**CHAPTER ONE**

**INTRODUCTION**

**1.1 Background of Study**

People often need information while on the go. Sometimes the information required is essential to the task at hand, such as finding a hotel for the night. Other times, the need is associated with a question prompted by a conversation or a nearby object (e.g. a billboard). Based on the importance of the need and the amount of time available, people use a variety of strategies to obtain the desired information (Sohn et al. 2018).

Written diaries may not be appropriate for everyone and can exclude certain populations, including those with poor literacy and disabilities. During the past decades, for storing any information, there was large use of hard copy diaries. In a manual diary, there is difficult to search for any information it requires more effort and more time. Storing this information is a heavy task. It creates many security problems such as anyone can see the personal information of any other’s person (Bhosale et al., 2019).

Technological advances in recent years have meant that digital display media are becoming more “paper-like” and wireless. Lightweight digital displays are now more mobile and portable than ever. The quality and readability of the display screens are approaching the brightness, resolution, and contrast of plain paper. New techniques for stylus-based entry are making possible more paper-like interactions, such as allowing richer and more flexible mark-up and manipulation of digital documents (Manjunath et al. 2017).

Some of the most influential people in history kept detailed journals of their lives. The importance of diary writing cannot be over-emphasized. It can be said that diary writing harnesses the mental, creative, and emotional benefits of the diary author. It also helps the author keep some personal moments and other memorable important events. In the generation that we live in today, people don’t like to carry their traditional paper diaries along with them, this is simply because of their size. As such they are, they cannot record activities/events as they happen (Adeyanju et al., 2015).

Digital diaries have today had more functions than just storing data as against the conception of most individuals. The continuous availability of internet services and electronic/technological gadgets has made many people adopt it as a medium to chronicle their lives with the added dimension of an audience, below are examples/forms of diaries;

1. Personal diary
2. Sleep diary
3. Travel diary
4. War diary
5. Dream diary
6. Fiction diary

**1.2 Statement of the problem**

Diaries often contain both good and bad content. The ease at which Data/information is leaked out from a manual diary results in the blackmailing of the person(s) with the bad/negative content(s) in the diary. This is a big and mind-disturbing issue as an individual’s good reputation can be dented with negative content from his/her diary, denying him/her the good of tomorrow. Hence privacy and confidentiality have become a problem for handwritten or manual diaries. Thanks to technology, the negative issues of the manual diary have been addressed by digital diary.

**1.3 Aims and Objectives of the Study**

**This project aims to design and implement a mobile diary application.** The objectives are as follows:

1. To design a **mobile diary application**.
2. To implement a **mobile diary application**.
3. To evaluate the **mobile diary application**.

**1.4 Scope of the Study**

This project work is focused on developing a mobile diary that is user-friendly, with the user interface and interactions within the app, simple to use and easy to understand. It will allow the user make entries and tag them with emotions and to analyze any information they have entered, and also eliminating data leakage, by making available a confidential password and login system to access one’s notes and data entries. This project work would be available for everyone who wishes to make use of the system.

**1.5 Limitations of the study**

This study's scope has been constrained by several core issues, including:

**Time** - The researcher's everyday busy academic pursuits limited the time allotted for research for this study.

**Access to literature** – Access to some material was restricted, although the available material was optimized.

**1.6 Significance of the study**

**This project's development is done for more than simply a knowledge test, rather, it is carried out to enable the user to enjoy the freedom and flexibility of having everything they could produce in a physical diary in one comprehensive application, terminate the ease of privacy intrusion caused by handwritten diary and the frequent loss or theft of diary for blackmail, which has been a serious threat to individuals and organizations and this project work would tackle that.**

**1.7 Project Organization**

The project is divided into five chapters. The outlines are presented below:

**Chapter One: Introduction**

Chapter one introduces this project work, the study's background, the problem statement, the purpose and objectives, the scope of the study, the constraints of the study, the relevance of the study, the project organization, and the definition of terms.

**Chapter Two: Literature review**

This chapter focuses on the literature review, and the contributions of other scholars on the subject matter being discussed.

**Chapter Three: Methodology and Design**

This chapter is concerned with the presentation of the results of system analysis and design. It presents the research methodology used in the development of the system to facilitate an understanding and effective future implementation of the system.

**Chapter Four: System Implementation Evaluation**

This chapter describes the system implementation and documentation, analysis of modules, and system requirements for implementation.

**Chapter Five: Summary, Conclusion, and** **Recommendation**

The chapter provides a summary of major findings, conclusions, and recommendations based on the study conducted.

**1.8 Definition of Terms**

1. **Diary:** A diary refers to a written or audio-visual record with discrete entries arranged by date reporting what has happened over a day or other period.
2. **Digital Diary:** A digital diary is a place where you can record personal events and experiences online, rather than keeping a traditional diary or notebook to express your thoughts and feelings. (Merriam 2022).
3. **Application:** An application is a program or group of programs designed for end-users to perform a specific task.
4. **Personal diary:** a person's private record of events, thoughts, feelings, etc., that are written down every day.
5. **Sleep diary:** a tool used in the diagnosis and treatment of sleep disorder. It is also called a sleep log.
6. **Travel diary:** is the documentation of a journey or series of the journey. At times it is called a road journal
7. **War diary:** A war diary is a regularly updated official record kept by military units of their activities during wartime.
8. **Dream diary:** A dream diary (or dream journal) is a diary in which dream experiences are recorded. It is often used in the study of dreams and psychology.

**CHAPTER TWO**

**LITERATURE REVIEW**

**2.1 Introduction**

The field of the digital diary is a relatively new but rapidly expanding area of research. In this project, the basic diary function was meant to be enhanced by the addition of other elements, such as location tracking and reminder applications, to offer users a better product with greater functionality. This project started with an assessment of the existing dairy applications and known contributions by researchers to see what was currently available.

To fully understand what was already available, research was conducted on some of the most recent publications and apps. This involved trying out several of the apps for a while to completely comprehend their functionalities. In this chapter, we discuss the most known contributions of digital diaries.

**2.2 Review Of Existing Literature**

Fernández et al., 2017 carried out a study on “A digital diary to promote self-care among elders” called “InMyDay”, specially designed for elderly users. The goal of this diary was to promote self-care and self-reaction, by allowing users to register their activities and emotions. designed and implemented a digital diary that allows self-reporting of emotional state and daily activities. The application stores daily entries in a pre-defined format, and these entries can be viewed again by the users. The application was implemented for the Android system, with local data storage. The diary was adapted from a pen-and-paper version, which consisted of several pages with pre-defined questions about self-care and emotions. The diary consisted of high-contrast images with fixed questions asking users how went their day and how they are feeling at the moment. This project work was more of a research work and was tested amongst 10 elderly.

Moreso Greenslade, 2019 Developed “A Personal diary application that uses user location and emotion tagging to enrich the user experience”. The application was successful in achieving the projects main aim, the fusion of location and emotion to create a personal and helpful diary application. The user can create entries and view these at any time, these entries can be deleted and edited and basic data analysis provides users with interesting facts about their entries and their relationship with locations. Location is used to assess the user’s current location, the user can customise these location with their own names rather than rely on impersonal generations from services such as Google Maps. The application then uses this to suggest location to the user next time they write an entry. The application offers a range of emotions to tag entries with so users can see where they are often happy, sad or angry with the applications analysis.

Futhermore Bhosale et al., 2019 researched the “Development of Multipurpose e-Diary application” and came up with a platform that provided online storage with full security. This application was window based. It made it easy to get a reminder for future activities such as upcoming events, deadlines of work, reminders, and daily tasks. They developed the first module Daily Task in that users can enter daily activities such as daily schedules, deadlines of any work, important meetings, notes, and functional plans. Users can view tasks, add a task, update tasks, and delete a task. In the second module, which is Financial Summary users can enter financial entries and receive reports on entries. In the Remainder module, the user can remind his important deadlines, birthdays, important events, and tours notification get on mobile SMS. User data security is guaranteed as there is a password login requirement before data can be accessed or modified.

