Software Requirements Specification

for

Mindful Life Companion

Version 1.0 approved

Prepared by: Andy Liang, Tommy Works, Anthony Chieng, Bryam Ochoa, Luis Sanchez, Emily Perez, Giovanni Chumpitazi

CS 3337-03

05/09/2024

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Tommy	5/2	Extra information	6.1
Giovanni	5/2	Removed features that were not worked on	

<Add rows as necessary when the document is revised. This document should be consistently updated and maintained throughout your project. If ANY requirements are changed, added, removed, etc., immediately revise your document.>

1. Introduction

1.1 Purpose

This document outlines the System Requirements Document (SRD) for the Mindful Life Companion App v1.0. This initial version focuses on the core functionalities for managing authentication, user accounts, and mood tracking, Journaling, and a variety of tools to support mental health. Future versions may address additional features that will be added. This document serves as a collective guide for the development team, detailing the required functionalities, performance expectations, and user interactions with the system.

1.2 Intended Audience and Reading Suggestions

Developers: Responsible for translating requirements into software functionality. Focus on sections 2 (Overall Description), 4 (Requirements Specification), and 5 (Non-Functional Requirements) for detailed specifications.

Project Managers: Will Oversee the project's progress and allocate resources accordingly. A general overview in sections 1 (Introduction) and 2 (Overall Description) is sufficient. Refer to the sections recommended for developers and developers if specifics are required.

Marketing Staff: Can utilize sections 1.3 (Product Scope) and 2 (Overall Description) to better understand what the system offers and the benefits for promoting it effectively.

Users: Sections 1.3 (Product Scope) and 2 (Overall Description) will offer a high-level overview of the software's functionalities and the relevance to their needs.

Testers: Should prioritize sections 3 (External Interface Requirements), 4 (Functional Requirements) and 5 (Other Nonfunctional Requirements)

Documentation Writers: Utilize this SRD as a source for technical details when crafting user guides and other supporting documents

1.3 Product Scope

This SRD focuses on the development of Mindful Life Companioin v1.0. This software aims to offer a variety tools to support mental health such as:

- Mood Tracking (adding, editing, viewing moods, and generating reports trends over time)
- Journaling (adding and editing entrie to support self-reflection and emotional processing)
- Motivational Quote Generator (Provides users with inspirational quotes personalized based on chosen categories such as stress management, self- compassion, work, relationships, etc.)
- Medication Reminder (Allows users to set an alarm and reminder for medication)

1.4 Definitions, Acronyms, and Abbreviations

- GUI - Graphical User Interface: The visual elements on a screen that users interact with, including buttons, menus, and icons
- API - Application programming Interface: A set of protocols and tools for building software applications that enable communication with external systems
- HIPAA - Health Insurance Portability and Accountability Act: United States law that protects sensitive patient health information
- SSL/TLS - Secure Sockets Layer/ Transport Layer security: Encryption protocols for data transfer between a web server and browser
- TBD - To be determined: Placeholder indicating a specific detail requires further specification
- WCAG - Web Content Acessability Guidelines: International standards for making web content accessible to users with disabilities

1.5 References

- Health Insurance Portability and Accountability Act (HIPAA)
- W3C Web Acessibility Guidelines
- Software Engineering, Ian Sommerville, Addison Wesley, 10th Edition

Bryam Ochoa

2. Overall Description

2.1 System Analysis

Goal: to offer a user-friendly and intuitive interface, promoting mental health awareness and providing tools for self-care.

Problem & Solutions

User Interface Design: Designing an intuitive and visually appealing user interface that is easy to navigate and provides a positive user experience can be a challenge. It requires understanding user preferences, incorporating user feedback, and ensuring compatibility across various Android devices and screen sizes.

Conduct user research and usability testing to gather feedback and iterate on the design. Follow Android design guidelines and use UI frameworks or libraries to create a consistent and visually appealing interface across different devices.

Data Security and Privacy: Handling sensitive user data, such as mood records and medication information, requires implementing robust security measures to protect user privacy and comply with data protection regulations.

Implement secure storage mechanisms to protect user data. Adhere to industry best practices and guidelines for data security and privacy, such as using secure communication protocols and regularly updating security measures.

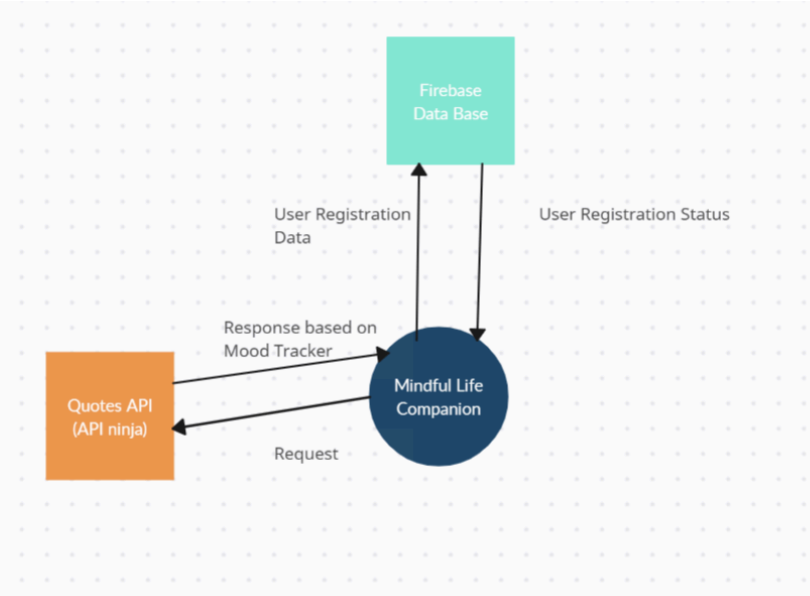
Integration with External APIs: Integrating external APIs for features like motivational quote generation or medication reminders may involve technical complexities, such as handling authentication, data retrieval, and ensuring seamless integration with the app's functionalities.

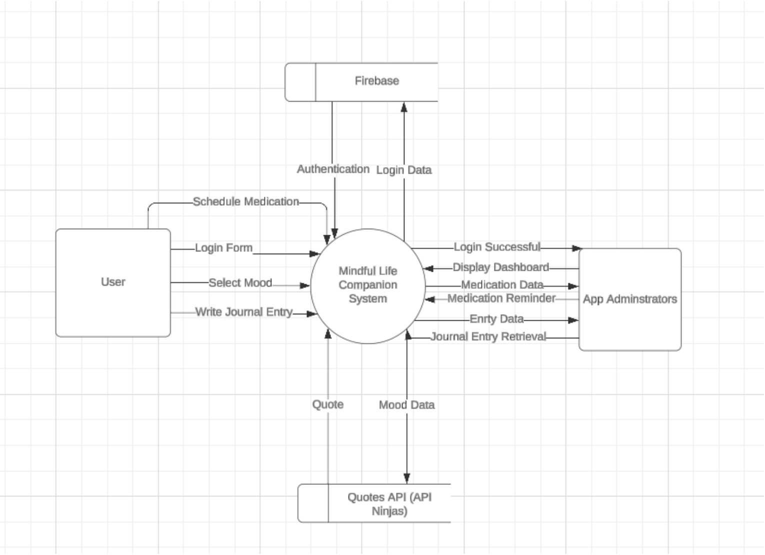
Thoroughly research and select reliable APIs that provide the required functionalities. Follow API documentation and best practices for authentication and data retrieval. Implement error handling and fallback mechanisms to handle API failures gracefully.

Device Compatibility: Ensuring the app functions correctly on a wide range of Android devices, operating system versions, and screen resolutions can be a challenge due to fragmentation in the Android ecosystem.

Conduct extensive testing across various Android devices, screen resolutions, and OS versions to identify and resolve compatibility issues. Use responsive design techniques and adapt the app layout dynamically based on the device's screen size and resolution.

