

**SOFTWARE ENGINEERING II**

**GROUP 4**

**PROJECT DOCUMENTATION**

**PROJECT TITLE: SAFE HEALTH**

**NAME AND MATRICULATION NUMBER OF PEOPLE THAT PARTICIPATED**

OSAYAME EKHATOR VUG/CSC/21/5332

LUCAS ELIJAH LONGKET VUG/CSC/21/5252

ROLAND EMMANUEL TERRY VUG/CSC/21/5256

GREAT CYRIL OSINACHI AGBAEGBU VUG/CSC/21/5396

MICHEAL MONDAY DAVID VUG/CSC/21/5334

UDUMA CHIMDINMA KALU VUG/CSC/21/5220

WISDOM SABBATH SANDY VUG/CSC/21/5277

**TABLE OF CONTENTS**

1. **Introduction**

1.1 Purpose

1.2 Scope

* 1. Target function

1. **SYSTEM REQUIREMENT**
   1. Software requirements

2.2.1 Functional requirements

2.2.2 Non – functional requirements

2.2 Technical requirements

1. **Features** 
   1. User registration and authentication
   2. Symptom checker
   3. Health calculators
   4. Weight tracker
   5. Reminders
2. **Architecture and Design**

4.1 User interface design

1. **USER GUIDE**

5.1 Registration and login

1. **CONCLUSION**
2. **INTRODUCTION**

The Health App is a comprehensive application designed to assist users in managing their health and wellness. It offers a range of features including user registration and authentication, a symptom checker, health calculators, a weight tracker, and reminders. This documentation aims to provide an overview of each feature and guide developers in integrating them into the app.

**1.1 Purpose**

The purpose of this documentation is to provide a comprehensive guide on how to use the Health App effectively. It aims to assist users in understanding the app's features, functionalities, and their usage.

**1.2 Scope**

This documentation covers the core features of the Health App, including user registration and authentication, symptom checker, health calculators, weight tracker, and reminders.

**1.3 Target function**

This documentation is intended for both end-users and developers who want to understand how to interact with the Health App or integrate it into their existing systems.

1. **SYSTEM REQUIREMENTS**

System requirements refer to a detailed description of the capabilities, functionalities, and constraints that a software system must possess to meet the needs of its users and stakeholders. System requirements play a crucial role in the software development process as they serve as a foundation for designing, implementing, and testing the software system.

**2.1 Software requirements**

A software requirement refers to a specific function, behaviour, or quality that a software system must possess in other to fulfil the needs and expectations of its users and stakeholders.

**2.2.1 Functional requirements**

Functional requirements refer to the specific behaviours, features, and capabilities that a software system must possess to fulfil its intended purpose and satisfy the needs of its users.

**Functional requirements include:**

* Users should be able to create an account with a unique username and password.
* The app should validate user input and ensure the uniqueness of usernames.
* Users should be able to log in securely using their credentials.
* The app should provide a symptom checker feature that allows users to select the affected body part and receive possible diagnoses or suggestions.
* The app should provide accurate and reliable information to help users make informed decisions about their health.
* The app should include various health calculators, such as BMI (Body Mass Index) calculator, calorie calculator, heart rate calculator, etc.
* Users should be able to input relevant parameters and receive calculated results.
* The app should provide accurate and up-to-date formulas for the calculations.
* Calculators should be user-friendly and provide clear explanations of the results.
* The app should allow users to track their weight over time.
* Users should be able to enter their weight regularly and view a graphical representation of their weight history.
* Users should be able to set reminders to track progress.
* The app should include a reminder feature to help users manage their health-related tasks.
* Users should be able to set reminders for medication intake, doctor appointments, exercise routines, etc.
* Reminders should be customizable, allowing users to set specific times, and frequencies.

**2.2.2 Non-functional requirements**

Non-functional requirements specify the criteria that define the overall behaviour and characteristics of a software system, rather than the specific functions it performs.

**Non-functional requirements include:**

1. **Performance:**

* The app should have fast response times for user interactions, ensuring smooth and seamless user experience.
* The symptom checker should provide quick and accurate results, minimizing any delays or lags.
* Health calculators should provide instant calculations and display results promptly.

1. **Security:**

* User registration and authentication should be secure, protecting user information and preventing unauthorized access.
* The app should use encryption protocols to ensure the confidentiality and integrity of user data.
* Reminders should be securely stored and transmitted to prevent unauthorized access or tampering.

