



MemoryCare

Project Engineering

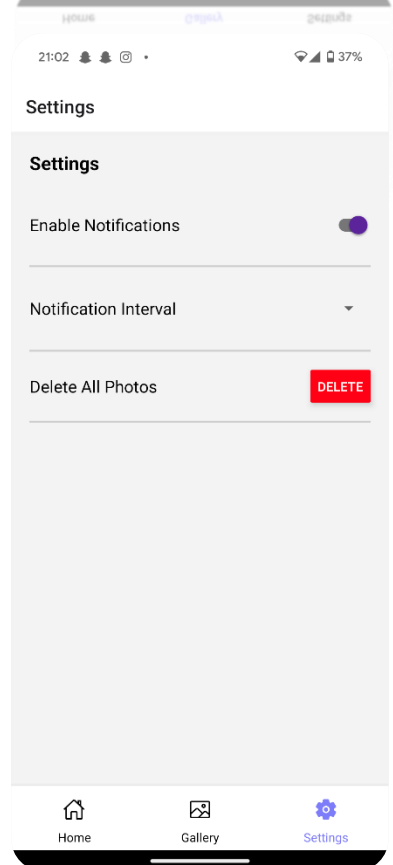
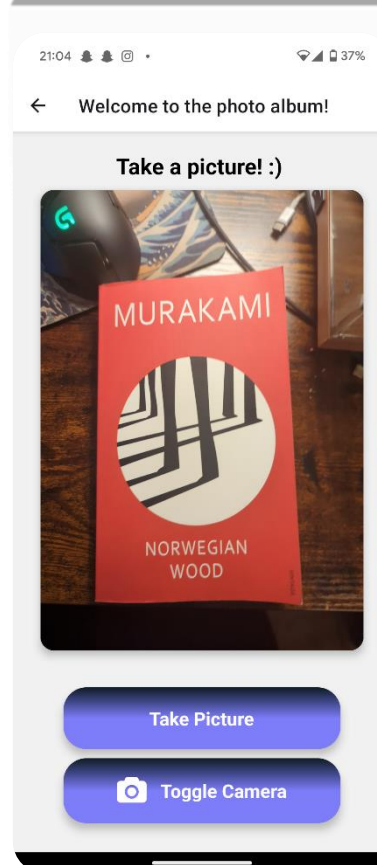
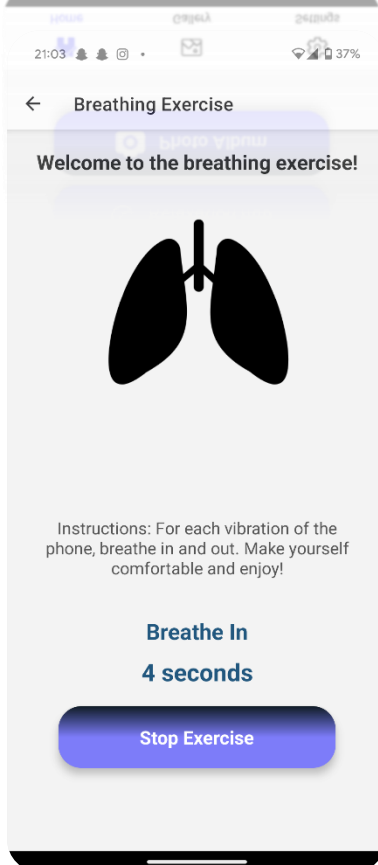
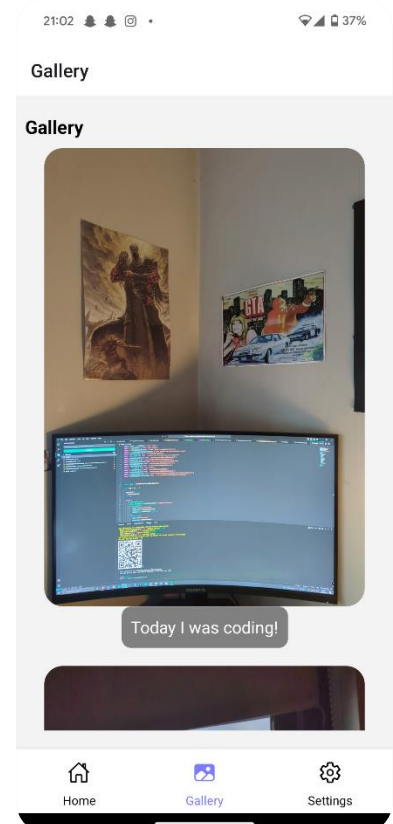
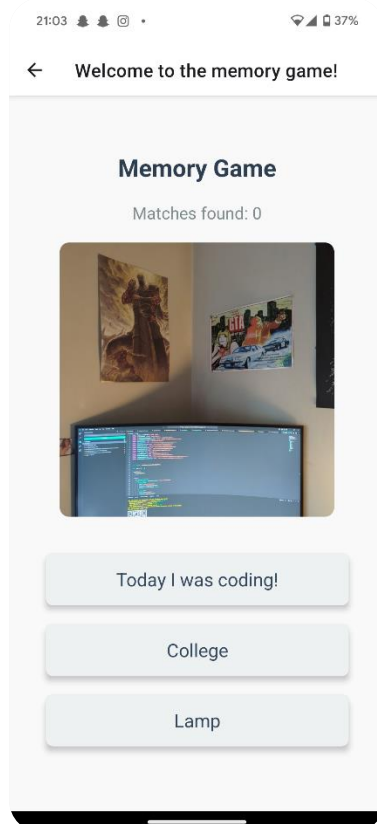
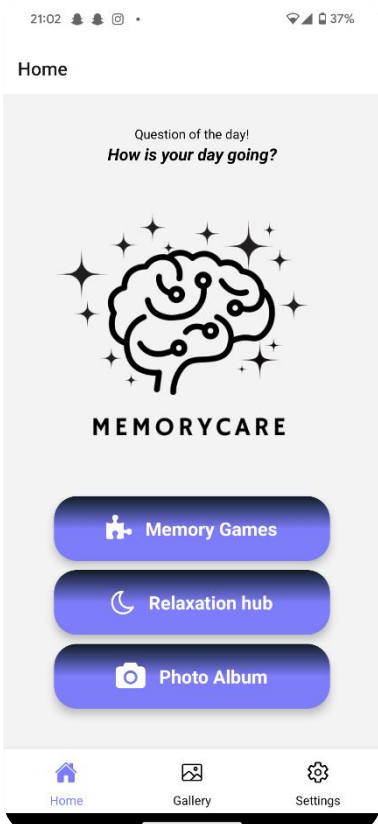
Year 4