2.2 Product Perspective

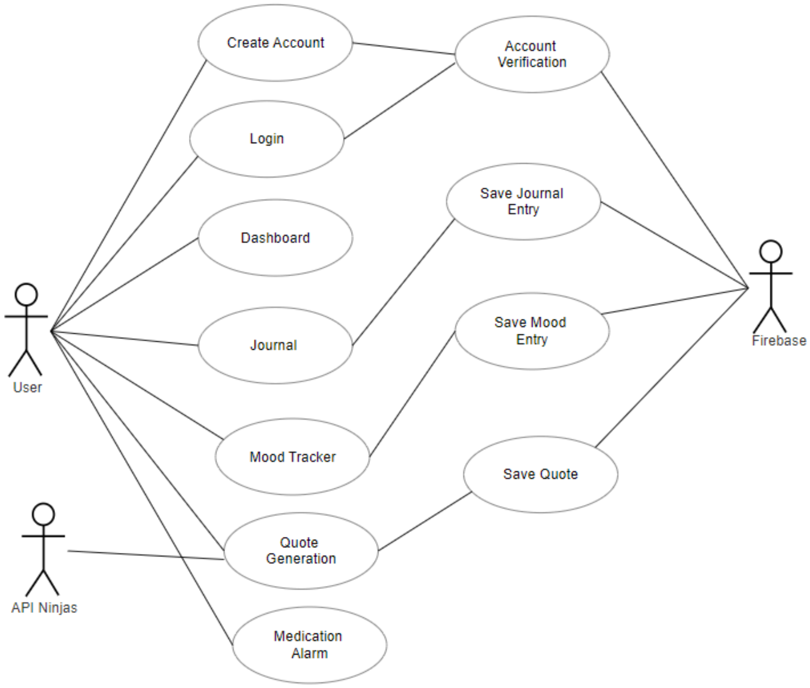


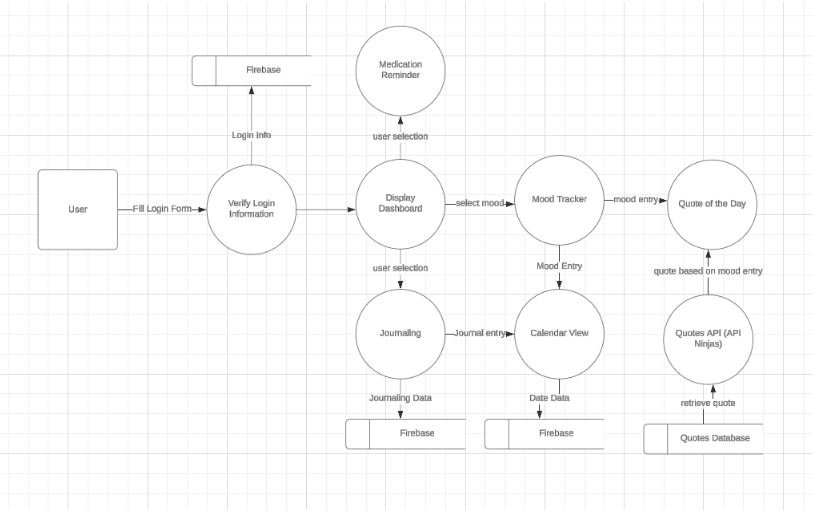


In the context of the mental health app being developed, the software can be considered as an independent product rather than a component of a larger system. It is designed to operate as a standalone application on Android devices. The app will have interactions with the user and may interact with external APIs to provide certain features like motivational quote generation.

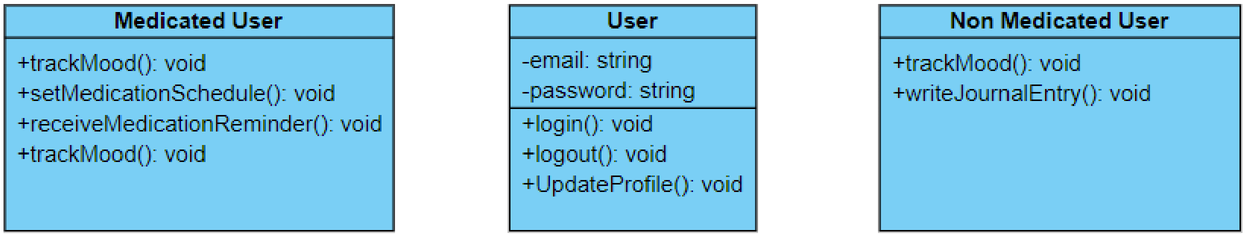
When comparing the mental health app to similar systems on the market, it is important to note that there are existing mental health apps available. However, the motivation for creating this software lies in providing a comprehensive and user-friendly platform specifically tailored for mental health management. The app aims to combine multiple features, such as mood tracking, health data management, journaling, motivational quote generation, and medication reminders, into a single cohesive application.

2.3 Product Functions





2.4 User Classes and Characteristics



MedicatedUser:

Characteristics: This class represents users who are on medication and utilize the app for medication tracking and management in addition to other features.

Pertinent Characteristics: Medicated users can use setMedicationSchedule to receive medication reminders (no input text required).

Subset of Product Functions: Medicated users can perform functions such as tracking mood, writing journal entries, setting medication schedules, and receiving medication reminders.

Importance: Medicated users are an important user class as they require specific features related to medication management.

NonMedicatedUser:

Characteristics: This class represents users who do not require medication tracking or management features.

Pertinent Characteristics: This user can use the Alarm Reminder to improve organization (user can input text for a reminder notification),

Subset of Product Functions: Non-medicated users can perform functions such as tracking mood and writing journal entries.

Importance: Non-medicated users may still find value in the app's other features, but they do not require medication-related functionality.

2.5 Operating Environment

The operating environment for the software will depend on the hardware platform, operating system, and other software components that it will be running on. Here's an overview of the operating environment based on the information provided:

Hardware Platform:

The software will be designed to run on devices compatible with the Android platform. The specific hardware requirements will depend on the Android Studio Hedgehog API 30, which supports Android 11 and above. It is essential to ensure that the software is compatible with the hardware capabilities of the intended devices, such as processor, memory, storage, and display.

Operating System:

The software will be compatible with Android 11 and above, based on the Android Studio Hedgehog API 30. It should be designed to leverage the features and capabilities provided by the Android operating system, ensuring proper integration and functionality.

Software Components and Applications:

The software will be developed using Android Studio, which provides a comprehensive development environment for Android applications. It will utilize various software components and libraries available in the Android SDK, including APIs for user interface design, data storage, connectivity, and security.

Coexistence with Other Software:

The software should be developed to coexist with other applications and services running on the Android platform. It should adhere to Android's guidelines for app integration, ensuring compatibility with other applications and avoiding conflicts.

Environment:

The software can be used in various environments, including but not limited to offices, homes, urban areas, and rural areas. It is designed to be used on Android devices, which are typically mobile and provide flexibility in terms of location and accessibility. The software should be adaptable to different network conditions, including both online and offline scenarios.

Availability:

The availability of resources such as internet connectivity, GPS, camera, and sensors will depend on the capabilities of the specific Android devices used. The software should gracefully handle scenarios where certain resources are not available or limited.

Security:

The software should incorporate appropriate security measures to protect user data and ensure the confidentiality and integrity of information. It should adhere to Android's security guidelines and best practices to mitigate potential security risks.

2.6 Design and Implementation Constraints

Hardware Limitations: The hardware limitations of the target devices can impact the software design and implementation. Developers need to consider the available processing power, memory capacity, storage capabilities, and network connectivity of the target devices to ensure optimal performance and compatibility.

