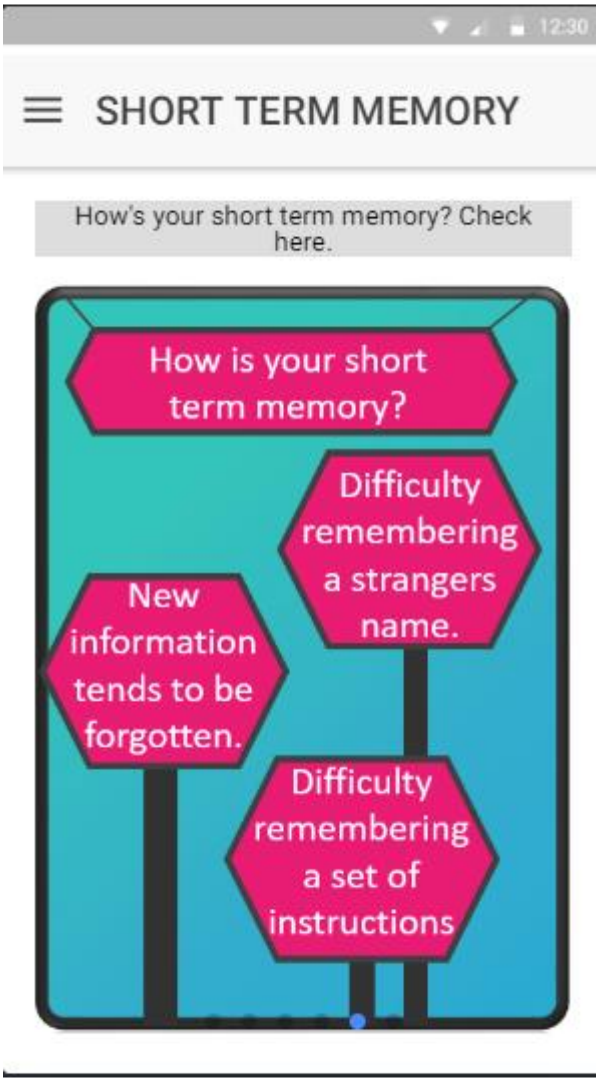
It is hard for me to remember
a telephone message

3.7 memory education page



UNDERSTANDING MEMORY

3.8 Short term page

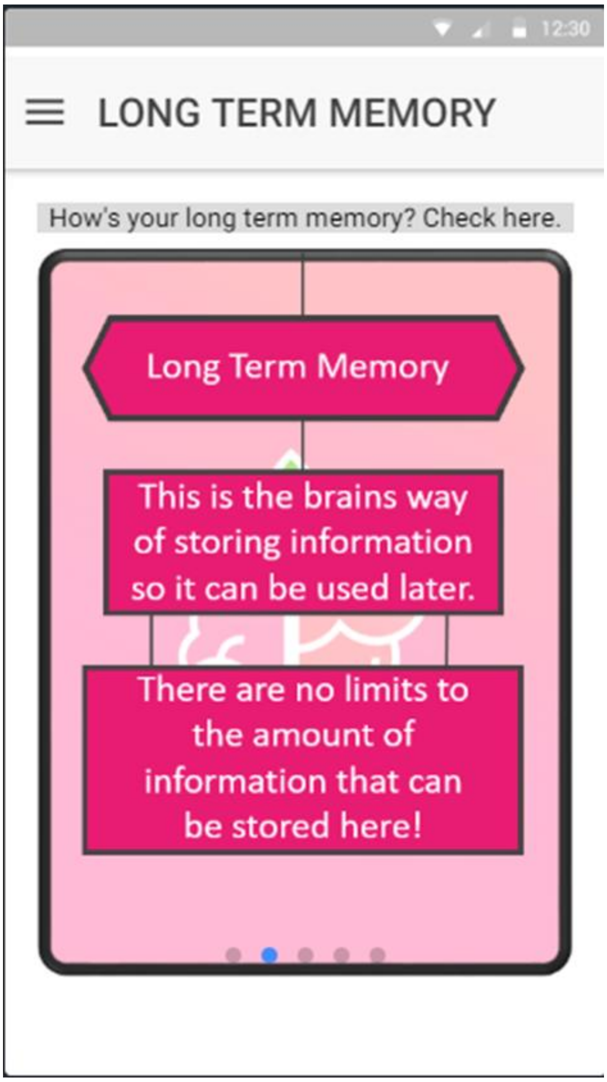


SHORT TERM MEMORY

How's your short term memory? Check here.

12:30
How is your short term memory?
Difficulty remembering a strangers name.
New information tends to be forgotten.
Difficulty remembering a set of instructions

3.9 Long term page



12:30

LONG TERM MEMORY

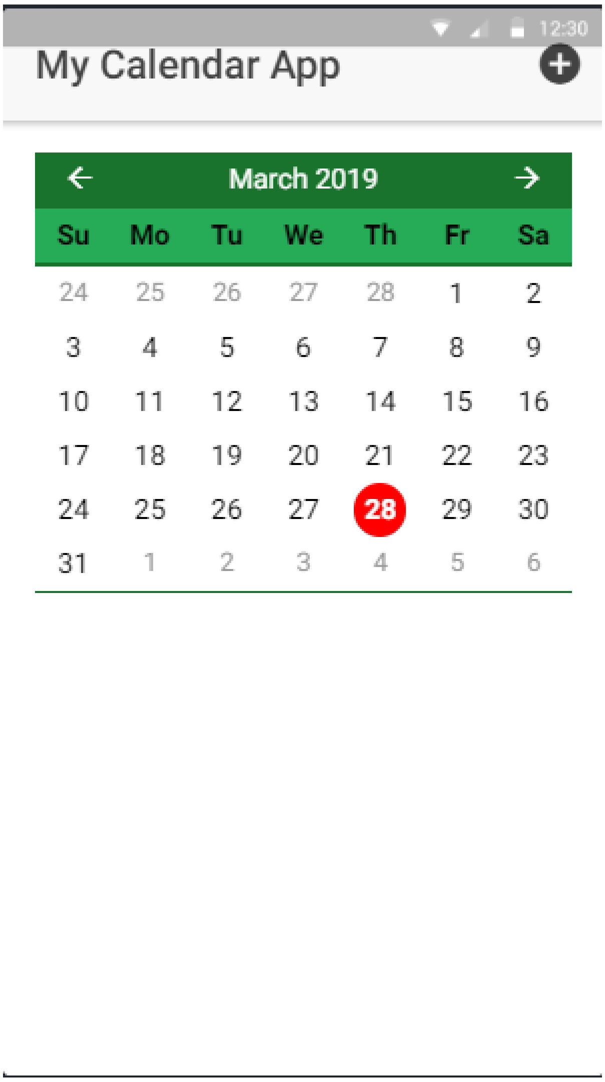
How's your long term memory? Check here.

Long Term Memory

This is the brains way of storing information so it can be used later.

There are no limits to the amount of information that can be stored here!

3.10 Calendar page



12:30

My Calendar App

March 2019

Su	Mo	Tu	We	Th	Fr	Sa
24	25	26	27	28	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6