Nathan Perry

Bachelor of Engineering (Honours) in Software and
Electronic Engineering

Atlantic Technological University

2023/2024



Declaration

This project is presented in partial fulfilment of the requirements for the degree of Bachelor of Engineering (Honours) in Software and Electronic Engineering at Galway-Mayo Institute of Technology.

This project is my own work, except where otherwise accredited. Where the work of others has been used or incorporated during this project, this is acknowledged and referenced.

Acknowledgements

Brian O'Shea – Project Supervisor

Niall O'Keeffe - Lecturer

Michelle Lynch – Lecturer

Paul Lennon – Lecturer

Patrick Hession – Classmate

Robert Muldoon – Classmate

Friends and Family for testing

Table of Contents

2	Introduction	Error! Bookmark not defined.
3	Poster	9
4	Research and Questions	10
5	Project Architecture Diagram	13
6	Project Plan	14
6	Functionality	15
6.1	Memory Game Logic and Code	Error! Bookmark not defined.
6.2	Delete Photos Function	Error! Bookmark not defined.
6.3	Camera Code	16
6.4	Bottom Bar Navigation	17
7	Ethics	18
8	Conclusion	19
10	References	20

1 Introduction

The goal of the project:

The app had to support people facing memory difficulties and needed to have a focus on innovation, minimalism, and the enhancement of cognitive function and mental well-being. The application wanted to be user friendly and primarily aimed at helping users with memory loss. It needed to have features that were fun, engaging and beneficial for the mind, an app where relaxation techniques can be practiced can where mental health can be managed.

