



Faculty of Engineering & Applied Science

Mobile App. Development Final Project:

Mood Tracking App

FroggyFeel

Final Report

Kinjal Shah	100743551
Jessica Leishman	100747155
Daniel Kamel	100754671
Jachimike Ezenwamadu	100783161

Github Repository: https://github.com/kinjals0130/MoodTracker_Application

Background

In today's world, mental health is a serious issue. Close to 1 in 4 people will experience a diagnosable mental health condition in their lifetime while others may suffer mental health issues that may not be as critical but still concerning [1]. This is a huge concern as the effects of these afflictions are detrimental to the people suffering from it and those around them which is why we need tools and technologies that can help people overcome their mental health problems. The use of mobile apps for promoting health features has grown in the past few years, causing people to be more invested in them so that the applications can be refined to accommodate all the various mental health conditions that currently exist [2].

Introduction

The android app that has been designed is a Mood Tracker called FroggyFeel. The purpose of the app is to help users monitor and manage their emotional well-being by allowing them to record and analyze their moods over time. The application provides a user-friendly interface for users to select a themed emoji indicating their current mood along with additional details such as the date, title and description of their mood and a voice note to indicate factors that may be influencing their mood. The collected data is visualized on the home and past notes screens, offering users insights into patterns that may affect their mental health. This fosters awareness of emotional patterns and provides users with information of things that may trigger them. This can empower users to make informed decisions about their lifestyle choices.

Details of Project

System Architecture Design

The diagram below, Figure 1, displays our system architecture design. The design serves as a starting point and blueprint for the design of our app as it shows the way the user interacts with the app, along with how the components all interact with each other. The database plays an important role in our architecture as it stores all the journal entry information that the user enters in both the "New Entry" screen and the "Edit Entry" screen, and it is being retrieved in both the "View Past Entries" and in correspondence the "Edit Entry" screen as well. It also stores the pin for private notes for later access.