Hishobkar et al., n.d. 2019 also performed an analysis of “Digital Diary”, this study is an effort to recommend events to users within a social networking site. The system collects event data from users while setting up their digital diaries. It also permits users to rate events they have attended or planned. The main scope of the project is to provide a digital diary, where the user has a hassle-free experience in filling in all his details, An integrated map UI where the user can check out all the locations of the events he’s about to visit, plus with the recommender system, the user will get to know about similar events happening nearby Users can create events according to their suitability like birthdays or any organization event (meeting, etc.) as they want. The event will be created with information like event name, event place, date, and when the user wants a reminder at which time. The user will also get notified when the user is present at a recorded place.

Orhani et al., 2022 carried out a study on “Electronic school diary for statistical analysis of student progress”, this study investigated the effects of using a digital diary to facilitate the calculation of administrative statistics of teachers and classroom caregivers. The purpose of this study was to explore the use of an electronic diary (e-diary) in supporting teachers and classroom caregivers to complete administrative work related to completing the school diary. A quantitative design was used to obtain data from teachers and classroom caregivers through questionnaires. Overall, the study showed many potential advantages of using a digital diary at school to facilitate administrative tasks. Also, the results of the study are showing that the teacher and classroom caregivers were very satisfied with the integration of the e-diary to complete the statistical tasks calculated automatically from the electronic diary.

Namyenya et al., 2022 conducted a project study on “E-diary: a digital tool for strengthening accountability in agricultural extension”, the E-diary has two categories of users. The first is the field agents, and the second category of users is the supervisors. The user of the e-diary must have an e-diary account, which is protected by a username and password. Each supervisor creates an account for the immediate subordinate. The E-diary works within the existing agricultural extension procedures such as planning, conducting of activities, reporting, monitoring, evaluation, and feedback. The planning is annual and results in the formulation of annual work plans. The E-diary is embedded with special accountability features to enable the supervisors to verify the reported daily activities of the field agent. These include a beneficiary verification mechanism, the location of the activity, and activity photos. Regarding the beneficiary verification mechanism, for each activity, the field agent has to record the name and phone number of the beneficiary or reference beneficiary in the case of a group. Using the captured phone numbers, the supervisor can call the beneficiaries and verify the field agent’s visit. For the location, the field agent has to record the name of the village in which the activity was conducted. In addition, the system automatically captures the GPS coordinates, which verify the location entry. Concerning the activity photos, the field agent has to attach at least one activity photo that also provides evidence of the reported activity.

Schmitz et al., 2022 also conducted Research on a “Digital Sleep Diary Standard”. This paper rests on an action design research project to design and develop a digital sleep diary as an app. The app was co-designed, involving end-users, researchers, and healthcare professionals in the design process. They used two questionnaires, a survey, and application analytics to collect quantitative data and complemented that with the qualitative data that was gained from the semi-structured interviews. This research was aimed at increasing compliance and counteracting memory bias, which was the main issues found while using the pen-and-paper version. They found that on a piece of paper, it cannot be controlled when and how participants enter information. So they concluded that these issues can be avoided with input control in a digital sleep diary. Choosing native inputs for time, numbers, Booleans (true/false), or text prevents the participant from entering incorrect data and ensures data homogeneity.

**2.3 Summary of Related Literature Reviews**

|  |  |  |
| --- | --- | --- |
| **Author & Year** | **Title & Description** | **Merit and Demerits** |
| Fernández et al., 2017 | A digital diary to promote self-care among elders  called “InMyDay”.  The developers implemented a digital diary, specially designed for elderly users. that allows self-reporting of emotional state and daily activities. The application stores daily entries in a pre-defined format, and these entries can be  viewed again by the users. The application was implemented for the Android  system, with local data storage. | The system had predefined emotions and questions for fast data entry into the diary.  The Pre-defined entry doesn’t allow much self expression as one is restricted to only entries in the diary. |
| Bhosale et al., 2019 | Development of Multipurpose e-Diary application.  This project was window based. It made it easy to get a reminder for future activities and daily tasks, also make financial entries in the financial summary module and receive reports on entries, reminders. | It provides vast user experience as it is an all in one application and user data is protected as there is a login requirement for authentication.  The system was window based as this reduces the portability of the system and restricts it to one section of users. |
| Greenslade, 2019 | A Personal diary application that uses user location and emotion tagging to enrich the user experience  This project develops an android application that offers the user Emotional tags to identify their feelings within an entry as well as tag them with locations. It also uses emotion and location to produce statistics and information that will help the user learn about their routines and lives. | The App is user friendly and users can tag their diary entries with emoji's and location to always have a record of places visited  Users are not able to search for or categories entries, especially when the user has created many entries. |
| Hishobkar et al.,2019 | Digital Diary  This research focuses on a mobile diary that allows users create events according to their suitability like birthdays or any organization event as they want. The events are created by the information provided like event name, event place, date and when the user want reminder at which time . The user will also get notified when present at the nearby location of the event. | The User interface of the application is simple, fast and smooth functioning.  The system is limited to only Android-based users. |

**2.4 Research Gap & Conlusion**

After carefully reviewing all these past work, the research project by Greenslade, 2019 was identified as the one most similar to the ideas and requirements that this project intended to achieve. Although this project would be designed using flutter and looks to create an application that works on the android platform entirely and users would be able to tag their diary entries with emotions and as well be able to search through their entries to see similar events.

**CHAPTER THREE**

**METHODOLOGY AND DESIGN**

**3.1 Introduction**

This chapter expands on the examination of the suggested system and its architecture. That is to say, the components, features, and design elements (outputs, inputs, database) of the proposed system, Digital Diary. The same goes for the tools and technology that are employed.

**3.2 Method of Data Collection**

The research methodology used in this research work includes documentation and primary observation of the existing system.

**3.2.1 Observation Method**

This approach was used to collect information and data for this study by observing how the manual system functioned. Careful observation using the observational technique was used to identify the glaring flaws in the current system.

**3.2.2 Documentation**

A supplementary form of data acquisition is documentation. Journals, manuals, previous projects, play store reviews, and other sources are used in this approach. This type of data collecting was employed because it provides a foundation for comparison with earlier studies.

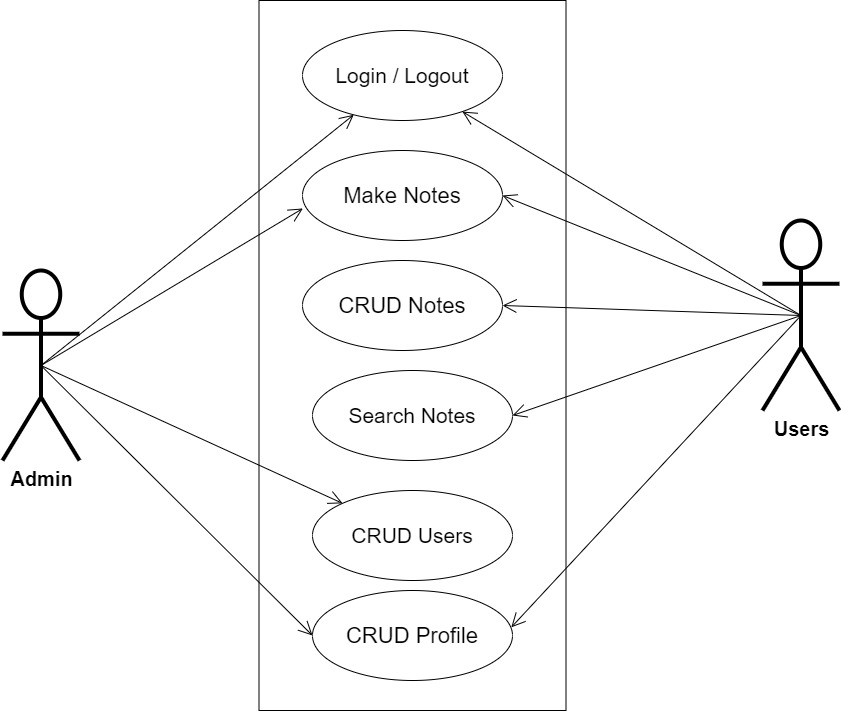
This project was carried out primarily using the agile development technique. The project will be able to adapt and grow as additional knowledge and techniques are acquired thanks to the agile approach. Breaking down each portion of the project into smaller phases will provide me with more time to focus on the parts that call for more reading and study. As this is an individual project, several techniques from the agile methodology won't work for a single person, but the fundamental development process will. These will consist of the more manageable tasks that are routinely monitored to make sure they are completed as effectively as possible. To make sure the system is on track to meet the needs that were previously outlined, regular evaluations of the system as a whole will be conducted, either by a developer or with users, to identify any changes that may be necessary to the system itself. Any additional features won't be added until the core functionality is performing to expectations.