The scope of the project and important features:

The scope of MemoryCare was to develop and integrate a range of functionalities specifically chosen to aid memory enhancement and relaxation. The application introduces a Daily photo mode, which prompts the user to capture daily experiences and type in a text prompt with every picture, a relaxation Hub that provides a serene space for mental relaxation that has two options, such as the breathing Exercises, to help manage stress and a meditation section which allows you to pick between a selection of audio tracks and to improve mental clarity.

The approach to the project:

Approaching the project with a user focused design philosophy by making a sleek and minimalist interface that facilitates ease of use was key to make the application suitable and else overwhelming for those with memory difficulties and for the broader audience, as well as making it being notification based to give constant reminders to the user to interact as it was an important feature when dealing with that demographic.

The main methods & technologies used:

MemoryCare's development integrated a blend of methods and advanced technologies. Visual Studio Code served as the IDE, being the main hub for the coding process with its powerful features and extensions. The core development was helped by the React Native framework, which was chosen for its cross-platform capabilities and allowing deployment on multiple device types. JavaScript was the coding language used and brought the app to together and helped when trying to offer a responsive user interface.

On an android device and on Expo Go app, testing was made a lot easier and a lot more responsive. This ensured that the app was suitable and functional for the Android platform. GitHub helped maintain the workflow and allowed me to go back to previous builds of the app. Microsoft Teams was chosen for communication with other students by doing log updates weekly to note down our progress. Canva was the tool of choice for making the app visually appealing and provided an intuitive interface for an easy workflow.

AWS S3 buckets were used for storing and fetching the audio tracks making it an important component of the Meditation section of the app. Local data storage was chosen as a suitable database as they offer a lightweight and powerful solution for storing user data like photos directly on the device.

Local data storage was chosen as a suitable database as they offer a lightweight and powerful solution for storing user data like photos directly on the device.

What was accomplished:

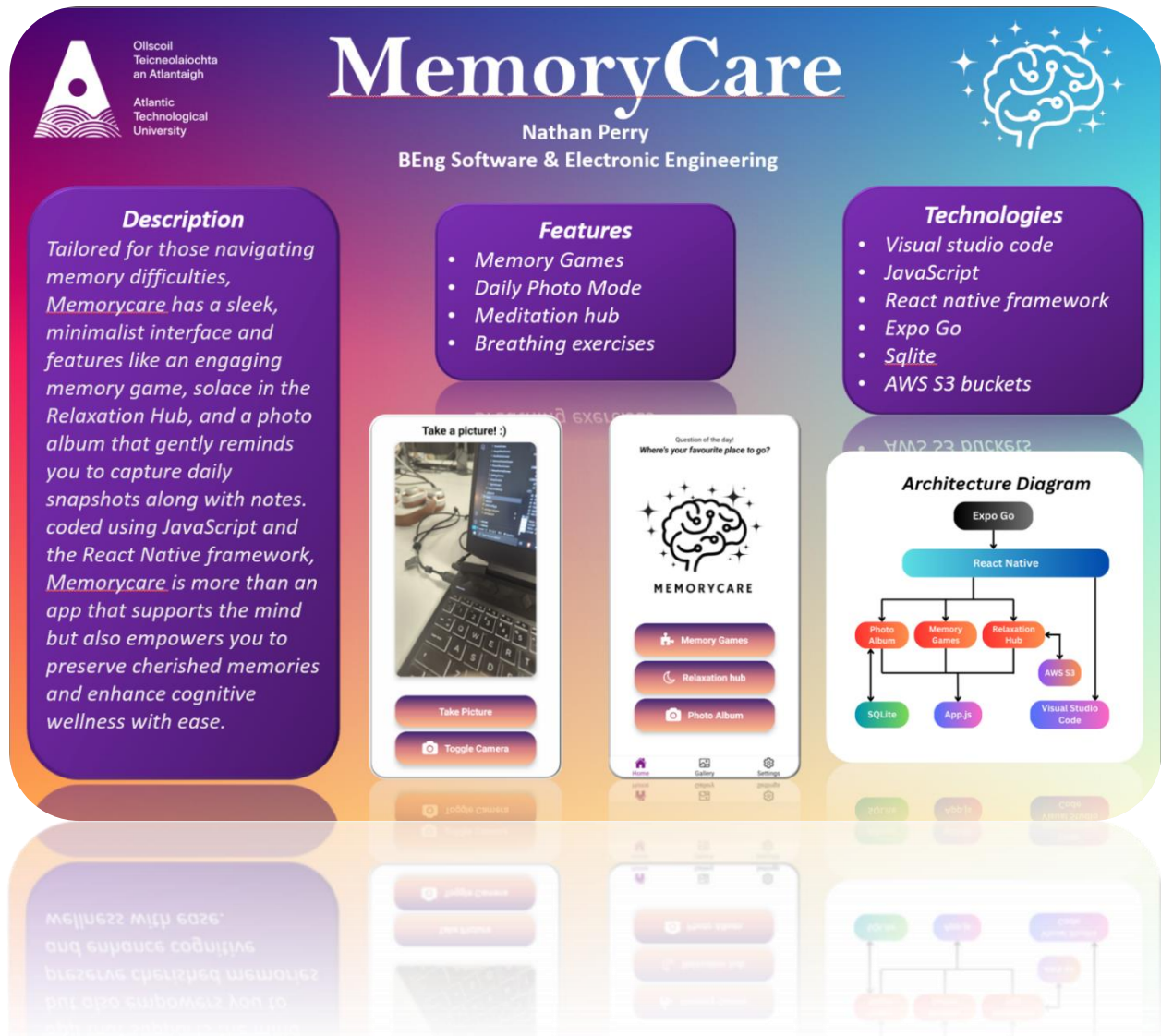
An app that is in a working state, feature complete and intuitive. One of the features being the memory game which utilized photos stored in the SQLite database to keep people engaged in the cognitive exercise. A meditation feature which fetches audio tracks stored in AWS S3 buckets, providing users with a serene space to relax in.

Another accomplishment is the innovative Daily Photo Mode, which prompts users to capture and reflect on daily moments with photos and text, fostering a habit of returning to the app. This is complemented by a notification system, that gives reminders at user-defined intervals that can be altered in the settings tab. Another feature in the app is the Breathing Exercise, which employs device vibration to guide users through breathing techniques. A gallery was also added which allows users to view their photo. For security there is the expo-local authentication system that the user is greeted with upon startup, ensuring their personal data remains secure and private.

Conclusion:

With the features in the finished app, MemoryCare achieved its objective of creating an environment that people needing memory assistance can find solace in and hopes to be an daily routine for the users.

2 Poster



3 Research and Questions

Why did I choose to make an app?

The decision to create an application was initially chosen when understanding of the target demographic. Recognizing that those experiencing the early stages of dementia or memory loss, are familiar with the use of smartphones due to their accessibility and popularity. it was evident that an app would offer an accessible means for them to engage with the memory assistance. Also, the opportunity to simultaneously learn at home and apply knowledge from a mobile app development module made it a lot more possible to create this kind of application.