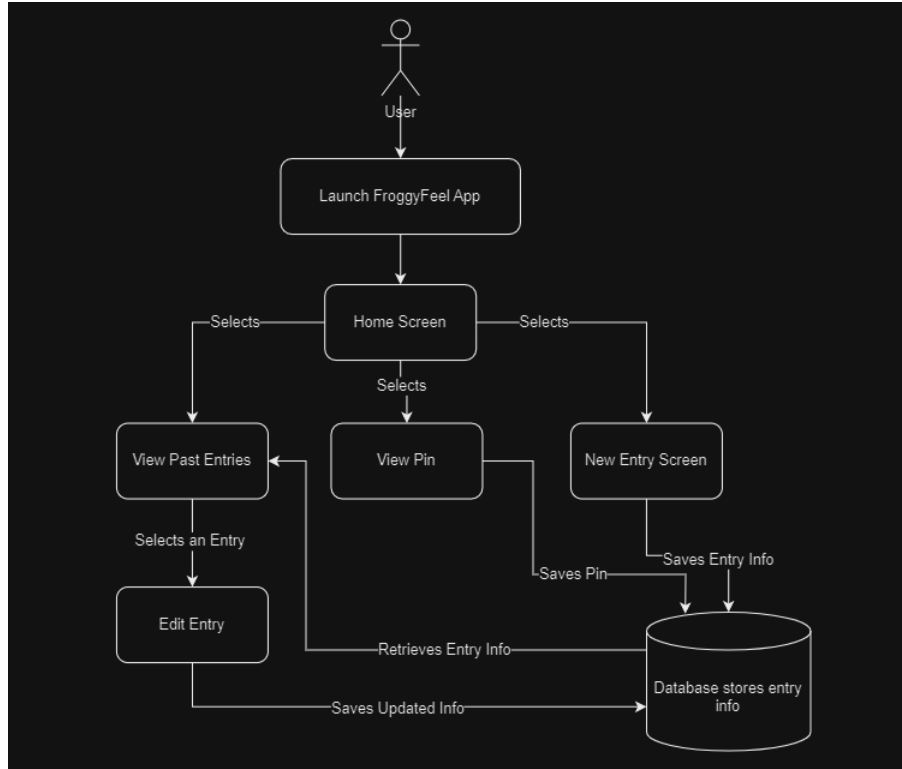


Figure 1: System Architecture Design

Easy to Use and Interactive UI

FroggyFeel is designed to be an easy to use app for users of all ages. The app has a cute and interactive frog theme, where users can click on a frog that represents or reflects their mood that day. On the main landing page, as shown in Figure 2, the user is welcomed to the app and they can easily interact with the frog emotion rating. When the user selects a frog emotion, they are displayed with a message with the mood they are feeling. The user's heat map of how they have been feeling over a while is displayed on the main page as well. Users can create a new entry or if they select the menu shown in Figure 3, they can view past entries or change or set their pin.

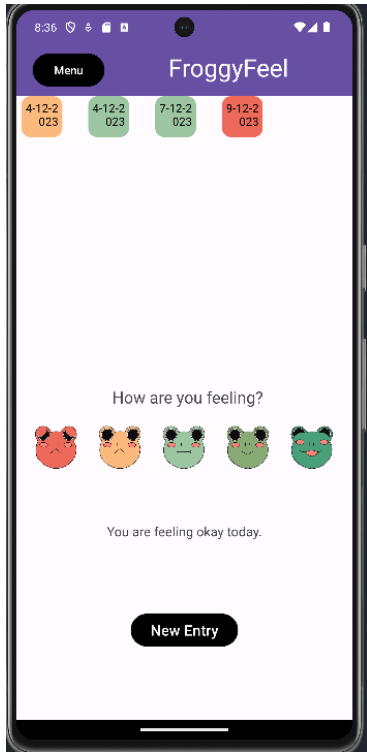


Figure 2: The Home Landing Screen

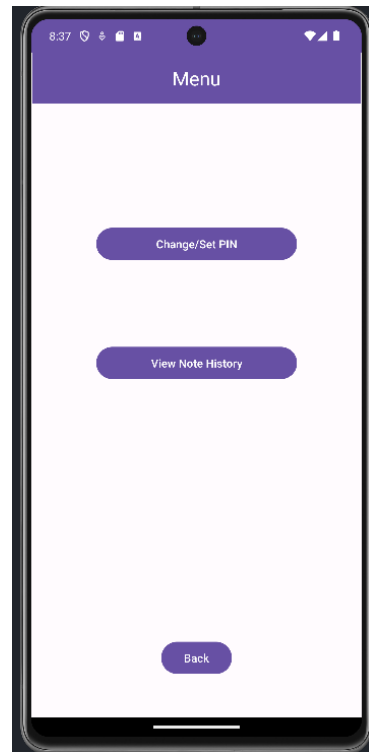


Figure 3: The Menu Screen

If the user selects to create a new entry, they are redirected to the entry page, Figure 5, where they can log how they are feeling. The user is prompted to choose a frog that best represents their mood that day and the entry colour changes with the mood they choose. The user can then pick a date with a drop down date picker, Figure 4, for more efficient entry of a date and so we can guarantee it is stored properly in the database as well.

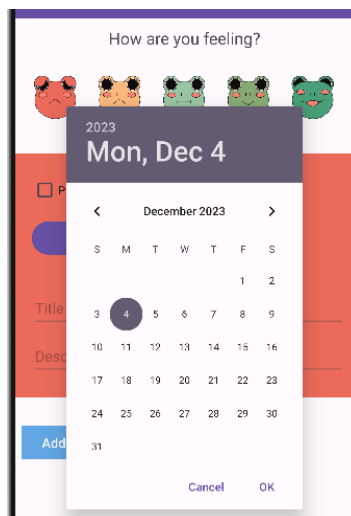


Figure 4: Date Picker

The next step is for the user to enter a Title and Description for their entry, this allows for the user to be able to log how they are feeling. Writing out emotions may be a useful tool for the majority of users, especially if they are going through a rough time as it can help them be in control or gain understanding of their emotions and feelings. Users can also record a voice note if they decide not to write it out. Talking out the way they are feeling may also be a useful outlet to some of the users as they can vent their feelings and have it out of their system. Lastly, users are able to make the note private, in this case they would have to create or enter a pin to view the private note. Having private notes gives the user a sense of protection and privacy, knowing that their feelings and emotions are only for themselves. The user can then discard their entry by selecting the back button or can save their note by clicking the save button.