**3.3 System Modelling**

The Unified Modeling Language was used in this project design which includes a set of graphic notation techniques to create visual models for the Object-oriented application. The UML applied in this new design includes Use Case Diagram, Class Diagram, and Activity Diagram.

**3.3.1 Use Case Diagrams**:

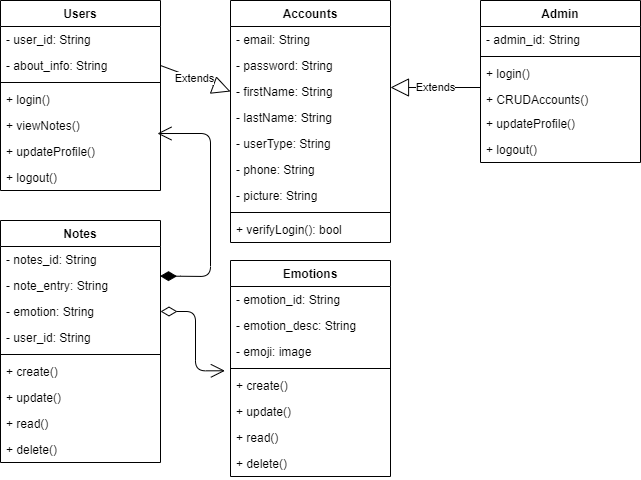
The use cases and primary tasks that a user will be able to perform within the program are depicted in the accompanying diagrams.



**Fig 3.1 System Use Case Diagram**

**3.3.2 Class Diagram**

The fundamental organization of each class in the system is depicted in the diagram below. Each class's essential variables and methods are displayed, and certain variables and methods have not been added to the diagram for the sake of clarity. There is also the opportunity to access and subsequently instantiate Main Activity, Home Page, and Analysis Home within each of the main classes. The navigation bar makes this possible, but these instantiations haven't been included in the diagram because it would complicate it.



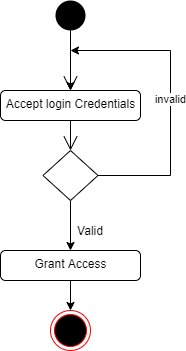
**Fig 3.2 System Class Diagram**

**3.3.3 Activity Diagrams**

Activity diagrams have been developed to further demonstrate how each class can access all others and to display the possible activities within each of the system's divisions.

**Login**

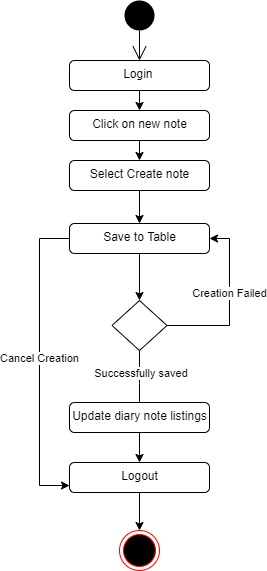
The process for gaining access to the system is depicted in the diagram below; to gain access, the email address and password must be accurate.



**Fig 3.3 Login Activity Diagram**

**Taking Diary Notes**

The process for making a diary note is depicted below



**Fig 3.4 Taking Diary Notes Activity Diagram**

**3.4 Database Design**

Input specification is the logical explanation of how data is stored in the computer's memory. SQL standards are vital for guaranteeing that structured data is uniform and independent of applications due to the flexibility experienced when using the system, as well as the convenience of retrieving and reading the data and assuring applicability throughout the internet. The following are some of the input specifications used in this project effort.

1. Accounts Table: contains basic information about all system users.
2. Note Table: contains every user note information.

**Table 3.1 Account Table input specification table**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| Email | Varchar | 150 | Email for login (case sensitive) |
| Password | Varchar | 150 | Access Code (case sensitive) |
| Firstname | Varchar | 150 | User first name |
| Lastname | Varchar | 150 | User last name |
| Phone | Varchar | 11 | User phone number |
| Picture | Varchar | 100 | User profile picture |
| acct\_id | Varchar | 64 | A unique string for identifying users |

**Primary key:** acct\_id

**Table 3.2 Note Table input specification table**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| Note\_id | Varchar | 150 | A unique string for identifying tricycles |
| Note\_entry | Varchar | - | Diary note |
| User\_id | Varchar | 150 | A unique string for identifying user |

**Primary key:** note\_id

**3.5 Output Design**

This declares and displays the outcome of the given input. The automated system's output is dependent on its input. The output specification is listed below.

**Table 3.3 Account Output Design Table**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Email** | **Password** | **Firstname** | **Lastname** | **Phone** | **Picture** | **Acct\_id** |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |

**Table 3.4 Note Output Design Table**

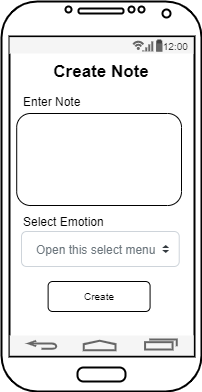
|  |  |  |
| --- | --- | --- |
| **Tricycle\_id** | **User\_id** | **Note\_entry** |
| XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX |

**3.6 Input & User Interface Design**

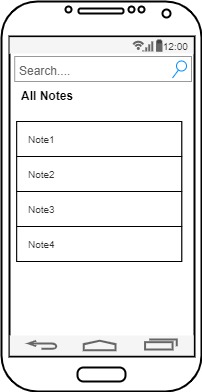
This shows a visual representation of the system interface; it will be made to be intuitive to use, quick to respond to, and visually appealing. Additionally, it will be properly protected, so signing in will be necessary to view some levels of the contents. A mid-fidelity wireframing application named Draw.io is used to assist with the designs.



**Fig 3.5 User Login Screen**



**Fig 3.6 Create Note Screen**



**Fig 3.6.3 Search Note Screen**

**3.7 System Requirement**

The goal of this project is to offer users comprehensive solutions for keeping track of their everyday activities, thoughts, and chores using only their mobile devices. Users will be able to sign in or register with ease. The following list of initial functional requirements.

* + 1. **Hardware Requirements:**

1. CPU: i5 core processor
2. Clock speed: 2.4 GHz
3. Main memory: 8GB RAM
4. Secondary memory: 500GB Hard Disk
   * 1. **Software Requirement:**

To build this project, there is the utilization of the following Software resources:

1. At least windows 10 OS (Operating System).
2. Flutter Installation.
3. Vs. Code / Android studio installation.
4. Emulator installation.

**3.8 Choice of Programming Language**

The proposed design will be implemented using flutter for its user interface (frontend) while Python will be used for the backend programming, Sqlite3 will be used for its database due to its portability, and Django REST Framework will be employed for its REST-full APIs, the combination of the above modern technology forms the technology for this research work

**CHAPTER FOUR**

**System Implementation Evaluation**

**4.1 Introduction**

This section covers in detail how the new system is implemented to ensure its effectiveness. It depicts functional (new) system examples as well as how the system is to be implemented.

* 1. **System Testing and Evaluation**

There are various reasons why the developed system should be tested. For example, only via testing will we be able to identify any flaws in the new system and provide solutions. This project employed both unit and integration testing to validate the design's efficacy and efficiency, as well as to guarantee the new system meets its functional requirements and is error-free.

**Unit Testing**

specific units or single components of the system are examined individually in this part to confirm that specific phases function properly and without problems.

**Integration Testing**

The software was tested utilizing integration testing, in which all parts were assembled and functioned as one. The connectivity between the various components was checked to ensure that they are properly integrated and that the units can work together as one.

**4.3 System Installation**

In order to use the proposed application on any computer system, the following steps need to be taken:

1. Make sure, android studio, JDK, and Android emulator are installed on the system.
2. Copy your project folder to any location of your choice.
3. Open the project folder in Visual Studio Code
4. In the terminal run “flutter pub get” to get all the dependencies in the pubspec.yaml file
5. Select the Android emulator as the device to be used.
6. Locate the main.dart file and run the file in debug mode.

