BlanceBuddy - LifestyleApp

Camelia Morari

July 2024

Contents

1 Introduction

1.1 Application Description

BalanceBuddy is an application designed to help users improve their lifestyle by monitoring their daily habits. The app tracks various aspects of a person's daily routine such as steps taken, water intake, coffee consumption and food habits. The user will be able to input a goal in the beginning of a day regarding any desired field. At the end of each day, a report is sent to the user, summarizing their activities, progress towards the goal and some some recommendations for improvement. BalanceBuddy aims to encourage users change their lifestyle into a healthier and more balanced one.

1.2 Used Technologies

The technologies used for this application are:

- Frontend: React
- Backend: JavaSpring
- Database: MySQL
- API Testing: Postman
- Authentication: JWT (JSON Web Tokens)
- Version Control: Git

I selected these technologies because they make up a robust and efficient technology stack that ensures scalability, flexibility and security for the application. React is chosen for the frontend due to its component-based architecture, which facilitates development. JavaSpring is employed for the backend because of its powerful features for building scalable and reliable applications. MySQL is used as the database for its performance and ease of integration with JavaSpring. Postman is an excellent tool for API testing, offering a user-friendly interface and comprehensive testing capabilities. JWT is utilized for providing a secure and stateless method of verifying users. Finally, Git is selected for version control and dynamic development due to its powerful branching and merging capabilities.

1.3 Functionalities

The main functionalities of the application are:

- **Authentication:**
  This functionality allows users to create a new account, to log in or to log out from an already existing account. Users can sign up by providing their email and password. They can log in using their credentials and log out when finished. Sessions are managed using JWT tokens.
- **Habit Tracking:**
  This functionality enables users to track their daily habits. The app collects the data regarding steps, which are automatically recorded using a sensor, water and food intake, coffee consumption and others by manual entry.
- **Goal Setting:**
  Users can set and manage their personal goals for various habits. They can define targets for each habit and update them later on. A notification system will remind users to work towards their goals.
- **Daily Reports:**
  A detailed summary is generated at the end of each day, including the user activities throughout the day and their progress. A recommendation is offered based on the user's performance alongside an encouraging quote.
- **User History:**
  A dashboard that provides a comprehensive overview of the user's progress with visual representations such as graphs and charts. These represent the progress and the user is able to view their past activities and habits.

- **Notifications:**
  Notifications are sent from time to time to keep users engaged and to remind them to stay on track with their goals. An extra notification is sent at the end of the day to inform the user when their daily report is ready.

1.4 User Stories

1. USER STORY #1 - User Registration
   As a new user,
   I want to create an account,
   So that I can start using the application.

   Acceptance Criteria:
   AC#1 - Link to Register
   Given I am a new user,
   When I go to the Login page,
   Then I should see a hyperlink to Register.

   AC#2 - Register Form
   Given I am a new user,
   When I follow the hyperlink to Register,
   Then I should see a registration form with the following fields:

FieldName	FieldType	Mandatory/Optional Field
Firstname	Free text, [6, 256] characters	Mandatory
Lastname	Free text, [6, 256] characters	Mandatory
Email	Free text, max 256 chars, email format	Mandatory
Password	Free text, min 6 chars, password format	Mandatory
Confirm Password	Should be the same with ”Password” field	Mandatory

   AC#3 - New Account Creation
   Given I am a new user,
   When I fill out the registration form correctly and click Register,
   Then I should see a ”Registration Successful” message, and my account should be created.

   Possible errors:

   - If the required fields do not meet the validation criteria, an error message is displayed on the screen informing the user what should be changed.
   - If some missing fields are detected, another error message is displayed, informing the user.
   - If the passwords do not match, a new error message is shown to the user.

2. USER STORY #2 - User Login/Logout
   As a registered user,
   I want to log in and out from the application,
   So that my data is protected.

   Acceptance Criteria:
   AC#1 - Login Form
   Given I am a registered user,
   When I go to the Login page,
   Then I should see a login form with fields for Username and Password.

   AC#2 - Successful Login
   Given I am a registered user,
   When I enter valid credentials and click Login,
   Then I should be redirected to the Main page of the application.

   AC#3 - Logout
   Given I am a logged-in user,
   When I click on the Logout button,
   Then I should be logged out and redirected to the Login page.

3. USER STORY #3 - Daily Steps Tracking
   As a user,
   I want to track my daily steps,
   So that I can monitor my physical activity.

   Possible errors:

   - If some missing fields are detected, another error message is displayed, informing the user.
   - If the credentials are invalid, an error message appears informing the user about the mistake.

   Acceptance Criteria:

   AC#1 - Steps Input
   Given I am a logged-in user,
   When I go to the Tracking steps page,
   Then I should see a field displaying the total number of steps I made today.

4. USER STORY #4 - Water Intake Logging
   As a user,
   I want to log my water intake,
   So that I stay hydrated.

   Acceptance Criteria:

   AC#1 - Water Intake Input
   Given I am a logged-in user,
   When I go to the Water Intake page,
   Then I should see an input field to enter my daily water intake in liters or glasses oro other measuring technique.

   AC#2 - Water Intake Logging
   Given I am a logged-in user,
   When I enter my water intake and click Save,
   Then my water intake should be logged and displayed in my activity history.

   Possible errors:

   - Input is a non-numeric value, so an error is thrown.

5. USER STORY #5 - Goal Setting
   As a user,
   I want to set daily, weekly or monthly, goals
   So that I can improve my lifestyle.