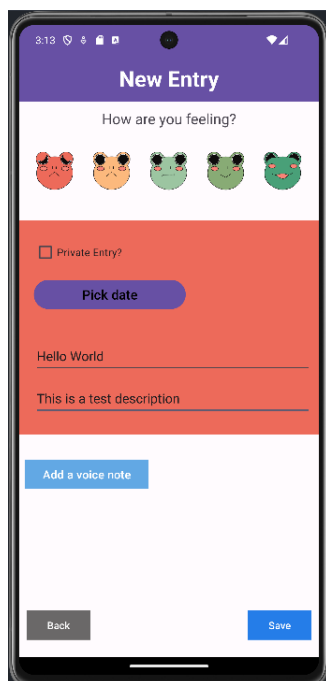


Figure 5: New Entry Screen

Once the user saves the note, they would be able to access this note from the menu selection “View Past Entries”. On this screen, shown in Figure 6, the user is able to see a quick overview of their entry, can delete an entry, or if they select an entry they are able to edit. The edit screen contains the title, description, date and all that they entered when making the journal entry. From this screen they are able to view the location they made the journal entry from, along with viewing the saved voice note.

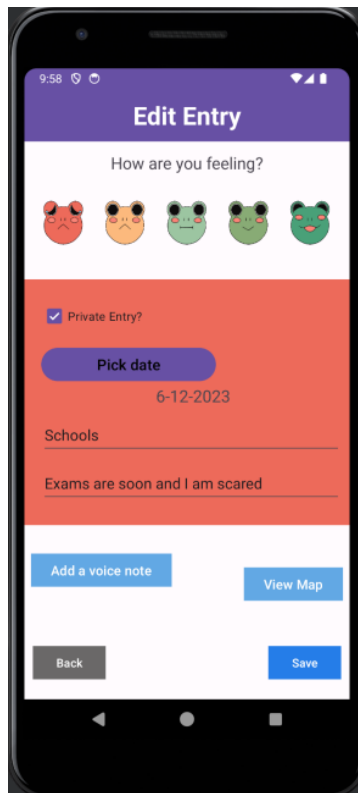


Figure 6: Edit Entry Screen

The goal was to create an app that is well designed to be easy to use and interact with. Since the target audience can vary in ages, it is important to ensure that young users may be able to independently navigate the app without parental guidance.

Authentication of Users

FroggyFeel offers the feature of private notes. When users click on the menu button, it switches to the menu page and shows two buttons. One of them is to enable users to change/set the pin. If users want to change their current pin then they select the change/set pin button and after that they are taken to the change/set pin page as seen on Figure 7. If they already have a pin and want a new one then they enter it into the option for current pins and input their new pin into the second option for new pins and click save. This will change the current pin for all preexisting private notes. For users who don't have a pin all they need to do is enter a pin in the option for new pins and click save. Using the new pin option only also deletes notes made with previous pins.