**4.4 Security Measures**

Since the scope of the application is public, literally all the information is made available to any user, but some functionalities are restricted to the admin, functionalities that have to do with managing the user accounts, updating the emotion icons are restricted from the general public. The restriction is carried out by using passwords when the admin webpage is accessed.

**4.6 Sample Outputs**

These describe and give the pictorial representation of the program or software; it shows and gives clear understanding of the design, and displays all the interfaces.

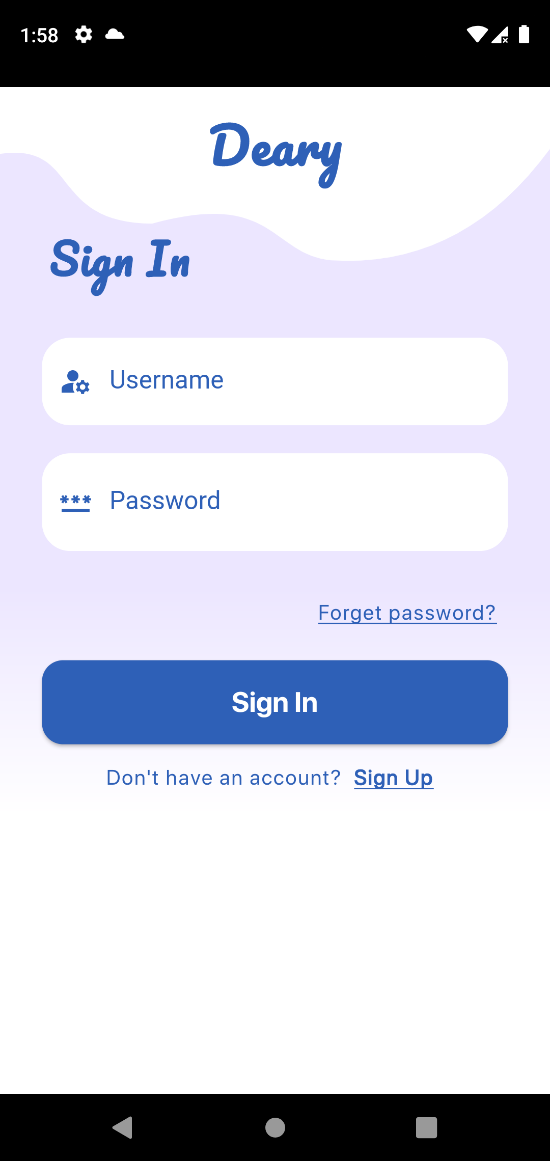


Fig 4.6.1: Splash Screen Fig 4.6.2: Login Screen

**Fig 4.6.1 Splash Screen**: This is the first screen displayed to every user that wishes to make use of the application.

**Fig 4.6.2 Login Screen**: The screen grants users access to the application only if the correct credentials are provided.

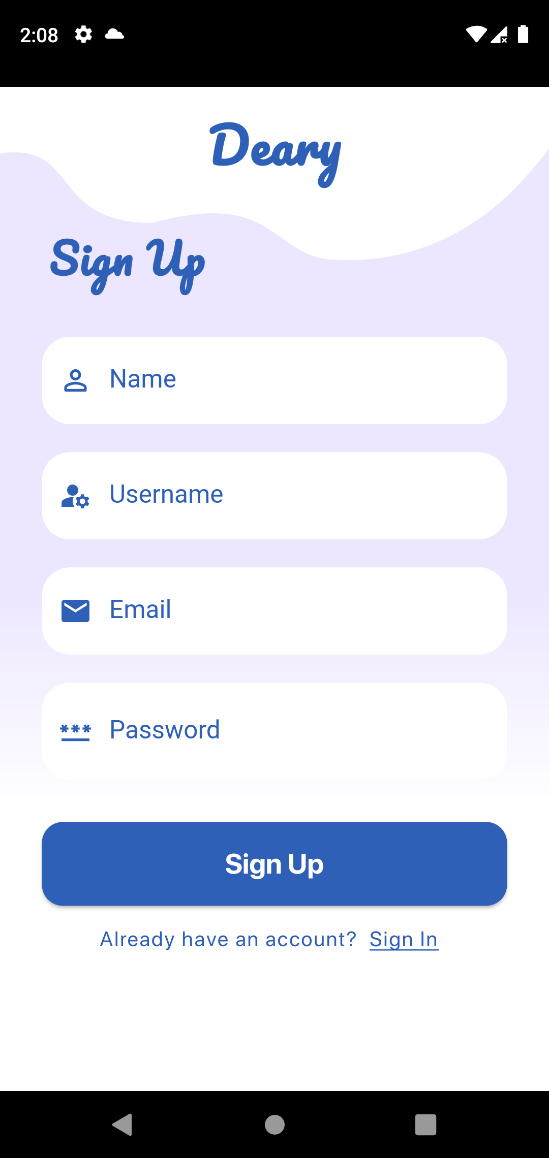
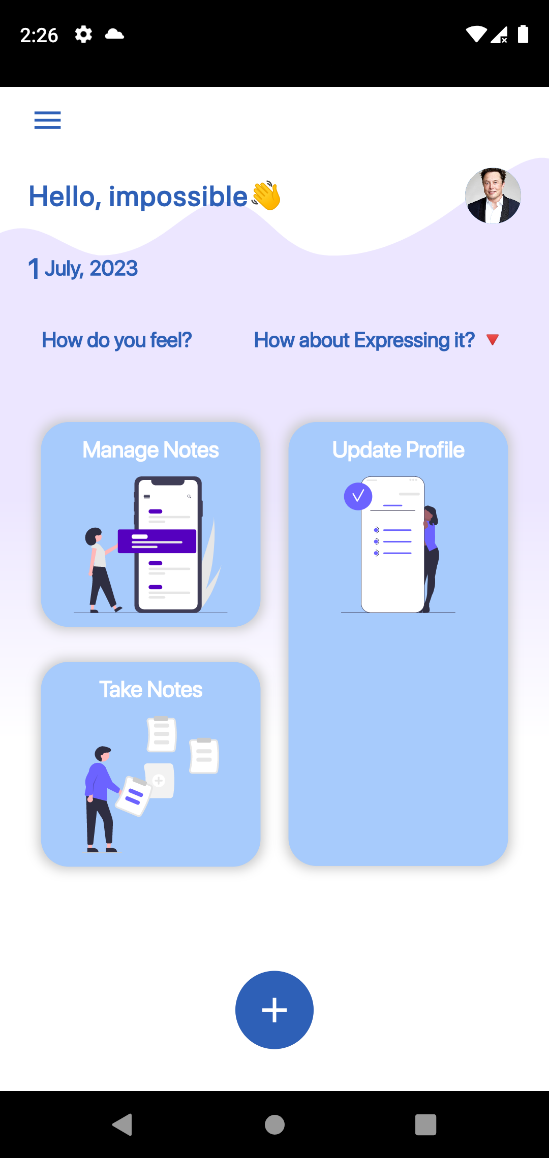


Fig 4.6.3: Signup Screen Fig 4.6.4: Home Screen

**Fig 4.6.3 Signup Screen**: The application ensures that only registered users can have access to the system, there the screen enables the creation of account for new users.

**Fig 4.6.4 Home Screen**: The screen depicts the dashboard of an authenticated user, it also highlights some of the functionalities that the application provides.