Interfaces with Other Applications: If the software needs to interface or integrate with other applications or systems, there may be constraints imposed by the APIs, protocols, or data formats used by those applications. Compatibility and interoperability considerations need to be taken into account during the design and implementation process.

Parallel Operation: If the software is expected to support concurrent or parallel operations, developers need to consider the design and implementation of mechanisms such as multi-threading, synchronization, and resource management to ensure correct and efficient execution.

Higher-Order Language Requirements: If the software is required to be developed using a specific programming language (Kotlin) or development framework, developers must adhere to those requirements. Constraints related to language syntax, libraries, or frameworks may impact the development process and available development tools.

Reliability Requirements: The software may have specific reliability requirements, such as high availability, fault tolerance, or error recovery. These requirements may influence the design and implementation of features like error handling, exception management, backup mechanisms, or redundancy.

Safety and Security Considerations: If the software operates in safety-critical or security-sensitive environments, there may be constraints related to safety standards, access controls, encryption, or vulnerability assessments. Developers need to address these constraints to ensure the software meets the necessary safety and security requirements.

Memory Constraints: The available memory on the target devices can be a constraint for the software. Developers must optimize memory usage, manage data structures efficiently, and consider memory limitations to ensure the software performs well without exhausting available resources.

2.7 User Documentation

We are thinking of possibly making a tutorial video to showcase the app's functionality and features. An online guide such as an FAQ might also be helpful. Release Notes will be provided to help users understand differences between software versions.

2.8 Assumptions and Dependencies

Changes in Technology: Advancements in technology, such as new hardware platforms, operating systems, or programming languages, may require adjustments to the software requirements. For example, if a new version of an operating system introduces significant changes, the software may need to be updated to ensure compatibility and leverage new features. As an example we started working in Android Studio Hedgehog and Iguana released a few days ago.

Time Limitations: If there are time constraints, some requirements may need to be deferred or outright cancelled if they are not properly accounted for.

2.9 Apportioning of Requirements

We need to enhance user experience so we are currently working on bettering the UI and doing dashboard mockups. The rest of the key features are yet to be implemented and debugging is a crucial step.

3. External Interface Requirements

3.1 User Interfaces

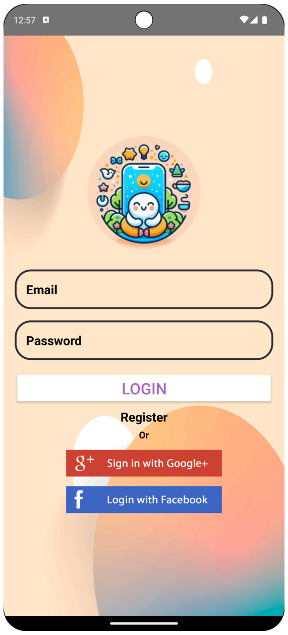
The Mindful Life Companion App will employ a mobile interface designed for Android devices. The interface will adhere to Web Content Accessibility Guidelines (WCAG) to ensure accessibility for users with disabilities.

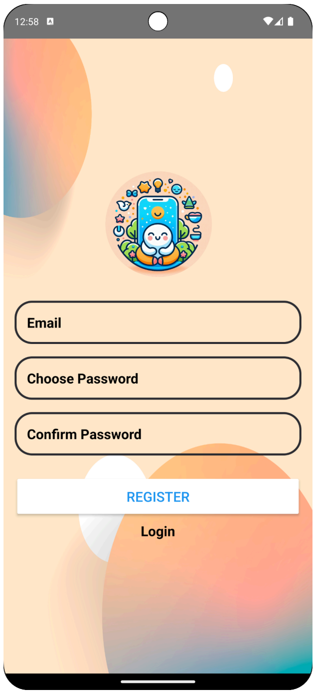
Standard Buttons and Functions: Common buttons and functions will appear consistently throughout the user interfaces. This may include standard buttons like "Register," "Login," and “Logout” along with functions like search, filtering, sorting, and navigation elements.

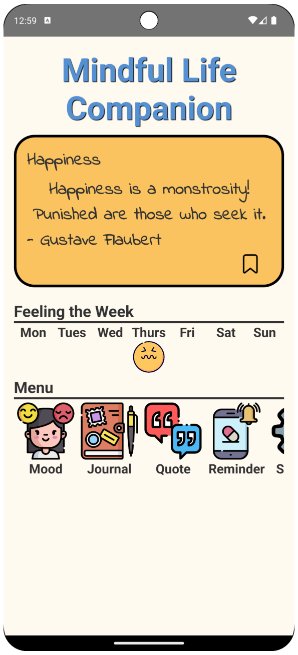
Error Message Display Standards: The user interfaces will follow consistent standards for displaying error messages. Clear and informative error messages will be provided to users in case of invalid inputs, system errors, or any other exceptional scenarios.

GUI Standards and Style Guides: The user interfaces will adhere to established GUI standards and follow the product family style guides, ensuring consistency and familiarity across different screens. This may include consistent color schemes, typography, iconography, and layout principles.

Touch Screen Interaction: The mobile interface will be optimized for touch screen interaction. Users will be able to navigate through screens, interact with buttons and controls, and perform gestures like swiping, tapping, and pinching to zoom.