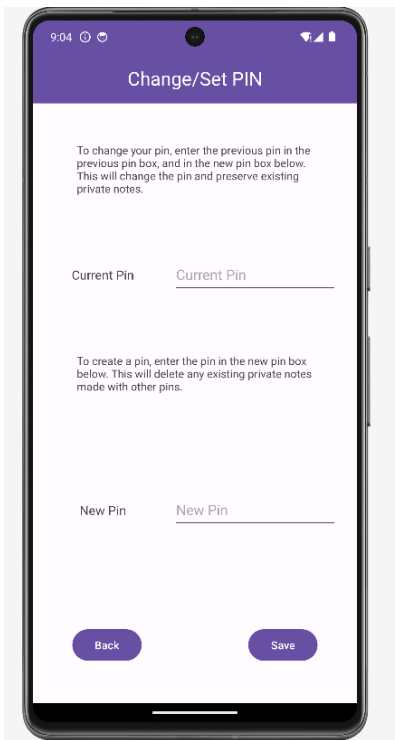


Figure 7: Change/Set Pin Screen

Once the pin has been set, users can make private notes. On the new entry screen there is a private entry option box that can be selected as seen in Figure 8 to make the current entry private. Once it has been selected, users can put in all the details as they would for non-private notes and click save. After that users can navigate to the menu screen and click on the view note history button to see all past notes as seen in Figure 9. All the information on non-private notes is visible to everyone but for private notes, Figure 9 shows that descriptions are hidden and the only thing that is visible is “PRIVATE NOTE” as seen in Figure 10. Users can then click on private notes and enter their pin to see all information.