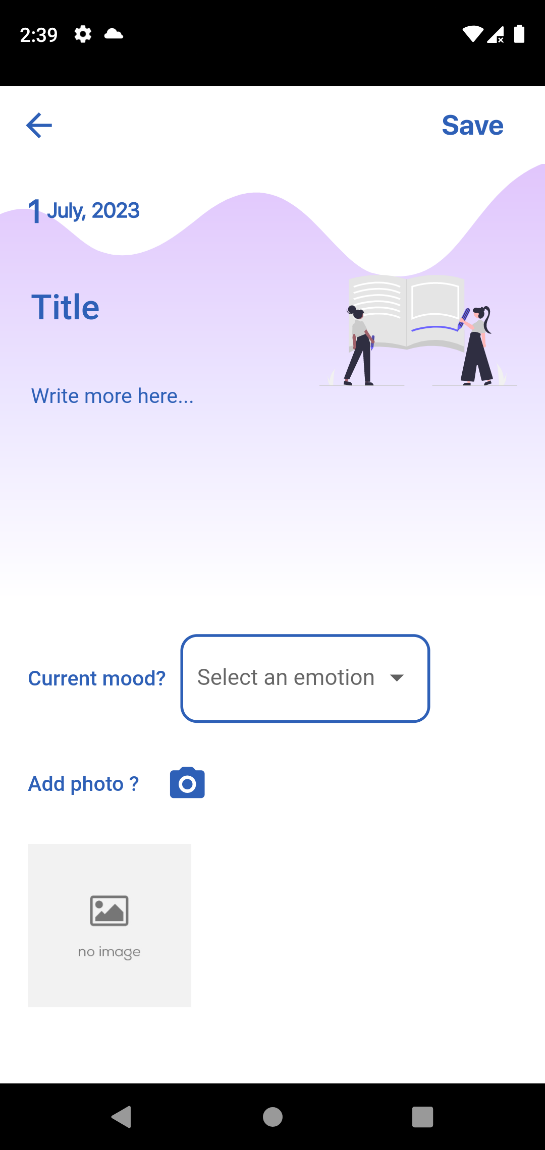
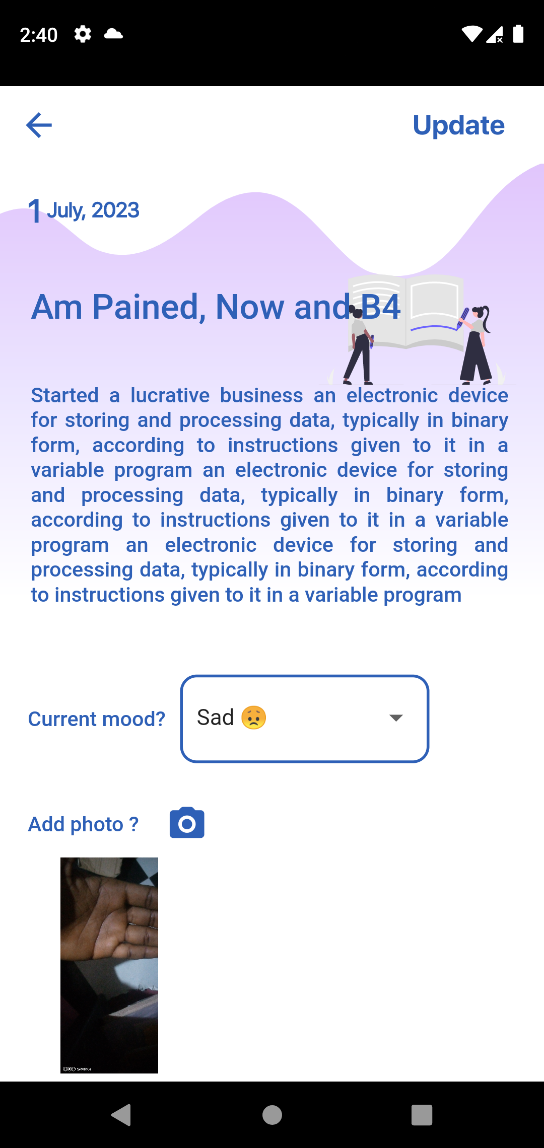


Fig 4.6.5: Take a Note Screen Fig4.6.6: Update a Note Screen

**Fig 4.6.5 Take a Note Screen**: The screen displays where an authenticated user can write down a note of things happening, the notice can contain a title, the detail of the note, the mood of the current mode of the user and an optional image.

**Fig 4.6.6 Update a Note Screen**: The screen can be use to update the content of an already existing note.

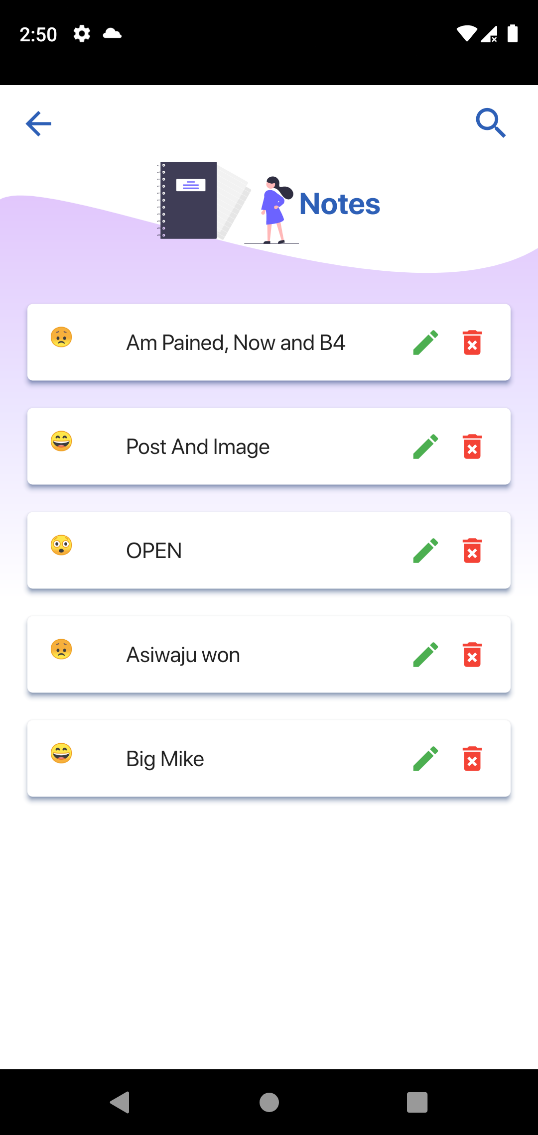
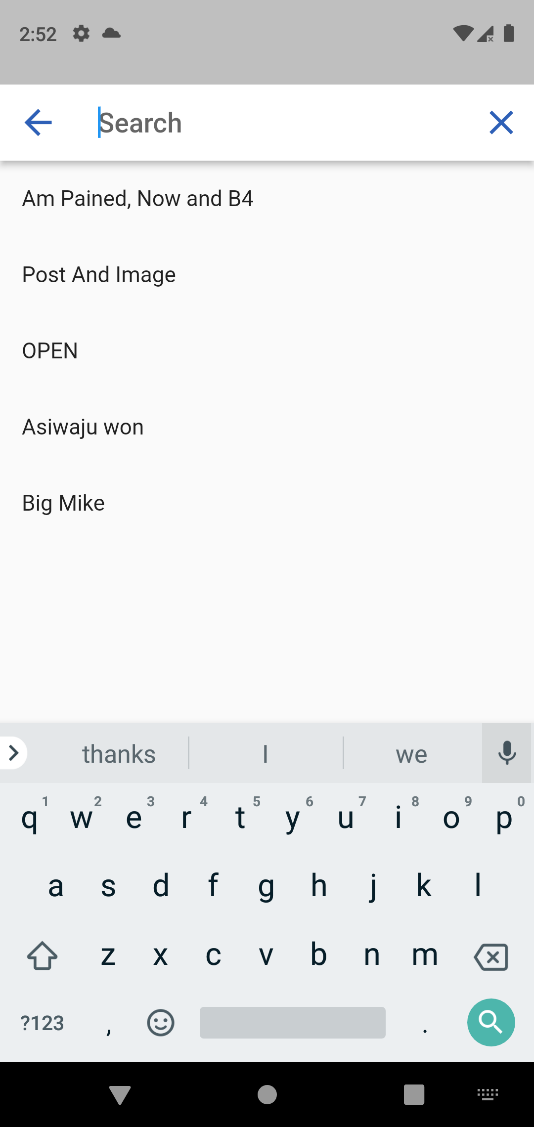


Fig 4.6.7: All Notes Screen Fig 4.6.8: Search Screen

**Fig 4.6.7 All Note Screen**: The screen contains the list of all notes recorded by the current authenticated user, with the options to edit and delete a note.