1. **Reliability:**

* The app should be always reliable and available for use, minimizing downtime or service interruptions.
* User registration and authentication should be robust and resistant to failures or errors.
* The app should handle errors and exceptions gracefully, providing informative error messages to users.

1. **Scalability:**

* The app should be designed to handle a growing number of users without significant performance degradation.
* The infrastructure supporting the app should be scalable to accommodate increased usage and user data.

1. **Usability:**

* The user interface should be intuitive, with clear navigation and user-friendly design.
* The symptom checker should provide easy-to-understand questions and instructions.
* Health calculators should have user-friendly interfaces, allowing users to input data easily.

1. **Compatibility:**

* The app should be compatible with various mobile platforms (iOS, Android) to reach a wider user base.
* It should support multiple screen sizes and resolutions to ensure a consistent user experience.

1. **Privacy:**

* The app should comply with applicable privacy regulations, protecting user privacy and data.
* User data should be anonymized and aggregated for research or analysis purposes to maintain privacy.

1. **Accessibility:**

* The app should be accessible to users with disabilities, adhering to accessibility standards for usability by individuals with visual, hearing, or motor impairments.
* Text should be adjustable, and the app should support assistive technologies like screen readers.

**2.2 Technical requirements**

This documentation represents information about the system code. They are maintained by programmers. The following frameworks were used during the software project:

**Frontend:** HTML, CSS, JAVASCRIPT, VUE.JS

**Backend:** NODE.JS (EXPRESS.JS, MongoDB)

1. **FEATURES**

System features refer to the specific functionalities or capabilities that a software system or application offers to its users. These features are designed and implemented to fulfill the requirements and needs of the intended users or stakeholders.

**3.1 User registration and authentication**

The User Registration and Authentication feature allow users to create an account and securely authenticate themselves within the Health App. This feature includes the following functionalities:

* **User Registration:** Users can sign up by providing their email address, username, and password. The app should validate the inputs and store them securely in the server.
* **User Login:** Registered users can log in using their email address and password. The app should authenticate the credentials and provide access to the user's personalized content.

**3.2 Symptom checker**

The Symptom Checker feature helps users identify potential health issues based on the symptoms they are experiencing. The feature includes the following components:

* **Symptom Input:** Users can enter the symptoms they are experiencing, by selecting the affected body part and receive possible diagnoses or suggestions.
* **Symptom Analysis:** The app should process the selected part and analyze for possible conditions. It should provide users with a list of potential conditions or diseases associated with their symptoms.

**3.3 Health calculators**

The Health Calculators feature provides users with various tools to measure and track their health indicators. The feature includes the following functionalities**:**

* **BMI Calculator:** Users can calculate their Body Mass Index (BMI) by entering their height and weight. The app should compute the BMI and provide the corresponding BMI category (e.g., underweight, normal weight, overweight).
* **Calories Calculator:** Users can estimate their daily calorie intake based on factors such as age, gender, weight, height, and activity level. The app should consider these parameters and provide a recommended calorie range.
* **Heart Rate Calculator:** Users can measure their heart rate by using their device's sensors or manually entering their pulse. The app should display the heart rate and provide a comparison to the normal range.
* **Water intake calculator:** Helps individuals determine the appropriate amount of water they should consume each day to stay properly hydrated.
* **One rep max calculator**: is a term used in strength training to represent the maximum amount of weight an individual can lift for a single repetition of a given exercise.
* **Blood alcohol concentration calculator:** Users estimate the level of alcohol in their bloodstream based on various inputs such as the number and type of drinks consumed, body weight, and time elapsed since drinking began.

**3.4 Weight tracker**

The Weight Tracker feature enables users to monitor and track their weight over time. It includes the following functionalities:

* **Weight Recording:** Users can input their weight at regular intervals, such as daily or weekly. The app should store the recorded weight entries and associate them with the user's profile.