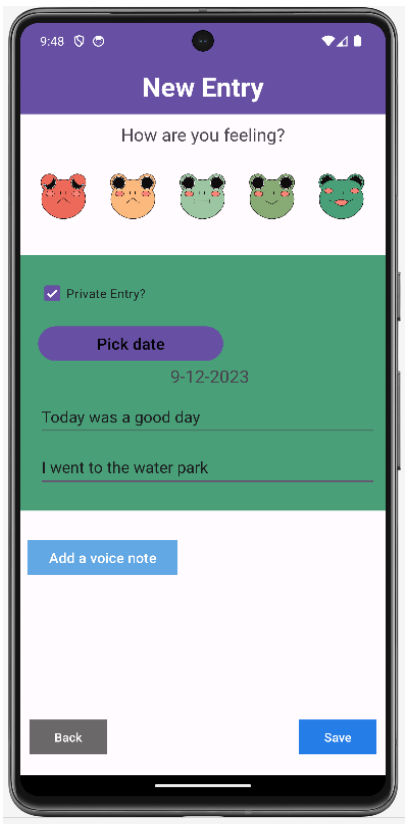


Figure 8: Private Checked

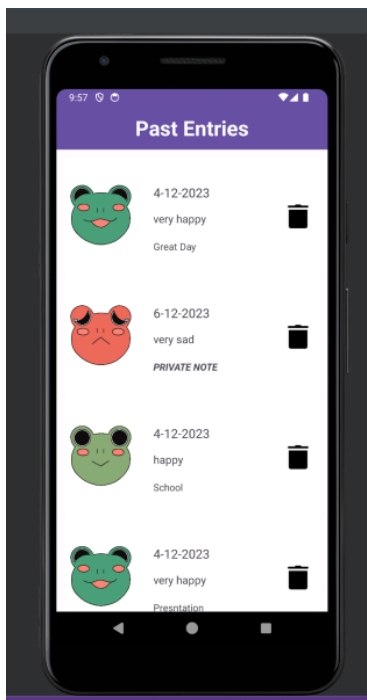


Figure 9: Shows Note with Private Entry

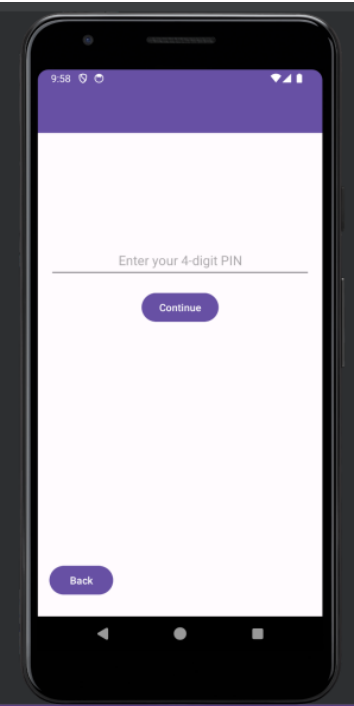


Figure 10: Prompts User to Enter Pin

Use of Maps

The app integrates the use of maps by allowing the user to be able to see the location of where they entered the journal entry from. This could be a good way for the user to keep track of where they were, as they may have had many good days when in a certain place, i.e. traveling, vacationing, visiting someone. Figure 11 shows the prompt that is displayed when the user first opens FroggyFeels on their device, to allow access to use the current location of the user's device. Given the permission, the app will then be able to track the location of where the entry has been created from. Users are able to see this information when they select the note in the "View Past Entries" page.

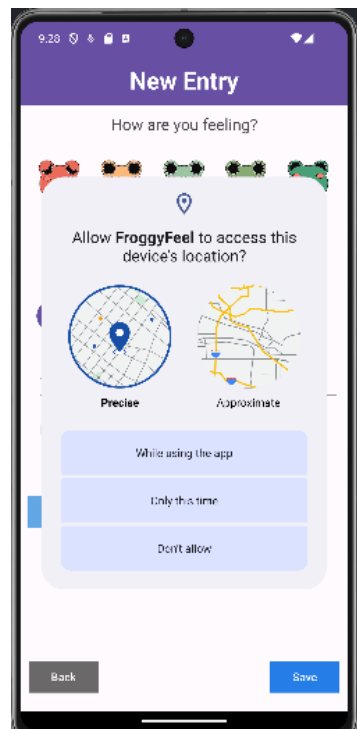


Figure 11: Prompts User for Access



Figure 12: Shows location of Entry Made

Use of Mobile Sensors

This application allows users to save voice notes with their entries. Compared to texts, voice notes provide a more nuanced understanding of a message as they are able to convey the speaker's tone and intonation thereby adding more meaning and context to the entries. As seen in Figure 13, to add voice notes the user would first need to give their app permission to use the

microphone. If access is allowed, on the new entry page the user can add a voice note by selecting the “Add a Voice Note” button. This takes them to a new voice note page, displayed in Figure 14, where the user can record a voice note, press stop once done and can play it back using the play button. When the audio is recording, stopped and playing back, this is displayed to the user so they are aware of the action being taken. The stop button is displayed once the recording has started. Once the user selects save, the voice note is saved in the database to be accessed later on, the user can hear back their voice recording in the “Edit Entry” page.