**Fig 4.6.8 Search Screen**: The application implement a search functionality to enable users quick access a note from numerous notes.

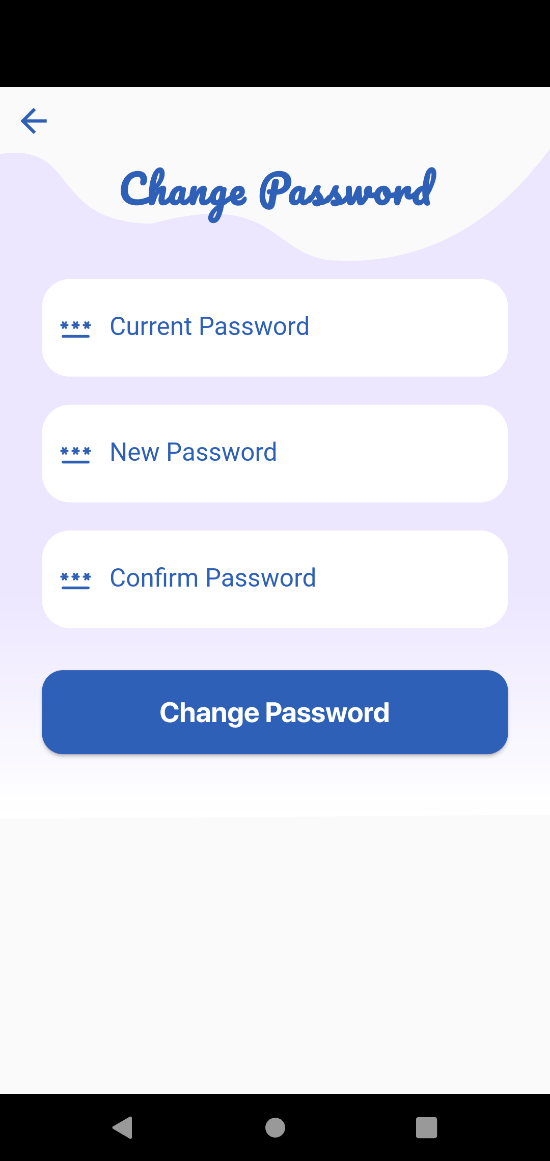
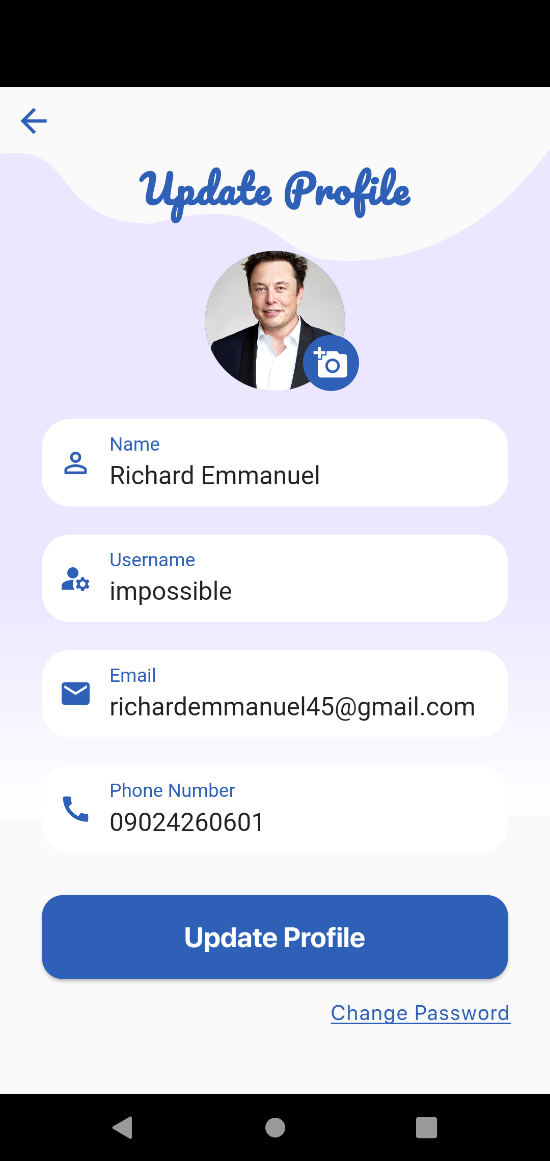


Fig 4.6.9: Update Profile Fig 4.6.9.1: Change Password

**Fig 4.6.9 Update Profile**: This screen enables users to update account related informations such as their name, username, email, phone number and picture

**Fig 4.6.9.1 Change Password**: This screen enables users to update their account password

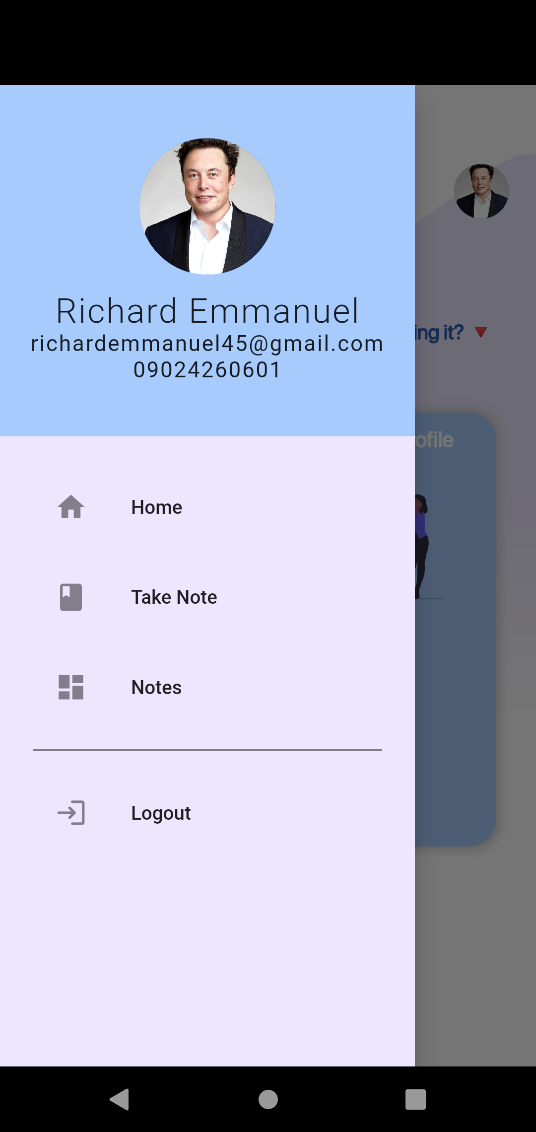
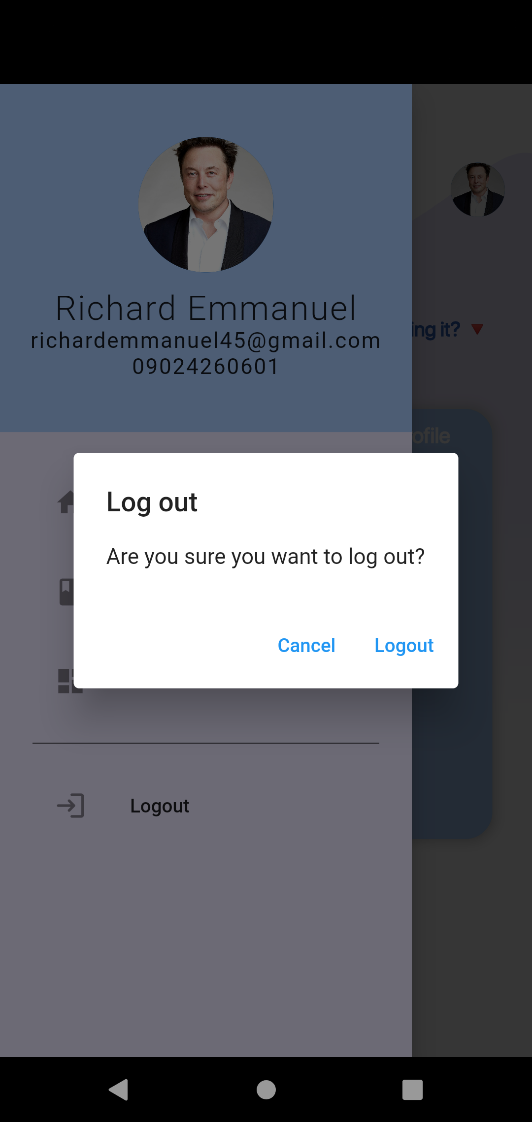


Fig 4.6.9: Navigation Drawer Fig 4.6.7: Logout Option

**Fig 4.6.9 Navigation Drawer**: This screen depicts the functionalities available on the application through the use of a sidebar.