What research I did to understand the user base?

the needs of its users are a fundamental part of MemoryCare, I investigated the challenges commonly faced by those with memory problems. Research revealed that not much action is taken in the early stages of conditions like dementia or general memory impairment. It was also clear that therapy techniques emphasizing general well-being, meditation, and repetitive memory exercises were beneficial yet not sufficiently practiced. This insight showed the importance of these kind of features. It was also important in what type of design should be within the app. A minimalist look and functionality would avoid overwhelming users who might struggle with processing too much information simultaneously.

What research should I do to get started on the app?

Research into mobile app development through various Youtube, as well as how to set up the environment for a react native app were pieces of research that had to be learnt before commencing development.

What features should be in the app?

In the early stages of research I considered a social media aspect to foster a sense of community in the app among users with similar experiences and enable family encouragement. However, grounded in research and reflecting on the core needs of well-being, meditation, and

memory practice, I ultimately centered the app's features around these areas. This was informed by the understanding that such functionalities could offer direct therapeutic benefits and encourage consistent use without adding complexity.

What database should I use?

Firebase and SQLite were potential options I was considering in the early stage of development. Firebase offered real-time data, easy to use SDKs and in some ways would make for a better application, but when considering privacy, the user base and the need for a constant internet connection made it less suitable for the app's purposes. SQLite was the better choice for this purpose, as it stores data directly on the device and keeps all that data local. This feature made the app faster, more secure and a better fit for the daily photos feature.

How does the user sign in?

When designing the sign-in process, it was important to consider the specific needs of the user, particularly those with memory difficulties. A traditional username and password approach was not suitable due to potential memory challenges. I decided to go with Expo's local authentication method, which allows for biometric options like fingerprint. This choice simplifies the login process, improves user accessibility and ease and makes testing and integration during the app's development phase a lot easier.

How do I test the app?

Testing the app particularly on Android was carried out on Expo Go, which is known to be widely used and easy to understand. This tool was especially beneficial as it aligned with one of my academic modules Mobile App Development, allowing me to improve my understanding with the platform. Expo Go provides a real-time preview of the app directly on a mobile device. This setup ensured an effective testing phase for MemoryCare.

What is good practice for making an app?

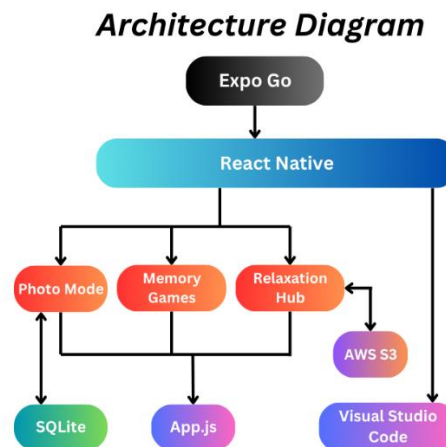
Good programming practices when making an app can significantly enhance both the efficiency of the development process and the quality of the final product. Project organization of the project files is important and would typically involve structuring the project into folders such as components, screens, and utilities. This organization makes it easier for me to find files and made the project side bar look a lot easier on the eyes.

Testing the app on an actual phone using Expo Go was also essential to understand how the app looks and works. This helped monitor performance and the UI, making bugs less likely to happen. GitHub aided in maintaining the code and reverting to earlier versions if something goes wrong. Implementing console logs throughout the development process is an effective way to find errors in the code and an excellent for real-time debugging.

What react native features should I be utilizing?

I adopt modern React features like components and utilize hooks like useState and useEffect. This aided in creating easier to use code by emphasizing the separation of concerns and a more direct approach to UI development. Having well-designed navigation was crucial for the development of this project and needed to be learnt and almost mastered to ensure the app ran smoothly a user-friendly experience while navigating through the app.