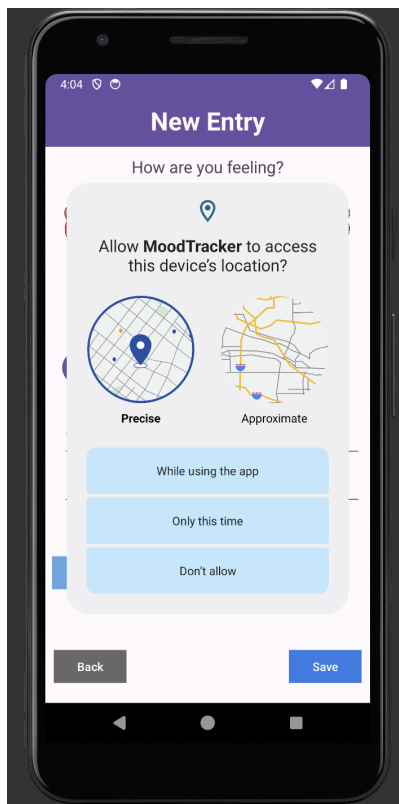


Figure 13: Prompts User for Access

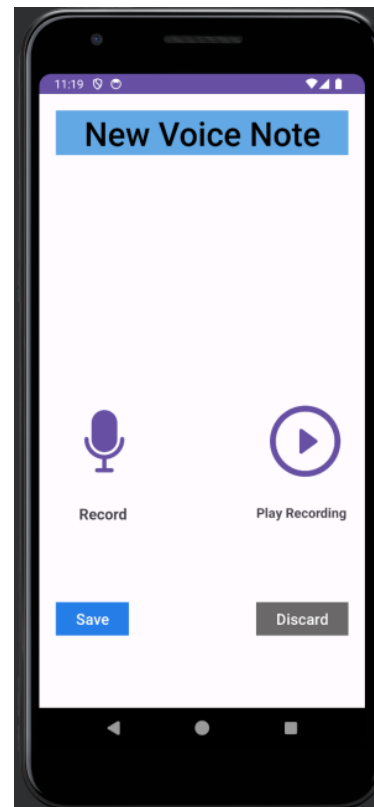


Figure 14: New Voice Note Screen

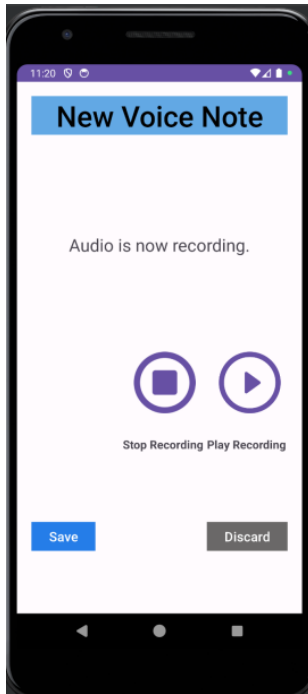


Figure 15: Audio Recording

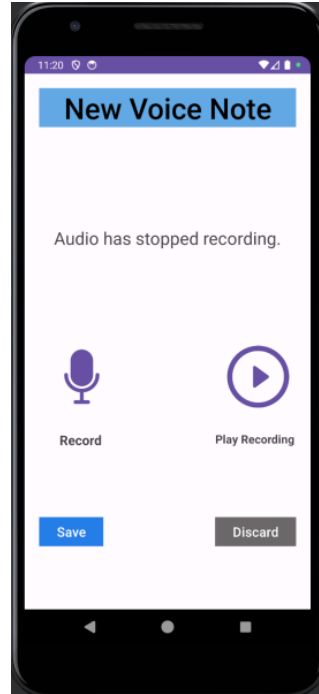


Figure 16: Audio Stopped



Figure 17: Audio Playing Back

Use of Database

A SQLite database was employed to provide database interactions in the android application. SQLite was chosen as it is an open source database that does not run on a separate server, making it effective and efficient to use. With SQLite, multiple databases can be created at once and can be easily accessible by any class in the program.

A class called SQLiteManager which extends SQLiteOpenHelper was used to manage database creation and its version, along with a Model class which held all the getters and setters of the information being passed into the database. As shown in Figure 18, a table called “JournalEntry” was created to store the information from the “New Entry” and “Edit Entry” screens. This table stores the entry’s id, title, emotion, description, date, privacy status, voice note, colour, longitude, and latitude. The second table called “User Pin” stores the information for the PIN used in the case of a private note, storing the id, PIN entry, and the date.