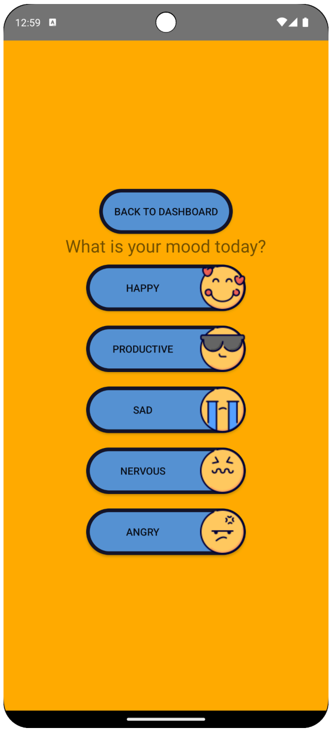




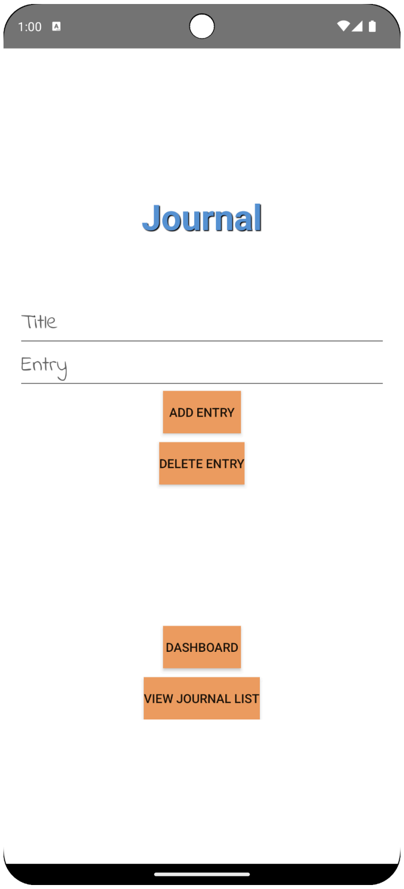


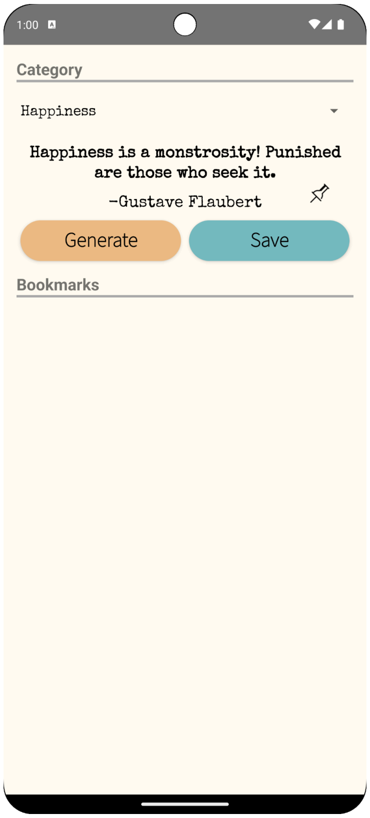
Happiness

Happiness is a monstrosity!
Punished are those who seek it.
- Gustave Flaubert



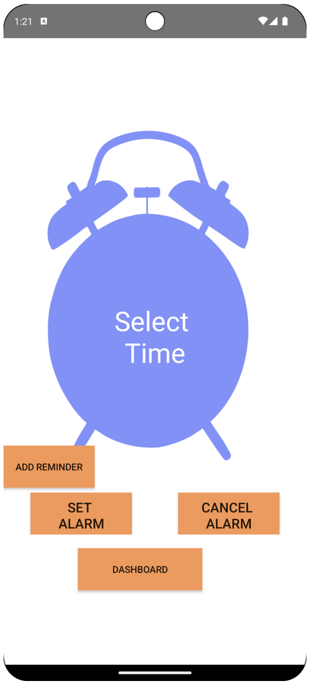
What is your mood today?





Happiness is a monstrosity! Punished are those who seek it.

-Gustave Flaubert



3.2 Hardware Interfaces

The app will interact with the hardware components of Android devices, including touchscreens, speakers, and sensors. Specific hardware requirements, such as processor specifications or memory capacity, will be determined by the Android Studio Hedgehog API 30, supporting Android 11 and above.

3.3 Software Interfaces

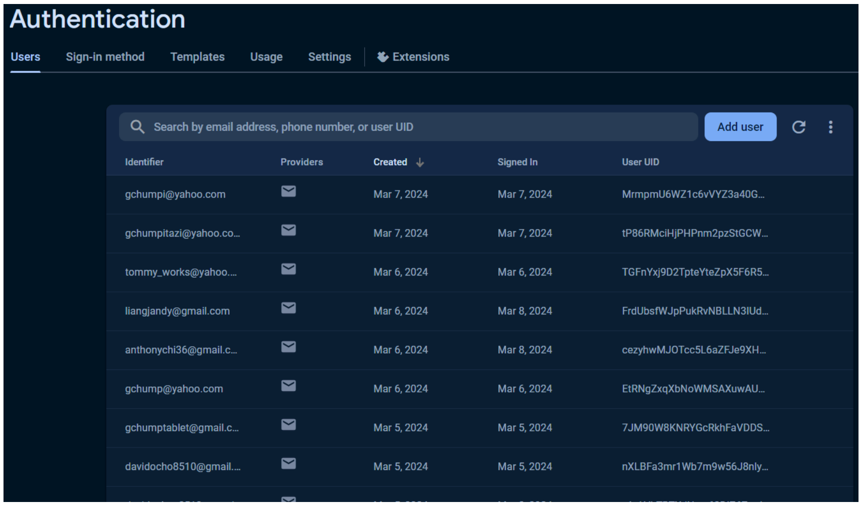
The app will utilize external software products and interfaces to provide certain functionalities. These include:

- APIs: Integration with external APIs for features like motivational quote generation and medication reminders.
- Android Studio: Development environment for Android applications, leveraging various software components and libraries available in the Android SDK.

3.4 Communications Interfaces

Communication functions required by the app include data synchronization, email notifications, and server interactions for data transfer. The app will utilize secure communication protocols such as SSL (Secure Socket Layer) /TLS (Transport Layer Security) for encrypted data transmission. It will also adhere to industry standards for communication security and encryption to ensure the confidentiality and integrity of user data.

Firebase

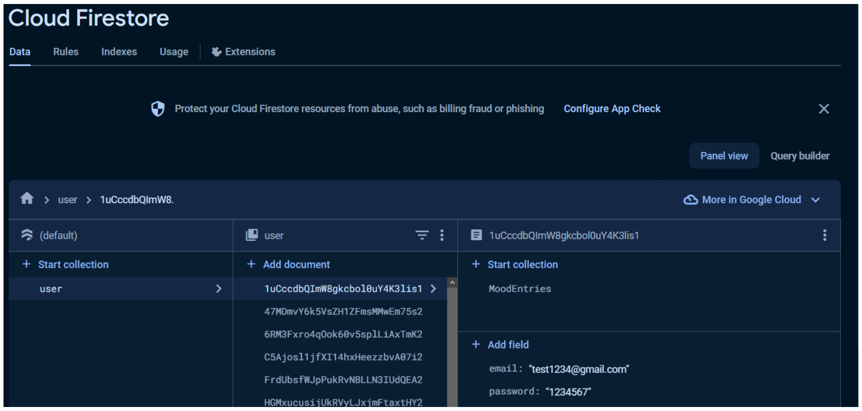


Authentication

Users | Sign-in method | Templates | Usage | Settings | Extensions

Search by email address, phone number, or user UID | Add user

Identifier	Providers	Created ↓	Signed In	User UID
gchumpi@yahoo.com	✉	Mar 7, 2024	Mar 7, 2024	MrmpmU6WZ1c6vVYZ3a40G…
gchumpitazi@yahoo.co…	✉	Mar 7, 2024	Mar 7, 2024	tP86RMciHjPHPnm2pzStGCW…
tommy_works@yahoo.…	✉	Mar 6, 2024	Mar 6, 2024	TGFnYxj9D2TpteYteZpX5F6R5…
liangjandy@gmail.com	✉	Mar 6, 2024	Mar 8, 2024	FrdUbsfWJpPukRvNBLLN3IUd…
anthonychi36@gmail.c…	✉	Mar 6, 2024	Mar 8, 2024	cezyhwMJOTcc5L6aZFJe9XH…
gchump@yahoo.com	✉	Mar 6, 2024	Mar 6, 2024	EtRNgZxqXbNoWMSAXuwAU…
gchumptablet@gmail.c…	✉	Mar 5, 2024	Mar 5, 2024	7JM90W8KNRYGcRkhFaVDDS…
davidocho8510@gmail.…	✉	Mar 5, 2024	Mar 5, 2024	nXLBFa3mr1Wb7m9w56J8nly…



Cloud Firestore

Data | Rules | Indexes | Usage | Extensions

Protect your Cloud Firestore resources from abuse, such as billing fraud or phishing Configure App Check

Panel view | Query builder

user > 1uCccdbQImW8. More in Google Cloud

(default)
+ Start collection
user

user
+ Add document
1uCccdbQImW8gkcbol0uY4K3lis1
47MDmvY6k5VsZH1ZFmsMMwEm75s2
6RM3Fxro4qOok60v5splLiAxTmK2
C5Ajosl1jfXI14hxHeezzbvA07i2
FrdUbsfWJpPukRvNBLLN3IUdQEA2
HGMxucusijUkRVyLJxjmFtaxtHY2