**3.5 Reminders**

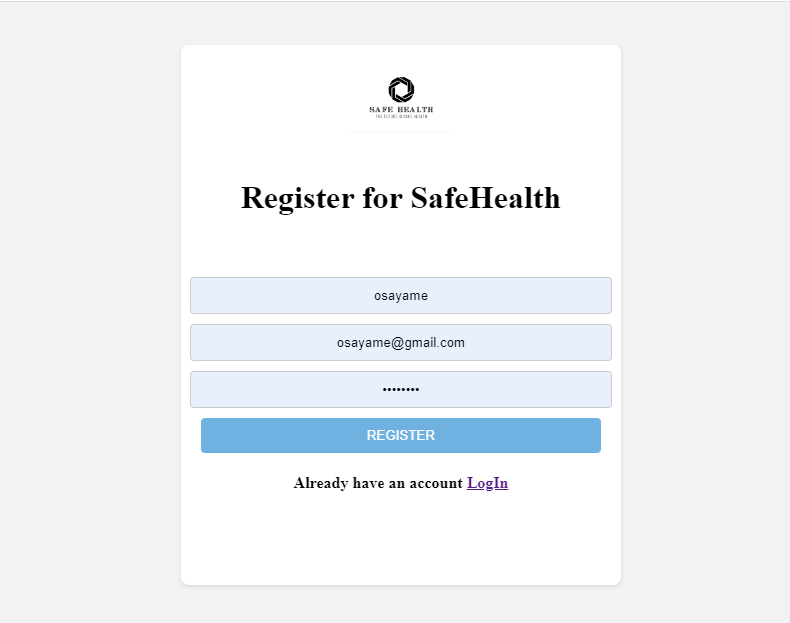
The Reminders feature assists users in managing their medication schedules, appointments, and other health-related tasks. It includes the following functionalities:

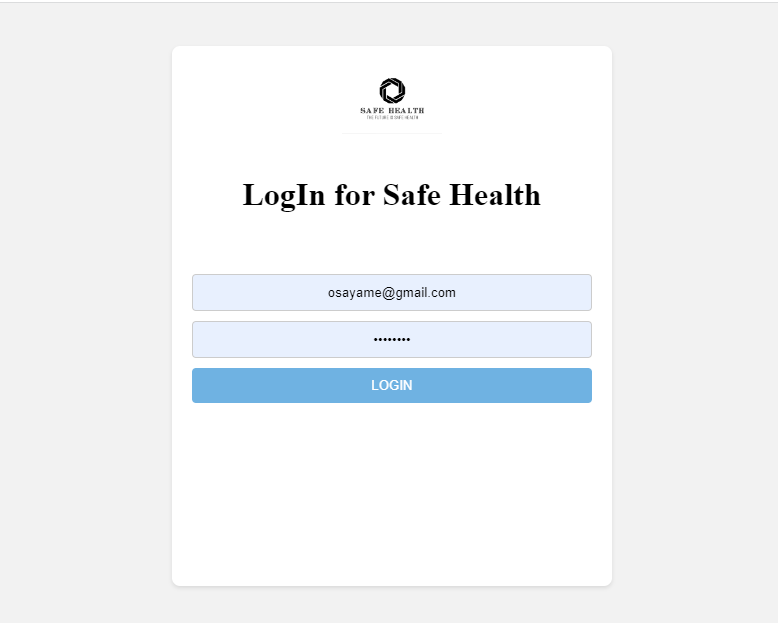
* **Task Creation:** Users can create reminders for taking medication, scheduling doctor appointments, or performing specific health-related tasks.
* **Task Management:** Users should have the ability to view, edit, or delete their scheduled tasks. The app should update the task status accordingly.