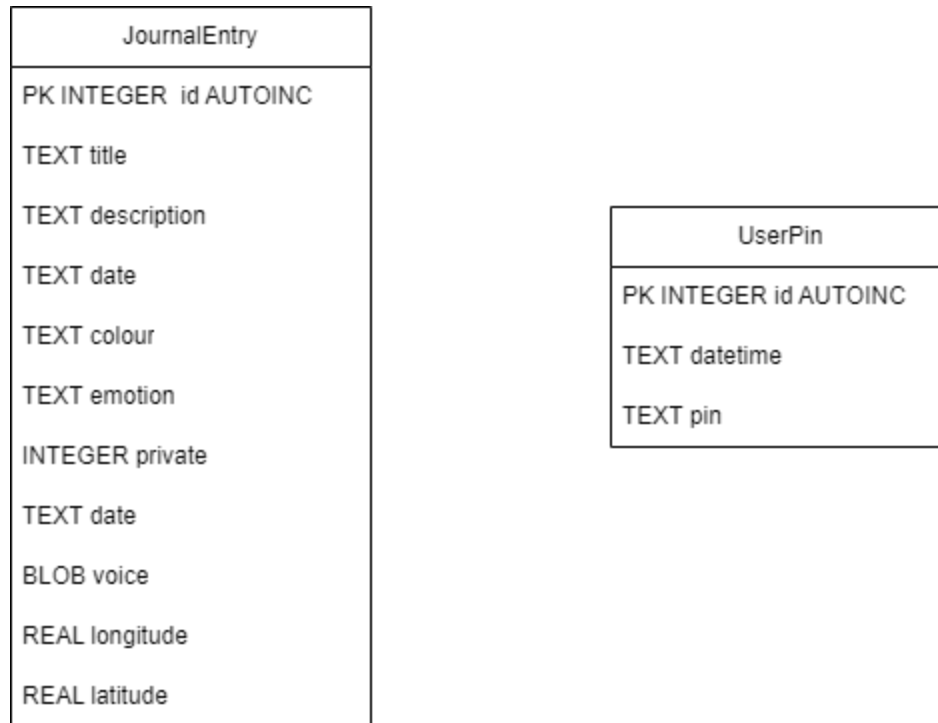


Figure 18: Database Schema Diagram

Implementation Challenges

A number of challenges were encountered during implementation of the application features. FroggyFeel has a number of complicated interactions between many different activities, with data being passed between them through the use of models, or the use of extra data in intents. This information is critical to successfully access the intended note in the past notes view, and to view the associated information with a note – including voice notes. There was some difficulty ensuring edits or updates were being saved to the correct note entry, and coordinating emotion button updates with the edit notes.

Voice notes also presented a significant challenge, as storing the voice notes locally on the device was not a feasible option. Storing the voice notes locally would have created a condition in which any user can listen to what may be private audio. By keeping the audio stored in the database, the voice notes are retrieved when necessary alongside the remainder of the note information needed for display. Playback was also difficult to implement, as null references

created issues where the app would crash if buttons were pressed in an unpredictable order. However, these issues were remedied for the final iteration of the mood tracking application.

Conclusion and Lessons Learned

Ultimately, the app development project FroggyFeel was a success. All the functions outlined within the proposal were implemented with a clear and illustrative user interface. The user is able to easily navigate from screen to screen, and perform the required mood tracking tasks. Fragments were incredibly valuable and used in a number of different ways, as they provided popups for the date picker and permissions request dialogs. Passing information through intents was also an important feature to ensure correct implementation for, as this was necessary to access any database content for editing or deletion. Passing information through intents is also used to display location information on the maps screen, as the latitude and longitude needs to be passed to the map activity for use when it is being created.

It is known that journaling thoughts and feelings can be a good way to understand them more clearly, especially if the user deals with stress, depression and anxiety as it can help gain control of your emotions [3]. All users are encouraged to journal through the use of the app as they can either write it out or talk it out through the voice recordings. Overtime, users may see a trend in their entries and could be motivated to seek further help if they notice a decline in their mood over a constant period of time. FroggyFeel is an app designed in the best interest of the users and aims to monitor and assist users in tracking their mood in hopes they use it as a tool to bring self-awareness.

References

- [1] Schueller, Stephen M et al. "Understanding People's Use of and Perspectives on Mood-Tracking Apps: Interview Study." JMIR mental health vol. 8,8 e29368. 11 Aug. 2021, doi:10.2196/29368
- [2] Caldeira, Clara et al. "Mobile apps for mood tracking: an analysis of features and user reviews." AMIA ... Annual Symposium proceedings. AMIA Symposium vol. 2017 495-504. 16 Apr. 2018
- [3] "Journaling for emotional wellness," University of Rochester Medical Center, <https://www.urmc.rochester.edu/encyclopedia/content.aspx?ContentID=4552&ContentTypeID=1#:~:text=Now%20it's%20called%20journaling.,and%20improve%20your%20mental%20health>