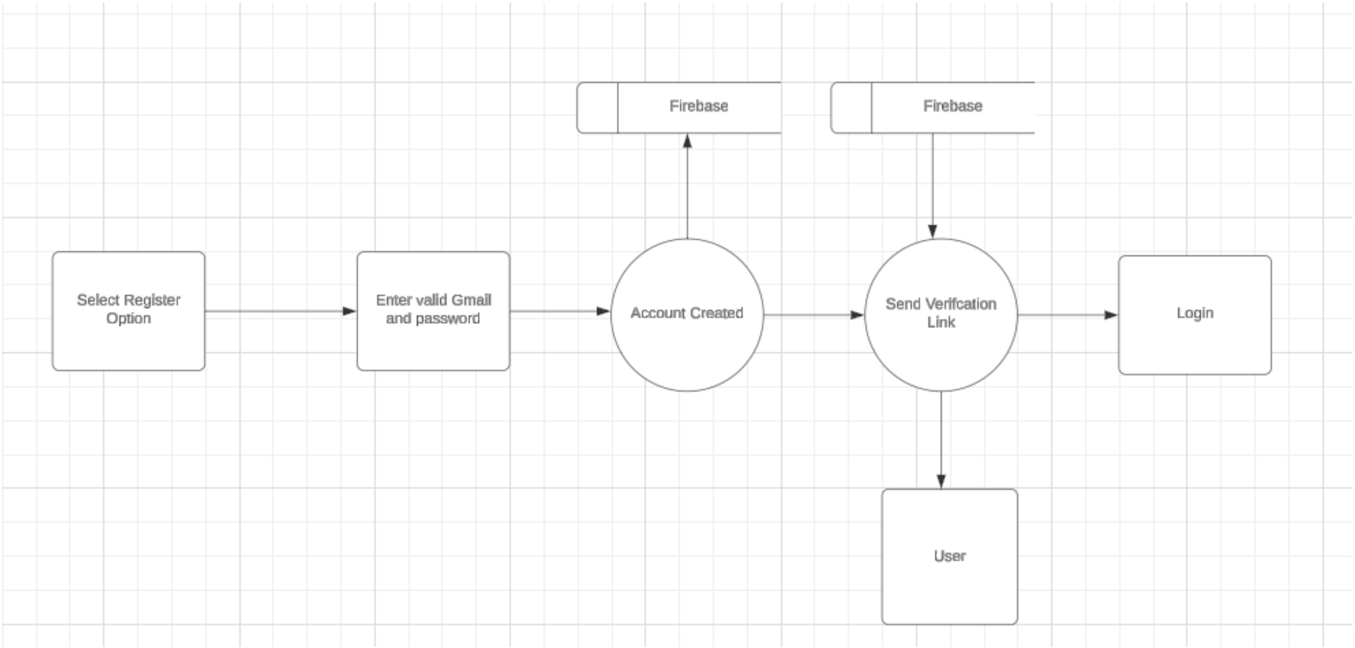
1uCccdbQImW8gkcbol0uY4K3lis1
+ Start collection
MoodEntries
+ Add field
email: "test1234@gmail.com"
password: "1234567"

4. Requirements Specification

4.1 Functional Requirements

4.1.1 The system shall allow a user to register a new account. (Actor: User, Why: to gain access to system functionalities

4.1.1 [Register DFD-2 Andy Liang (firebase =database)

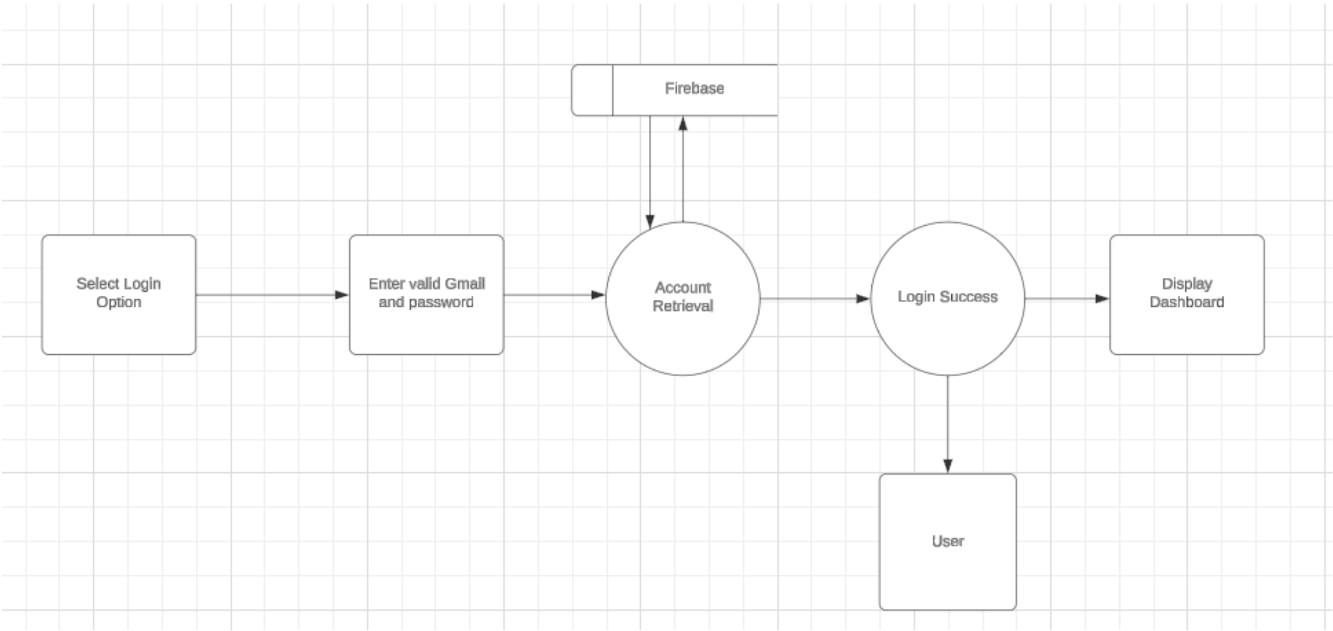


Child Requirements:

- 4.1.1.1 Collect User Information: The system shall collect user information during registration. Information collected includes email, password, user name, and may collect relevant fields if required.
- 4.1.1.2 Validate User information: The system shall validate all entered user information:
  - 4.1.1.2.1 Email: Email addresses also act as usernames, so they must be unique and be verified through email confirmation link
  - 4.1.1.2.2 Password: Password must meet a minimum length and must only be formed with valid characters the system allows
- 4.1.1.3 Store User Information: The system shall securely store user information used for registration and send out an email verification link upon registration
- 4.1.1.4 Generate Account: The system shall create a new user account upon successful verification

4.1.2 [Login DFD-2 Andy Liang

4.1.2 The system shall allow users to login to an existing account.(Actor: User, Why: to gain access to system functionalities and access previously saved session data



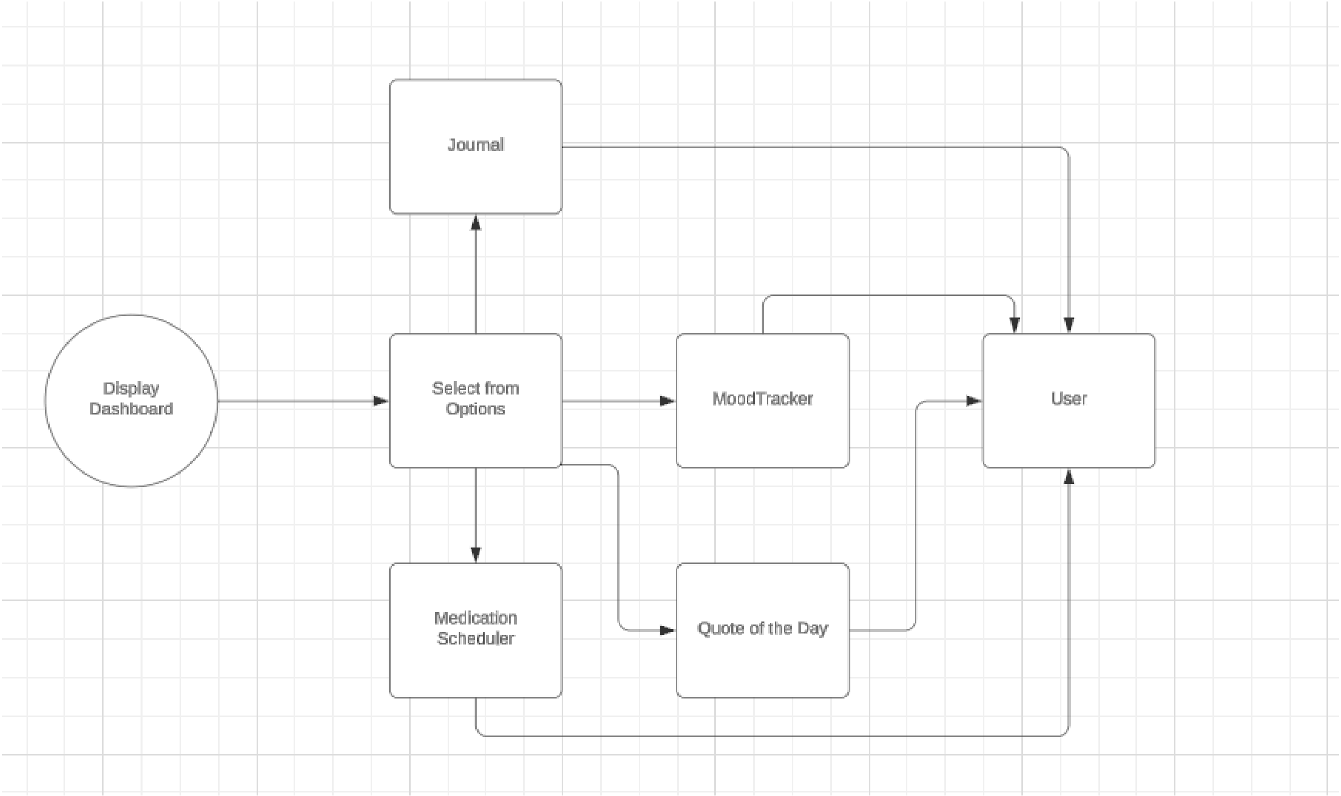
Child Requirements:

- 4.1.2.1 Collect User Input: The system shall allow users to enter their email and password to login
- 4.1.2.2 Validate User Information: The system shall validate all entered user information such that an email address and password must match a verified entry for Authentication
- 4.1.2.3 Account Retrieval: The system shall retrieve a users profile with previous saved data upon Login Success
- 4.1.2.4 Page Redirect: The system shall redirect users to landing page (Dashboard) upon successful login

Luis Sanchez

4.1.3 [Dashboard DFD-2

4.1.3 The system shall allow users to focus on offering users a friendly and feature-rich dashboard experience empowering users to engage with various mindfulness tools and resources seamlessly.



4.1.3.1 Display Dashboard:

- Shows a dashboard after user login with selectable options.

4.1.3.2 Option Selection:

- Users can choose from Journal, Medication Scheduler, Mood Tracker, or Quote of the Day.

4.1.3.2.1 Journal:

- Allows users to record thoughts and feelings.

4.1.3.2.2 Medication Scheduler:

- Enables Users to manage medication schedules.

4.1.3.2.3 Mood Tracker:

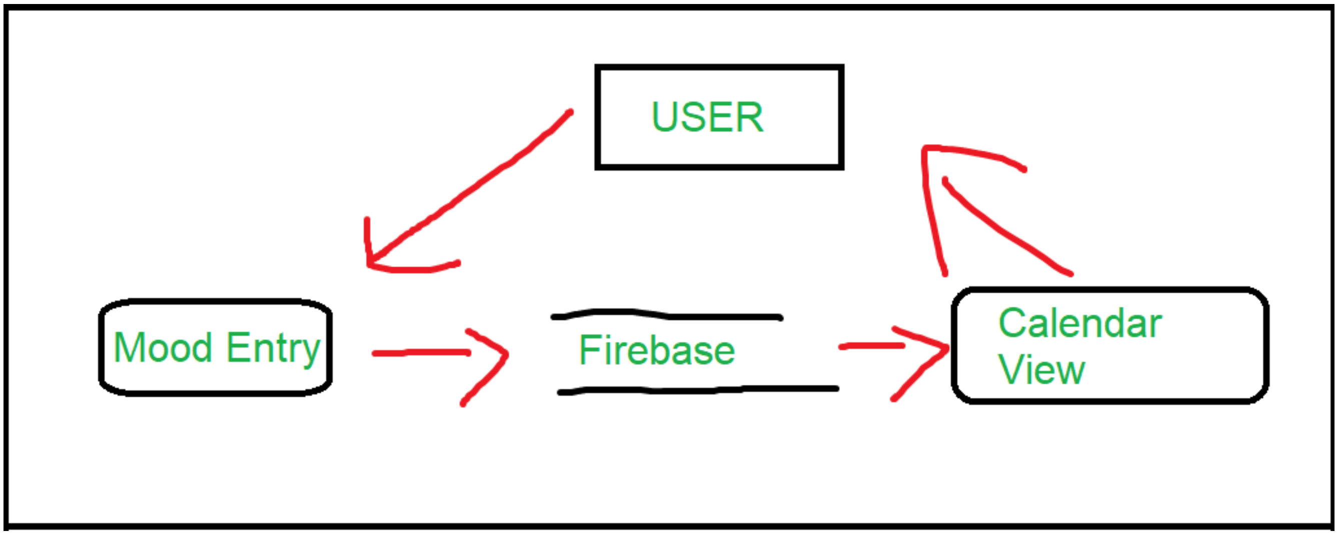
- Lets user log their mood to track emotional well-being.

4.1.3.2.4 Quote of the Day:

- Presents a daily inspirational quote.

4.1.4 [Mood Tracker DFD-2 Giovanni Chumpitazi , Anthony Chieng]

4.1.4 The System shall allow users to set and track their mood daily. (Actor: User, Why: to provide feedback to the user about their mental wellness)



Child **Requirements:**

4.1.4.1 Quote API:

- Provides the quote of the day

4.1.4.2 Quote of the day :

- Presents an inspirational quote for each day.

4.1.4.3 Mood entry:

- Allows users to record their mood.

4.1.4.4 Mood Display:

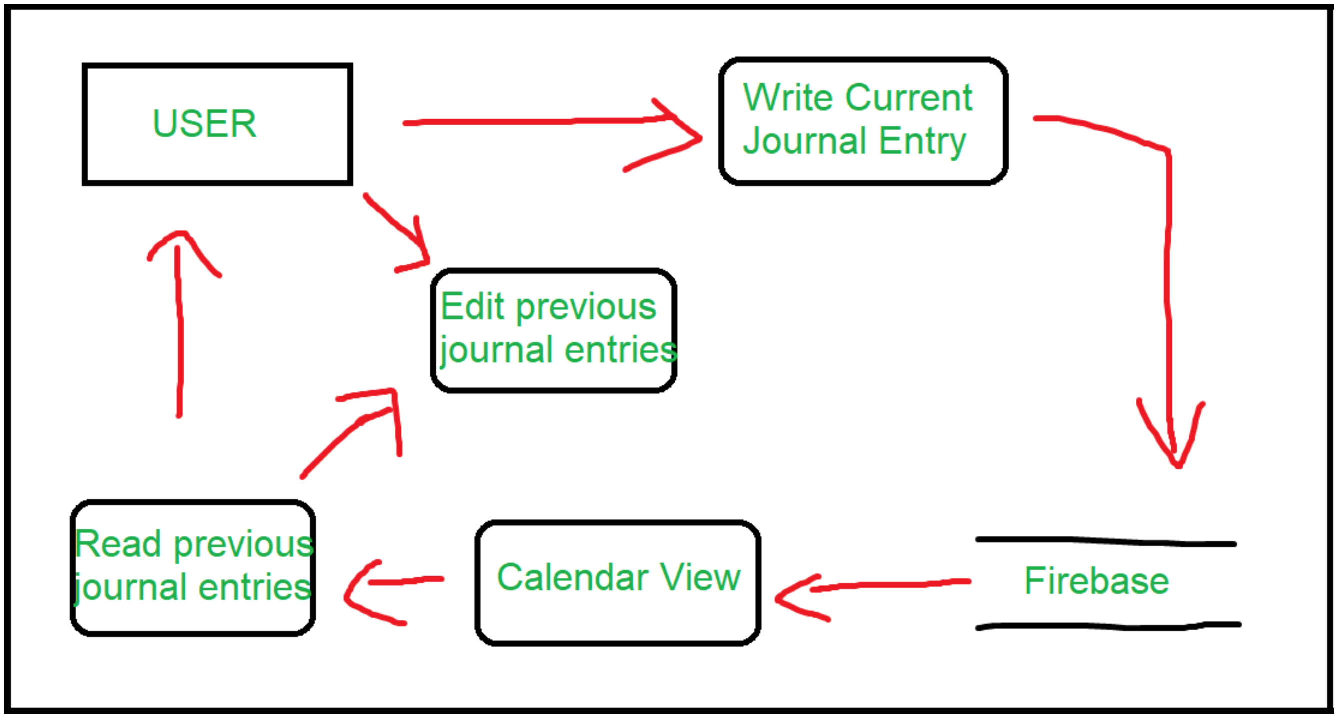
- Enables Users to see their mood.