**Fig 4.6.9.1 Logout Option**: This option logs out the user from the application, user has to reauthenticate again in order to gain access to the system

**CHAPTER FIVE**

**Summary Conclusion and Recommendation**

**5.1 Summary**

This research project aims to design and implement a mobile diary application to address the limitations and privacy concerns associated with traditional manual diaries. The project focuses on developing a user-friendly mobile diary app with features such as entry tagging, emotion analysis, and password-protected access to ensure data confidentiality. The study's scope is centered on creating a comprehensive and secure digital diary solution that can be utilized by individuals seeking a convenient and private way to document their daily activities and experiences. Although limited by time and access to literature, the significance of this project lies in providing users with the freedom and flexibility of a digital diary while eliminating privacy intrusion and the risk of loss or theft associated with handwritten diaries.

**5.2 Conclusion**

In conclusion, the research project met its objectives and made substantial progress in the creation of a mobile diary application. The application leverages technology to give users with a user-friendly and secure platform for recording their everyday activities and experiences. Privacy and secrecy are safeguarded by features such as entry tagging, emotion analysis, and password-controlled access. The value of this initiative stems from its capacity to solve the constraints of conventional handwritten diaries, providing users with a modern and simple method for chronicling their life. The mobile diary application is a useful tool for users who want to save personal moments and recollections in an effective and safe manner.

**5.2 Recommendation**

Based on the findings and outcomes of the research project, the following recommendations are proposed:

1. Continual Improvement: The development of the mobile diary application should be an iterative process, with regular updates and enhancements based on user feedback and emerging technological advancements. This will ensure that the application remains relevant, user-friendly, and equipped with the latest features and functionalities.
2. Enhanced Security Measures: While the mobile diary application already includes a password-protected access system, additional security measures should be considered. Implementing measures such as two-factor authentication and data encryption will further enhance the privacy and confidentiality of user data.
3. Collaboration Features: Consider incorporating collaboration features into the mobile diary application, enabling users to share entries or collaborate on diary projects with friends, family, or colleagues. This would facilitate group activities, such as shared travel diaries or collaborative storytelling, enhancing the social aspect of diary writing.
4. Data Backup and Recovery: To prevent data loss, it is crucial to implement a robust data backup and recovery system within the mobile diary application. Regular automatic backups should be performed, and users should have the ability to restore their data in case of device loss or technical issues.

By implementing these recommendations, the mobile diary application can continue to evolve as a reliable, user-centric, and feature-rich platform, meeting the needs and preferences of a diverse user base.

**References**

Adeyanju, I. A., Alo, O. O., & Aderibigbe, A. O. (2015). Development of an Electronic Diary Application for Windows Smartphones. *Computing, Information Systems Development Informatics & Allied Research (CISDIAR) Journal*, *6*(2), 45–52. http://www.cisdijournal.net/uploads/V6N2P6\_-\_\_CISDIAR\_JOURNAL.pdf

Bhosale, A. T., Moholkar, M. R. S., & Takale, M. R. K. (2019). *Development of Multipurpose e-Diary*. *VI*(301), 301–303.

Fernández, M., Rodríguez, I., Rossel, P. O., Fuentes, C., & Herskovic, V. (2017). InMyDay: A digital diary to promote self-care among elders. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10586 LNCS*, 486–497. https://doi.org/10.1007/978-3-319-67585-5\_49

Greenslade, A. N. (2018). *A Personal diary application that uses user location and emotion tagging to enrich the user experience*.

Hishobkar, A. A., Technology, I., Mumbai, N., Gaonkar, R., Technology, I., Mumbai, N., Chintamani, J., Technology, I., & Mumbai, N. (n.d.). *Digital diary*.

Namyenya, A., Daum, T., Rwamigisa, P. B., Birner, R., Namyenya, A., Daum, T., Rwamigisa, P. B., & Birner, R. (2022). *Information Technology for Development E-diary : a digital tool for strengthening accountability in agricultural extension*. https://doi.org/10.1080/02681102.2021.1875186

Orhani, S., Saramati, E., & Drini, L. (2022). *Electronic school diary for statistical analysis of student progress*. *1*(3), 58–65.

Schmitz, L., Sveinbjarnarson, B. F., Gunnarsson, G. N., Davíðsson, Ó. A., Davíðsson, þ. B., Arnardottir, E. S., Óskarsdóttir, M., & Islind, A. S. (2022). Towards a Digital Sleep Diary Standard. *Sleep Medicine*, *100*, S305. https://doi.org/10.1016/j.sleep.2022.05.824

Shields, A. (n.d.). *Unleashed : A personal diary app for tracking walks with your dog Final Report*.

Sohn T, Kevin A. L, William G. G, and James D. H (2018). A Diary study of mobile information needs. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08). Association for Computing Machinery, New York, NY, USA, 433–442. <https://doi.org/10.1145/1357054.1357125>.

**APPENDIX**

**Views.py**

from rest\_framework import generics

from rest\_framework import status

from rest\_framework.response import Response

from rest\_framework import permissions

# MyApp Import

from Notes.serializers import (

    NoteSerializers,

    MoodSerializers,

    AllNotesSerializers,

    DetailNotesSerializers,

)

from Notes.models import (

    Mood,

    Notes,

)

import base64

# Create your views here.

class CreateNoteView(generics.CreateAPIView):

    """This view create note for a user"""

    serializer\_class = NoteSerializers

    permission\_classes = (permissions.IsAuthenticated,)

    def create(self, request, \*args, \*\*kwargs):

        """Creates a note"""

        super().create(request, \*args, \*\*kwargs)

        return Response(status=status.HTTP\_201\_CREATED)

class GetMoodsView(generics.ListAPIView):

    """This view create note for a user"""

    serializer\_class = MoodSerializers

    permission\_classes = (permissions.IsAuthenticated,)

    model = Mood

    queryset = Mood.objects.all()

class GetNotesView(generics.ListAPIView):

    """This view gets all notes from a user"""

    serializer\_class = AllNotesSerializers

    permission\_classes = (permissions.IsAuthenticated,)

    model = Notes

    queryset = Notes.objects.all()

    def get\_queryset(self):

        return Notes.objects.filter(user\_id=self.request.user).order\_by('-date\_created')

class DetailNoteView(generics.RetrieveAPIView):

    """This view returns a user"""

    serializer\_class = DetailNotesSerializers

    permission\_classes = (permissions.IsAuthenticated,)

    def get\_object(self):

        print(Notes.objects.get(pk=self.kwargs.get('note\_id')))

        return Notes.objects.get(pk=self.kwargs.get('note\_id'))

class EditNoteView(generics.UpdateAPIView):

    """This view returns a user"""

    serializer\_class = NoteSerializers

    queryset = Notes.objects.all()

    permission\_classes = (permissions.IsAuthenticated,)

class DeleteNoteView(generics.DestroyAPIView):

    """This view deletes a note"""

    serializer\_class = NoteSerializers

    queryset = Notes.objects.all()

    permission\_classes = (permissions.IsAuthenticated,)

    def delete(self, request, \*args, \*\*kwargs):

        super().delete(request, \*args, \*\*kwargs)

        return Response(status=status.HTTP\_200\_OK)

**Homepage**

import 'dart:ui' as ui;

import 'package:diary/controllers/logout\_controller.dart';

import 'package:diary/controllers/notes\_controller.dart';

import 'package:diary/routes/routes.dart';

import 'package:flutter/material.dart';

import 'package:diary/services/constants.dart';

import 'package:diary/controllers/profile\_controller.dart';

import 'package:diary/services/home\_decor.dart';

import 'package:get/get.dart';

import 'package:fluttertoast/fluttertoast.dart';

class Home extends StatelessWidget {

  Home({super.key});