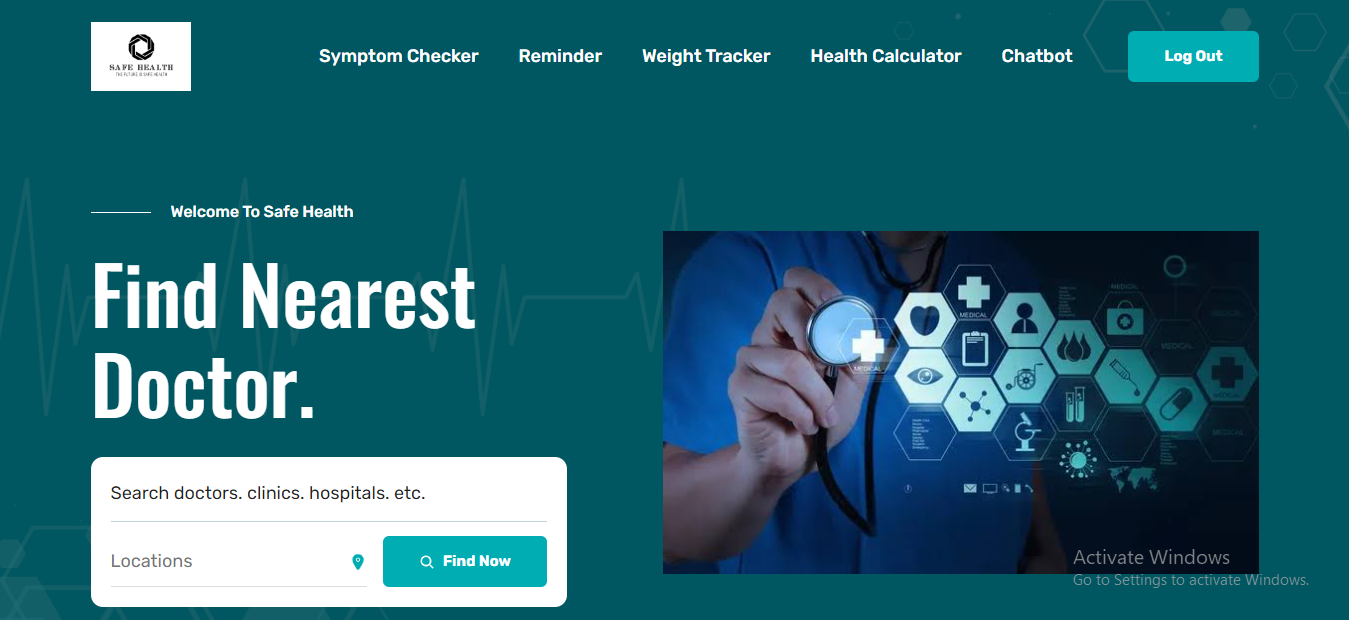
1. **ARCHITECTURE AND DESIGN**

**4.1 User interface design**

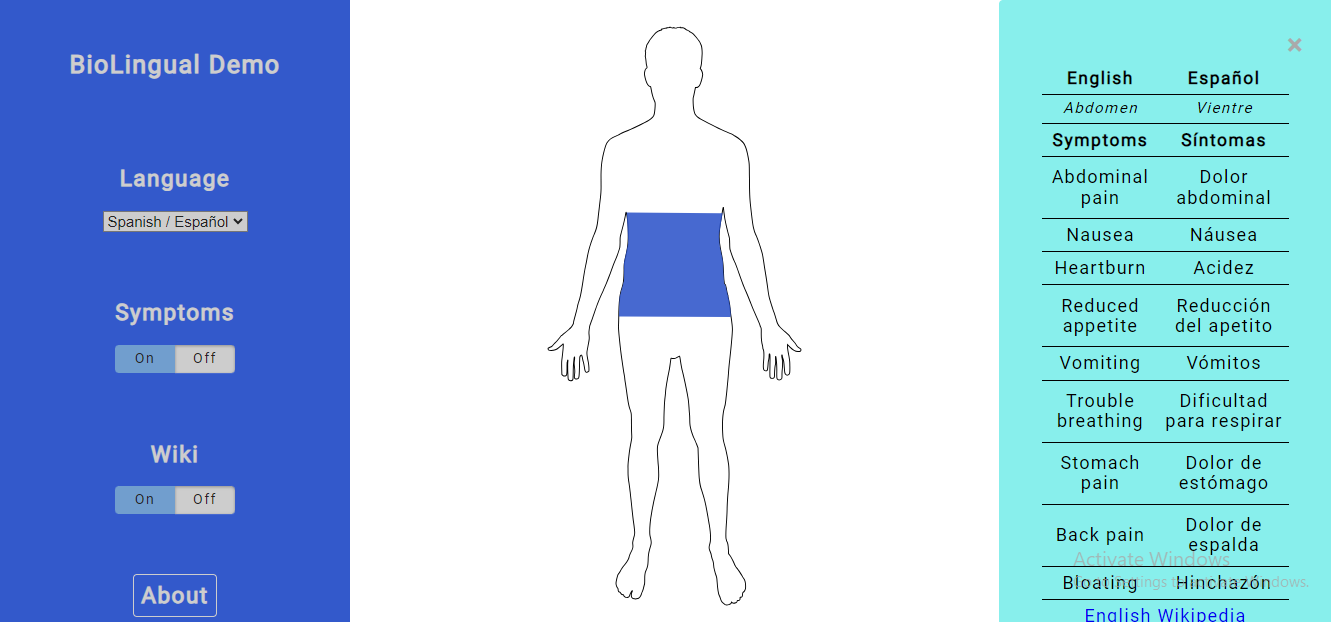


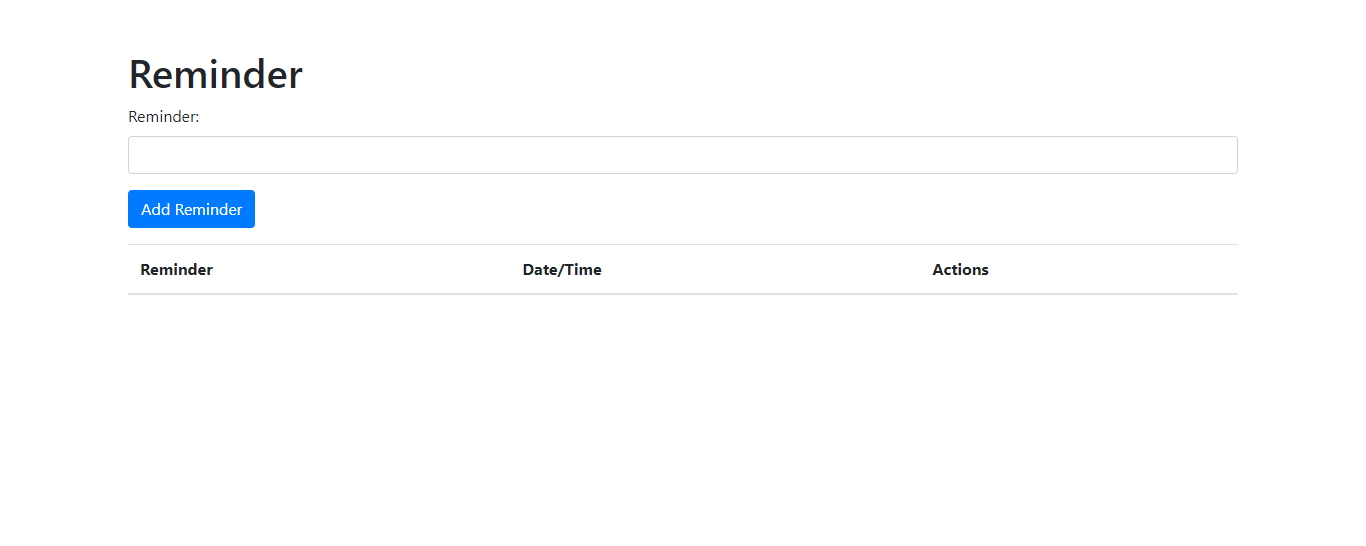


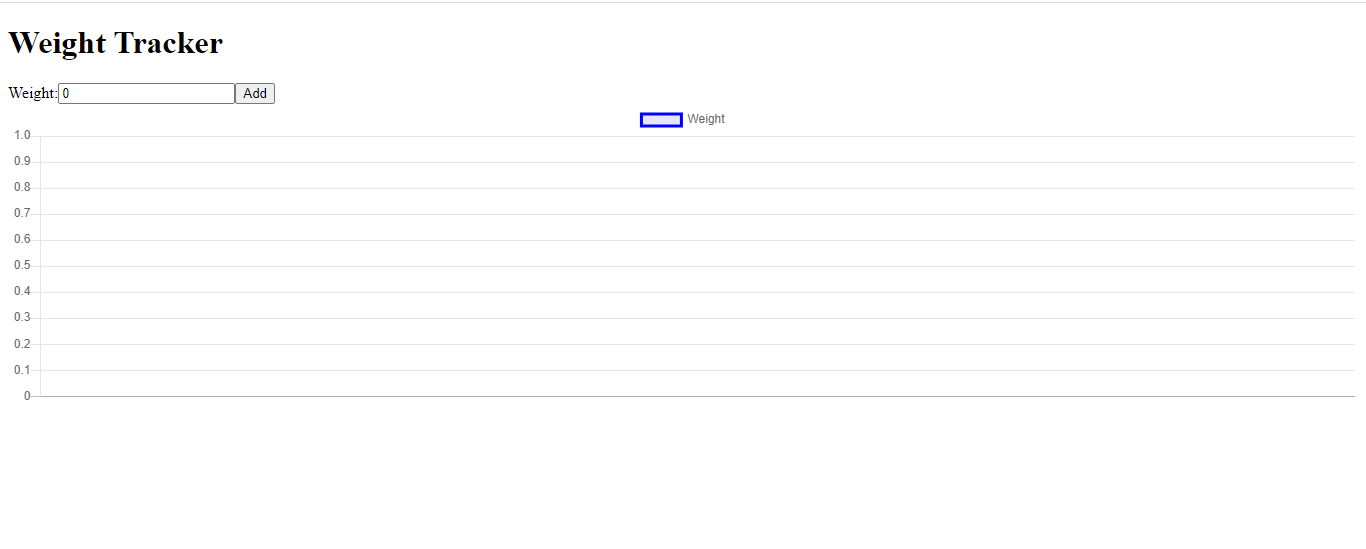


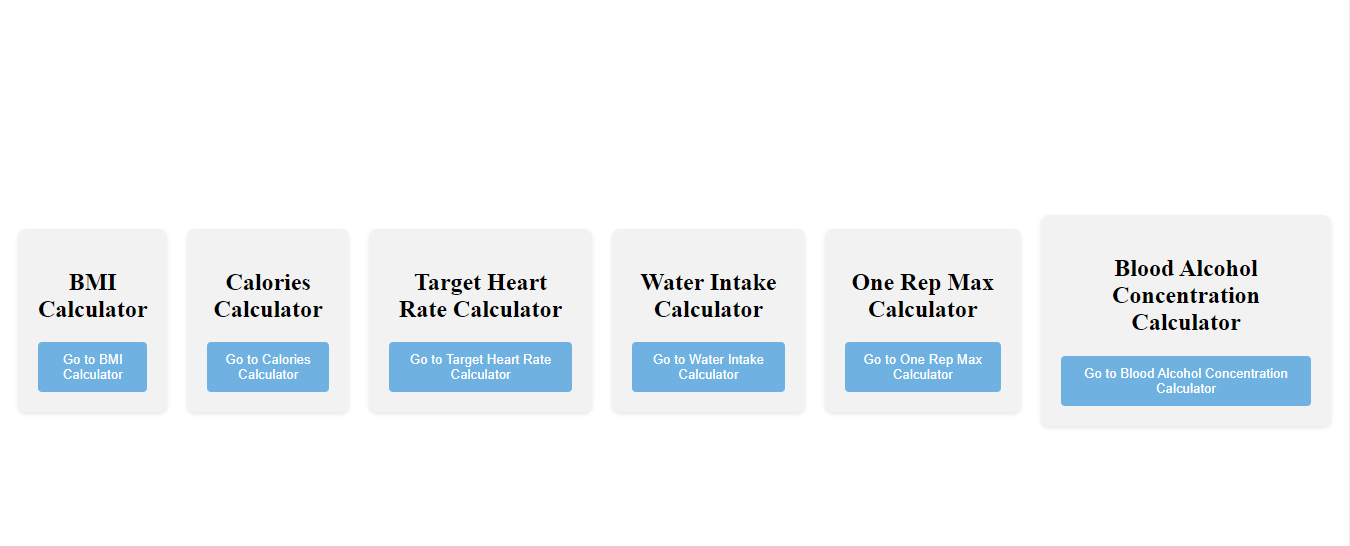












1. **USER GUIDE**

**5.1 Registration and login**

**Registration**

* To register as a new user:
* Launch the Health App.
* Tap on the "Register" button on the home screen.
* Provide the required information, including name, email address, and password.
* Agree to the terms and conditions, if prompted.
* Tap on the "Register" button to complete the registration process.

**Login**

* To log in as an existing user:
* Launch the Health App.
* Tap on the "Login" button on the home screen.
* Enter your registered email address and password.
* Tap on the "Login" button to proceed.

1. **CONCLUSION**

This software documentation provides an overview of the Health App's key features, including user registration and authentication, symptom checker, health calculators, weight tracker, and reminders. Developers can utilize this documentation as a guide to implement these features effectively and ensure a seamless user experience. Remember to prioritize user privacy and security throughout the development process.