4.1.4.5 Calendar View:

- Displays the monthly calendar

4.1.5 [Journaling DFD-2, Anthony Chieng]

4.1.5 The System shall allow users to journal their thoughts daily. (Actor: User, Why: to have access to their thinking on certain days and look back on them)


USER
Write Current Journal Entry
Edit previous journal entries
Read previous journal entries
Calendar View
Firebase

Child Requirements:

4.1.5.1 Write journal entry from user input: The system shall take user input in the journaling page after it has been selected.

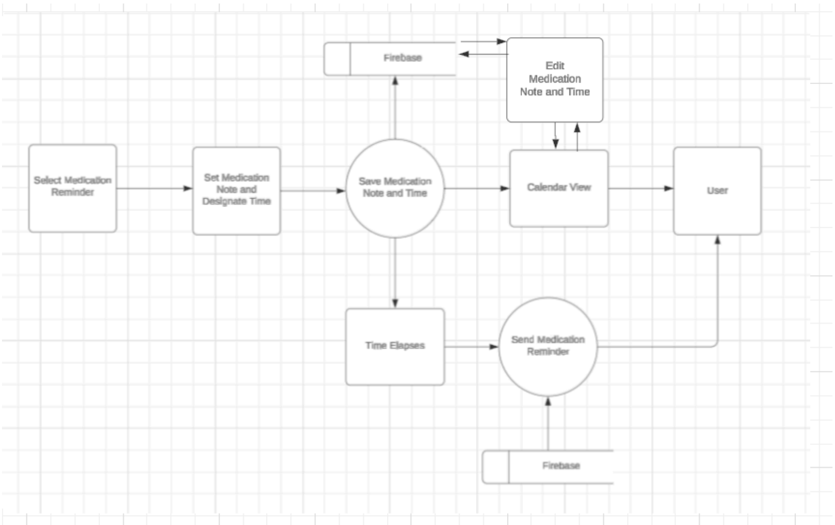
4.1.5.2 Save and place entry into calendar view: The system shall save the journaling entry and store it to Firebase.

4.1.5.3 View journal entry list: The system shall allow the user to view current and previous journal entries generated by Firebase.

4.1.5.4 Edit journal entries: The system shall allow the user to edit both current and previous journal entries.

4.1.6 [Medication Reminder DFD-2 Bryam Ochoa

4.1.6 The System shall allow users to set medication reminders. (Actor: User, Why: to allow users on medication to better track their own schedule for taking much needed medicine)


Firebase
Edit Medication Note and Time
Select Medication Reminder
Set Medication Note and Designate Time
Save Medication Note and Time
Calendar View
User
Time Elapses
Send Medication Reminder
Firebase

4.1.6.1 Write medication entry based on user input: The system shall take user input in the medication reminder page after it has been selected with a limit of 100 characters and a necessary time input.

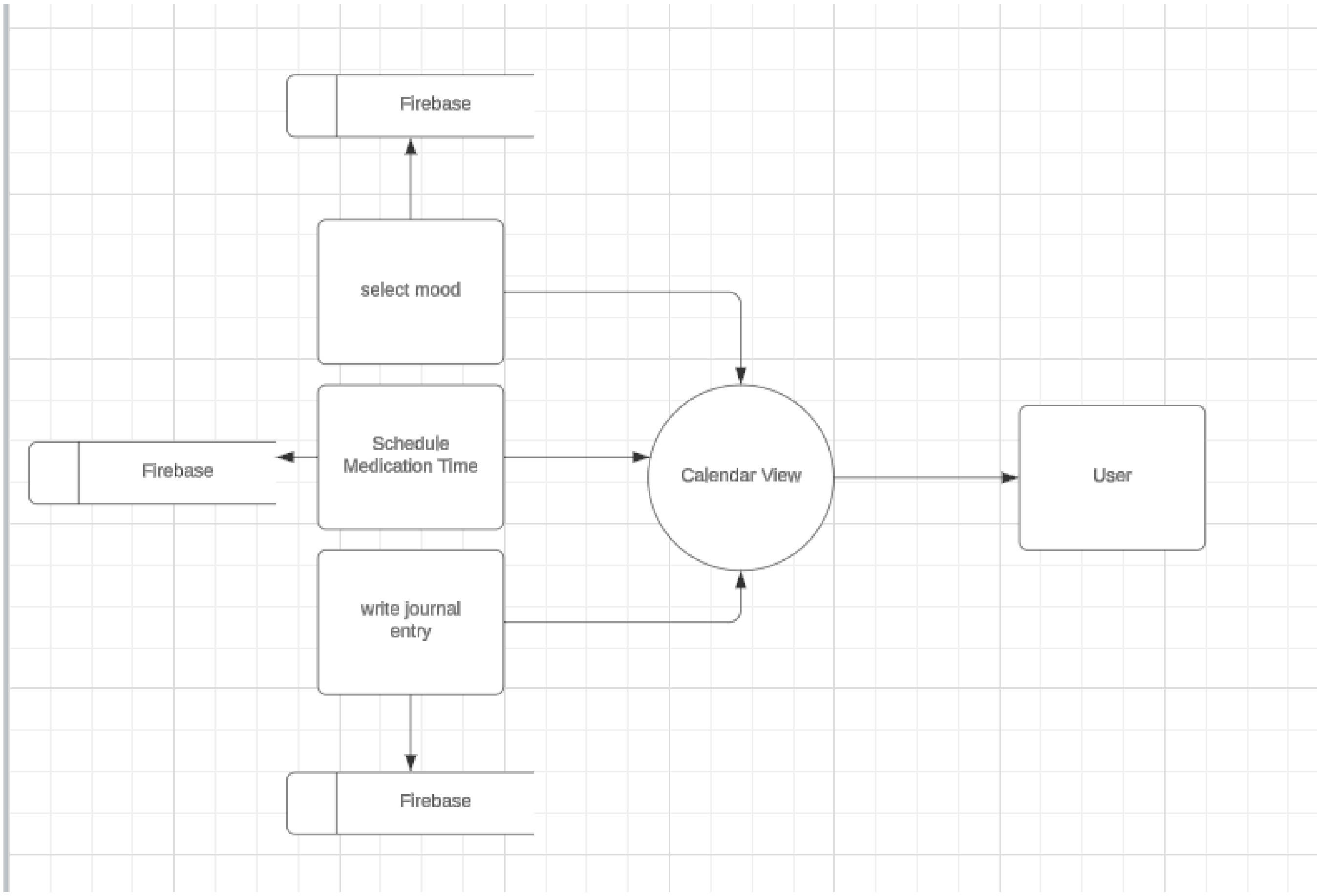
4.1.6.2 Save and place entry into calendar view: The system shall save the medication entry and store it to Firebase