  ProfileController profileController = Get.put(ProfileController());

  DateTime timeBackPressed = DateTime.now();

  final scaffoldKey = GlobalKey<ScaffoldState>();

  @override

  Widget build(BuildContext context) {

    final Size size = MediaQuery.of(context).size;

    return SafeArea(

      child: Scaffold(

        floatingActionButton: Center(

          heightFactor: 1.5,

          child: FloatingActionButton(

            onPressed: () => Get.toNamed(Routes.takeNote),

            backgroundColor: primaryColor,

            elevation: 0.1,

            child: const Icon(

              Icons.add,

              size: 30,

              color: tertiaryColor,

            ),

          ),

        ),

        floatingActionButtonLocation: FloatingActionButtonLocation.centerFloat,

        drawer: const NavigationDrawer(),

        backgroundColor: tertiaryColor,

        body: Stack(

          children: [

            CustomPaint(

              size: Size(size.width, (340 \* 1.7777777777777777).toDouble()),

              painter: MyShape(),

            ),

            Column(

              crossAxisAlignment: CrossAxisAlignment.start,

              children: [

                Padding(

                  padding: const EdgeInsets.only(left: 10),

                  child: Builder(builder: (context) {

                    return IconButton(

                      onPressed: () => Scaffold.of(context).openDrawer(),

                      icon: const Icon(

                        Icons.menu,

                        color: primaryColor,

                        size: 25,

                      ),

                    );

                  }),

                ),

                Padding(

                  padding: const EdgeInsets.only(

                    top: 10,

                    left: 20,

                    right: 20,

                  ),

                  child: Obx(

                    () => profileController.isLoading.value

                        ? const CircularProgressIndicator()

                        : Row(

                            mainAxisAlignment: MainAxisAlignment.spaceBetween,

                            children: [

                              Text(

                                "Hello, ${profileController.userProfile?.username}👋",

                                style: const TextStyle(

                                  color: primaryColor,

                                  fontSize: 20.0,

                                  letterSpacing: 1.0,

                                  fontFamily: 'SFPReg',

                                  fontWeight: FontWeight.w600,

                                ),

                              ),

                              CircleAvatar(

                                maxRadius: 20,

                                minRadius: 20,

                                child: ClipOval(

                                  child: InkWell(

                                    onTap: () =>

                                        Get.toNamed(Routes.editProfile),

                                    child: Image.memory(

                                      profileController.userProfile!.image,

                                      height: 40,

                                      width: 40,

                                      fit: BoxFit.fitWidth,

                                    ),

                                  ),

                                ),

                              ),

                            ],

                          ),

                  ),

                ),

                Padding(

                  padding: const EdgeInsets.only(

                    left: 20,

                    top: 20,

                  ),

                  child: Row(

                    mainAxisAlignment: MainAxisAlignment.start,

                    children: [

                      Text(

                        day,

                        style: const TextStyle(

                          color: primaryColor,

                          fontSize: 20.0,

                          fontFamily: 'SFPReg',

                          fontWeight: FontWeight.w600,

                        ),

                      ),

                      Text(

                        " $month,",

                        style: const TextStyle(

                          color: primaryColor,

                          fontSize: 15.0,

                          fontFamily: 'SFPReg',

                          fontWeight: FontWeight.w600,

                        ),

                      ),

                      Text(

                        " $year",

                        style: const TextStyle(

                          color: primaryColor,

                          fontSize: 15.0,

                          fontFamily: 'SFPReg',

                          fontWeight: FontWeight.w600,

                        ),

                      ),

                    ],

                  ),

                ),

                Padding(

                  padding: const EdgeInsets.only(

                    left: 30,

                    top: 30,

                    right: 30,

                  ),

                  child: Row(

                    mainAxisAlignment: MainAxisAlignment.spaceBetween,

                    children: const [

                      Text(

                        "How do you feel?",

                        style: TextStyle(

                          color: primaryColor,

                          fontSize: 15.0,

                          fontFamily: 'SFPReg',

                          fontWeight: FontWeight.w600,

                        ),

                      ),

                      Text(

                        "How about Expressing it? 🔻",

                        style: TextStyle(

                          color: primaryColor,

                          fontSize: 15.0,

                          fontFamily: 'SFPReg',

                          fontWeight: FontWeight.w600,

                        ),

                      ),

                    ],

                  ),

                ),

                const SizedBox(height: 40),

                Expanded(

                  flex: 1,

                  child: LayoutBuilder(

                    builder: ((context, constraints) => Row(

                          mainAxisAlignment: MainAxisAlignment.center,

                          children: [

                            Padding(

                              padding: const EdgeInsets.all(10.0),

                              child: Column(

                                children: [

                                  HomeDecor(

                                    height: .3,

                                    constraints: constraints,

                                    title: 'Manage Notes',

                                    svg: 'assets/tasks.svg',

                                  ),

                                  const SizedBox(height: 25),

                                  HomeDecor(

                                    height: .3,

                                    constraints: constraints,

                                    title: 'Take Notes',

                                    svg: 'assets/add\_notes.svg',

                                  ),

                                ],

                              ),

                            ),

                            Padding(

                              padding: const EdgeInsets.all(10.0),

                              child: Column(

                                children: [

                                  HomeDecor(

                                    height: .65,

                                    constraints: constraints,

                                    title: 'Update Profile',

                                    svg: 'assets/reminder.svg',

                                  ),

                                ],

                              ),

                            ),

                          ],

                        )),

                  ),

                ),

              ],

            ),

          ],

        ),

      ),

    );

  }

}

class NavigationDrawer extends StatelessWidget {

  const NavigationDrawer({super.key});

  @override

  Widget build(BuildContext context) {

    ProfileController profileController = Get.put(ProfileController());

    NotesController notesController = Get.put(NotesController());

    return Drawer(

      backgroundColor: secondaryColor,

      child: SingleChildScrollView(

        child: Column(

          crossAxisAlignment: CrossAxisAlignment.stretch,

          children: [

            buildHeader(context, profileController),

            buildMenuItems(context, notesController),

          ],

        ),

      ),

    );

  }

}

Widget buildHeader(BuildContext context, dynamic profileController) => Material(

      color: blueColor,

      child: InkWell(

        onTap: () {

          Get.toNamed(Routes.editProfile);

        },

        child: Container(

          padding: EdgeInsets.only(

              top: MediaQuery.of(context).size.height \* .05,

              bottom: MediaQuery.of(context).size.height \* .05),

          child: Column(

            children: [

              CircleAvatar(

                maxRadius: 50,

                minRadius: 50,

                child: ClipOval(

                  child: Image.memory(

                    profileController.userProfile!.image,

                    height: 160,

                    width: 160,

                    fit: BoxFit.fitWidth,

                  ),

                ),

              ),

              const SizedBox(height: 12),

              Text(

                "${profileController.userProfile.name}",

                style: const TextStyle(

                  fontSize: 25,

                  fontWeight: FontWeight.w300,

                  letterSpacing: 1,

                ),

              ),

              Text(

                "${profileController.userProfile.email}",

                style: const TextStyle(

                  fontSize: 16,

                  letterSpacing: 1,

                ),

              ),

              (profileController.userProfile.phone != null)

                  ? Text(

                      "${profileController.userProfile.phone}",

                      style: const TextStyle(

                        fontSize: 16,

                        letterSpacing: 1,

                      ),

                    )

                  : const Text(""),

            ],

          ),

        ),

      ),

    );

Widget buildMenuItems(BuildContext context, dynamic notesController) =>

    Container(

      padding: const EdgeInsets.all(24),

      child: Wrap(

        runSpacing: 10,

        children: [

          ListTile(

            leading: const Icon(Icons.home),

            title: const Text('Home'),

            onTap: () {

              Navigator.pop(context);

              Get.toNamed(Routes.home);

            },

          ),

          ListTile(

            leading: const Icon(Icons.book\_rounded),

            title: const Text('Take Note'),

            onTap: () {

              Navigator.pop(context);

              Get.toNamed(Routes.takeNote);

            },

          ),