4 Project Architecture Diagram



The diagram above shows how MemoryCare is structured. Expo go is used for testing, simulating the app and running the framework, which is used in the visual studio code IDE, the environment used for writing and editing the code. React native is the base technology that enables the development of the main components. Them being the photo mode, memory games and the relaxation hub. The photo album sends and fetches data like pictures and text to and from the local SQLite database. The relaxation hub interacts with AWS S3 buckets that is used to store and stream audio files (specifically the meditation screen). The App.js is the entry point file for the application. Being connected to all the main components means it's responsible for bootstrapping and rendering these features on the device and displaying it on Expo Go.

5 Project Plan

Project planning was done through two main sources, one being the Microsoft teams I did during the first semester which consists of updates on the project's progression, research, new technologies used and audio logs between me and my classmates about how much we got done in the past week. Another source was Jira software which was convenient and useful during the first development stage of the project.

<https://nathanperry057.atlassian.net/jira/software/projects/KAN/boards/1?atlOrigin=eyJpIjoYUQ5NDFiYjcxMThmNGEwN2FkZTY4ZTNjMzk4NTc1OWEiLCJwIjoiajI9>

6 Functionality

6.1 Memory game logic and code

```
export const shuffleArray = (array) => {
  for (let i = array.length - 1; i > 0; i--) {
    const j = Math.floor(Math.random() * (i + 1));
    [array[i], array[j]] = [array[j], array[i]];
  }
  return array;
};
```

A challenge found when making the memory game component was finding efficient ways to shuffle the photos in the SQLite database. This was achieved with the Fisher-Yates shuffle algorithm which was found on stack overflow [2]. The algorithm works with an *i* in the last index of an array after each iteration it decrements by 1 and continues until its greater than 0. With the `math.random` function a random index between 0 and *i*. Its then multiplied by (*i* + 1) to the correct range. A destructing assignment is then used to shuffle the array and changes the probability so it'll appear in any position.

```
const fetchItems = () => {
  db.transaction(tx => {
    tx.executeSql(
      'SELECT * FROM photos;',
      [],
      (_, { rows }) => {
        const data = [];
        for (let i = 0; i < rows.length; i++) {
          data.push(rows.item(i));
        }
        const shuffledData = shuffleArray(data);
        setSelectedImage(shuffledData[0]);
        setTextOptions(shuffleArray(shuffledData.slice(0, 3).map(item => item.text)));
      },
      (t, error) => {
        console.log('Error fetching photos:', error);
      }
    );
  });
};
```

This shuffle array function is then imported into my memory game screen and used in the fetch items function which interacts with a database to fetch items by calling `db.transaction`. Inside this transaction, with `tx.executeSql` a SQL query is executed and selects all records from the

photos table and if successful the callback receives the results. An empty array holds the photos. `shuffleArray(data)` is then called to randomly shuffle these items. It then takes the first three elements of the shuffled data and maps them to their text property, it will then shuffle them again using `shuffleArray`. The `setTextOptions` will provide randomized text options related to the images, this is then used to display the randomized image and the three text options.

6.2 Delete Photos function

```

96   const deleteAllPhotos = () => {
97     db.transaction(tx => {
98       tx.executeSql(
99         'DELETE FROM photos;',
100        [],
101        (_, { rowsAffected }) => {
102          if (rowsAffected > 0) {
103            Alert.alert('Delete successful', 'All photos have been deleted.');

```

6.3 Camera code

```

const takePicture = async () => {
  //Camera permission
  if (!hasPermission) {
    Alert.alert('Error', 'Camera permission not granted.');

```

When the app is first loaded into this screen it will request for camera permission

Upon component mount, the screen requests camera permissions. If granted, the `cameraReady` state is set to true, allowing the user to take photos. The camera can then be used and toggled

between the front and back with the `toggleCameraType` function. Once a picture is taken the `takePictureAsync` function is called and captures the URI, sets off a vibration and brings up a modal that prompts the user to enter a description.