4.1.6.3 Edit entry data from calendar view: The system shall allow the user to edit current medical reminder saved in firebase and override it with new information

4.1.6.4 Send out reminder to user’s device : The system shall retrieve its note and time data from firebase to send out a medication reminder when the designated time is reached

4.1.7 [Calendar View DFD-2 Emily Perez

4.1.7 The System shall allow users to view inputted data in a calendar view. (Actor: User, Why: to provide users an easier time viewing mental health changes over a period of time)



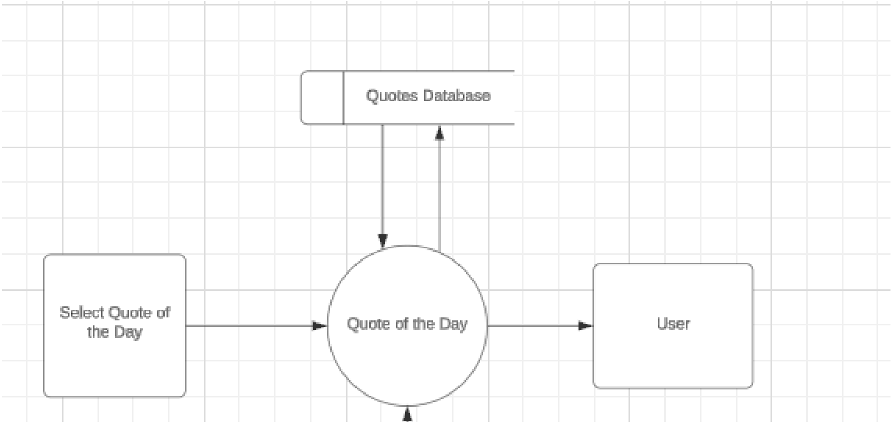
4.1.7.1 Mood Data viewable from Calendar View: Current and past mood data saved in firebase will appear in the app’s calendar view

4.1.7.2 Medication Reminder viewable/editable from Calendar View: Current medication data saved in firebase will appear in the app’s calendar view and will be able to be edited

4.1.7.3 Journal entries viewable/editable from Calendar View: Current and past journal entry data saved in firebase will appear in the app’s calendar view and will be able to be edited

4.1.8 [Quote of the Day DFD-2 Tommy Works

4.1.8 The System shall allow users to get a generated quote. (Actor: User, Why: if users are feeling down or need a way to get invigorated, a quote can help them if even a bit)



4.1.8.1 Quote Data generated from API Ninja’s Quotes API: A quote will be generated using a Quote API that is connected to a quotes database

4.2 External Interface Requirements

This contains a detailed description of all inputs into and outputs from the software system. It complements the interface descriptions in section 3 but does not repeat information there. Remember section 3 presents information oriented to the customer/user while section 4 is oriented to the developer.

It contains both content and format as follows:

- Name of item
- Description of purpose
- Source of input or destination of output
- Valid range, accuracy and/or tolerance
- Units of measure
- Timing
- Relationships to other inputs/outputs
- Screen formats/organization
- Window formats/organization
- Data formats

- Command formats
- End messages

4.3 Logical Database Requirements

This section specifies the logical requirements for any information that is to be placed into a database.

This may include:

- Types of information used by various functions.
- Frequency of use
- Accessing capabilities
- Data entities and their relationships
- Integrity constraints
- Data retention requirements

If the customer provided you with data models, those can be presented here. ER diagrams (or static class diagrams) can be useful here to show complex data relationships. Remember a diagram is worth a thousand words of confusing text.

4.4 Design Constraints

Specify design constraints that can be imposed by other standards, hardware limitations, etc. This should be a more technical description of the overview given in section 2.5.

Tommy W.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

-For the mental health app, the following performance requirements are specified:

-The app should support a minimum of 100 users (Hypothetical) active users simultaneously.

-95% of user interactions, such as submitting forms or accessing resources, should be processed in less than 2 seconds.

-The app should be capable of handling a peak workload of 50 transactions per minute during high traffic periods

5.2 Safety Requirements

-Safety is a critical consideration for the mental health app. The following safety requirements should be met:

-Users' sensitive information, such as personal health data or conversations with mental health professionals, must be securely encrypted to prevent unauthorized access.

-The app should include safety features such as emergency contact integration or crisis hotline access for users in distress.

-Compliance with relevant regulations such as HIPAA (Health Insurance Portability and Accountability Act) should be ensured to protect users' privacy and confidentiality.

5.3 Security Requirements

-Given the sensitive nature of the app's data, the following security requirements are specified:

-User authentication should be implemented using secure methods such as multi-factor authentication or biometric authentication.

-Data transmission between the app and server should be encrypted using industry-standard protocols like SSL/TLS.

-Regular security audits and vulnerability assessments should be conducted to identify and address potential security threats.

-Compliance with data protection regulations such as GDPR (General Data Protection Regulation) should be maintained.

5.4 Software Quality Attributes

-The mental health app should prioritize the following quality attributes:

-Usability: The app should be intuitive and user-friendly, with clear navigation and instructions for users experiencing mental health challenges.

-Reliability: The app should operate consistently without unexpected crashes or downtime, ensuring uninterrupted access to mental health resources.

-Maintainability: The app should be designed with modular architecture and clean code practices to facilitate easy updates and bug fixes.

-Accessibility: The app should be accessible to users with disabilities, complying with accessibility standards such as WCAG (Web Content Accessibility Guidelines).

5.5 Business Rules

-Business rules for the mental health app may include:

-Users should be age 13 and above.

-Users must agree to the app's terms of service and privacy policy before accessing mental health resources.

-Professional Guidance: The app should provide a disclaimer stating that it is not a substitute for professional medical or mental health advice. Users should be encouraged to consult qualified healthcare professionals for diagnosis, treatment, or specific advice related to their mental health concerns.

Continuous Improvement and Updates: The app should commit to continuous improvement, bug fixes, and updates to enhance user experience, address security vulnerabilities, and incorporate user feedback. Updates should be communicated to users, and efforts should be made to ensure a smooth transition between versions.

Tommy W.

6. Legal and Ethical Considerations

The mental health app must adhere to legal and ethical standards to ensure the well-being and rights of its users. The following considerations should be addressed:

Legal Compliance

- The app must comply with all relevant laws and regulations governing mental health services, data privacy, and digital health applications in the regions where it operates.

- Compliance with regulations such as HIPAA, GDPR, and any other local or international data protection laws is essential to safeguard users' privacy and confidentiality.

- Terms of service, privacy policy, and any other legal documents must be transparent, easily accessible, and regularly updated to reflect changes in regulations or app functionalities.

Ethical Guidelines

- The app should uphold ethical principles such as respect for user autonomy, beneficence, non-maleficence, and justice in its design, development, and operation.
- Respect for user privacy and confidentiality should be paramount, with clear policies and procedures in place to handle sensitive information responsibly.

- The app should avoid stigmatizing language or design elements that may contribute to discrimination or exacerbate mental health stigma.

- Ethical guidelines for content moderation and user interactions, including guidelines for handling sensitive topics or crises, should be established and enforced rigorously.

- Collaboration with mental health professionals, advocacy groups, and regulatory bodies can help ensure ethical practices and continuous improvement in addressing users' mental health needs.

By addressing these legal and ethical considerations, the mental health app can build trust with users and stakeholders while promoting the responsible use of technology in supporting mental well-being.

Appendix A: Glossary

Refer to section 1.4.

Appendix B: Analysis Models

No analysis models to be shown yet.

Appendix C: To Be Determined List

TBD - Hardware Platform Specifications

TBD - Tutorials Format and Posting Location

TBD - Online Guide Availability and Format

TBD - Future Version Requirements and Prioritization