6.4 Main Container for Bottom Bar Navigation

[1]

```
src > JS MainContainer.js > MainContainer > <function>
1 import * as React from 'react';
2 import { createBottomTabNavigator } from '@react-navigation/bottom-tabs';
3 import Ionicons from 'react-native-vector-icons/Ionicons';
4 import 'react-native-gesture-handler';
5
6
7 //Screens
8 import HomeScreen from './screens/HomeScreen/HomeScreen';
9 import DetailsScreen from './screens/DetailsScreen/DetailsScreen';
10 import SettingsScreen from './screens/SettingsScreen/SettingsScreen';
11
12
13 // Screen names
14 const homeName = 'Home';
15 const detailsName = 'Gallery';
16 const settingsName = 'Settings';
17
18 const Tab = createBottomTabNavigator();
19
20 export default function MainContainer(){
21   return (
22     <Tab.Navigator
23       initialRouteName={homeName}
24       screenOptions={({ route }) => ({
25         tabBarIcon: ({ focused, color, size }) => {
26           let iconName;
27           if (route.name === homeName) {
28             iconName = focused ? 'home' : 'home-outline';
29           } else if (route.name === detailsName) {
30             iconName = focused ? 'image' : 'image-outline';
31           } else if (route.name === settingsName) {
32             iconName = focused ? 'settings' : 'settings-outline';
33           }
34           return <Ionicons name={iconName} size={size} color={color} />;
35         },
36         tabBarActiveTintColor: '#7d7cf9',
37         tabBarInactiveTintColor: 'black',
38         tabBarLabelStyle: { paddingBottom: 5, fontSize: 13 },
39         tabBarStyle: { padding: 5, height: 70 },
40       })
41     )
42   );
43 }
44 <Tab.Screen name={homeName} component={HomeScreen} />
45 <Tab.Screen name={detailsName} component={DetailsScreen} />
46 <Tab.Screen name={settingsName} component={SettingsScreen} />
47 </Tab.Navigator>
48 );
49 }
```

MainContainer.js is the central navigation structure of my React Native app using `createBottomTabNavigator` from React Navigation. It organizes three main screens that can be navigated to using the bottom navigator. Each with icons for aesthetic appeal.

7 Ethics

MemoryCare being a mobile application aimed at helping people with memory challenges, it was understood before and during the development that many ethical considerations must be addressed to make sure the project upholds high standards of respect for its users. The app manages sensitive personal data, such as photos and possibly health-related information depending on the text the users add with said photos, which requires a suitable way of storing this content. This mindset and view on the app made local storage using SQLite a suitable choice.

Accessibility was a core approach to designing the app. Designing an interface that is easy to navigate for all users is crucial. This involved having clear, large fonts, suitable colour schemes, and intuitive navigation. It is important not to frustrate them with complex functionalities and a hard learning curve. It is also very important to make sure that users have control over their data and the ease of use when interacting with it. Features like the ability to turn on and off notifications, having customizable notifications intervals, the ability to delete photos from the gallery, and a local authentication system made for a suitable range of control for the user. Through these ethical decisions, MemoryCare aims to enhance the lives of its users in a dignified and respectful manner.

8 Conclusion

In conclusion, the overall journey of researching, developing and problem solving with this project has been a very educationally beneficial experience for me. I was able to put in a great amount of passion and the process of programming it challenged my creativity.

The project allowed me to explore wider aspects of software engineering, user interface design, understanding the importance of a user base and the researched needed to implement features that would cater to the users. Each problem that came up served as a major learning experience unlike any other project I've undertaken and gratification for any milestone or challenge.

It's also very rewarding to know it's an app that's trying to solve a problem in area like memory loss or dementia and hope it gives some bit of solace in their days with through meditation, breathing and cherished photos. The overall project would be a standard for me that I'll try to achieve for every future project I plan to take on.

9 References

[1] Bottom Navigation Bar Tutorial in React Native by Indently

Link -<https://www.youtube.com/watch?v=1kSLd9oQX7c>

[2] Stack Overflow [Online].