   Acceptance Criteria:
   AC#1 - Goal Setting Form
   Given I am a logged-in user,
   When I go to the Goal Setting page,
   Then I should see a form to set goals for different parameters.

   AC#2 - Save Goals
   Given I am a logged-in user,
   When I set my goals and click Save,
   Then my goals should be saved and displayed on my Dashboard.

   AC#3 - Edit Goals
   Given I am a logged-in user,
   When I edit my goals and click Save,
   Then my goals should be saved and displayed on my Dashboard.

   Possible errors:

    - Input is a non-numeric value, so an error is thrown.

6. USER STORY #6 - Daily Report
   As a user,
   I want to receive a daily report of my activity,
   So that I can analyze my progress.

   Acceptance Criteria:

   AC#1 - Daily Report View
   Given I am a logged-in user,
   When I go to the Daily Report page,
   Then I should see a summary of my activities throughout the day.

7. USER STORY #7 - User History
   As a user,
   I want to see a overview of my progress,
   So that I can easily track my habits.

   Acceptance Criteria:

   AC#1 - History View
   Given I am a logged-in user,
   When I go to the History page,
   Then I should see a detailed history of my activity including graphical representations like graphs and charts.

8. USER STORY #8 - Notifications
   As a user,
   I want to receive timely notifications,
   So that I do not forget to input parameters or see the daily report.

   Acceptance Criteria:

   AC#1 - Notification Settings
   Given I am a logged-in user,
   When I go to the Notification Settings page,
   Then I should see options to set reminders for receiving notifications.

   AC#2 - Notification Alerts
   Given I am a logged-in user,
   When it is time to log steps or water intake,
   Then I should receive a notification reminding me to do so.

1.5 Interfaces, Module Communication and Protocol

- **Login/Register:** Here the users can sign up or log in filling up the required fields.
- **User History:** Displays the user's progress with charts and graphs
- **Goal Setting:** Users can set and edit their daily, weekly or monthly goals.
- **Daily Report Page:** Shows a summary of the user's activities during the day and progress towards their goals.
- **Tracking Page:** Allows user to log in their tracking parameters.

The user can freely navigate between the pages as long as it is logged in. The Frontend-Backend communication is handled via RESTful APIs over HTTP. The frontend sends requests to the backend, which processes the data and returns responses. For the authentication, JWT tokens are used to manage

user sessions and secure API endpoints.
The protocol used for this application is HTTP for the connection between server and client and for sending requests, on port 80.

1.6 Message Flow

1. User log in:
   The user enters their username and password, the server sends a request of type POST /api/auth/login followed by a body written in a JSON format, with the username and password given by the user, returning a JWT token:

   ```
   {
     "username": "user@example.com",
     "password": "password123"
   }
   ```

2. User registration:
   The user enters their information, including, firstname, lastname, username (email), password and confirmed password, the server sends a request of type POST /api/auth/register followed by a body written in a JSON format, with the credentials given by the user, returning the id of the newly created account:

   ```
   {
     "firstname": "John",
     "lastname": "Doe",
     "username": "newuser@example.com",
     "password": "password123",
     "confirmPassword": "password123"
   }
   ```

3. User enters homepage:
   When the user wants to return to the homepage of the app, a request of type GET /homepage is sent to the server, returning the ”homepage.html” to the user, with the interface of the main page of the application.

4. User views daily report:
   If a user selects to view their daily report, a request of type GET /dailyReport/{user_id} will return the daily report page ”dailyReport.html”.

5. Add a habit:
   When a user decides to input information regarding a habit, a request of type POST /habit/add is made, returning a value confirming that the habit addition was made succesffully and updates the user's progress.

   ```
   {
     "userId": "12345",
     "habitType": "waterIntake",
     "value": 250,
     "date": "07-08-2024"
   }
   ```

6. Log Sleep:
   A request of type POST /sleep/log is made in order to input the parameters regarding the sleeping schedule of the user. As a response, the progress of the user updates and a confirmation message is sent.

   ```
   {
     "userId": "12345",
     "habitType": "waterIntake",
     "value": 250,
     "date": "07-08-2024"
   }
   ```

7. Tracking Exercise:
   When the user wants to log details about their physical exercise, the type of the request is going to be POST /exercise/log, where the user can choose the type of exercise he did during the day, and for how much time. In response, a confirmation message is sent and the user's progress updates.

   ```
   {
     "userId": "67890",
     "exerciseType": "running",
     "duration": 30,
     "date": "07-08-2024"
   }
   ```

8. Set a goal:
   If a user wants to set a daily goal for a specific habit, a request of type POST /goal/set is made, returning as response a value confirming that the goal setting was made.

   ```
   {
     "userId": "12345",
     "habitType": "steps",
     "targetValue": 10000,
     "period": "daily"
   }
   ```

   In order to edit a goal, a PUT /goal/edit request is made, and for deletion of a goal, a DELETE /goal/delete request is made.

9. Generate a Daily Report:
   The server makes a request of type POST /report/generate, receiving in return the generated daily report:

   ```
   {
     "userId": "12345",
     "date": "07-08-2024"
   }
   ```

10. Send Reminder Notification:
    A request of type POST /notification/reminder is sent containing the user id and the type of announce the notification is for. In response, a value is being sent in order to verify if the notification has been sent correctly or not.

    ```
    {
      "userId": "12345",
      "type": "waterIntakeReminder"
    }
    ```

1.7 Scalability Considerations

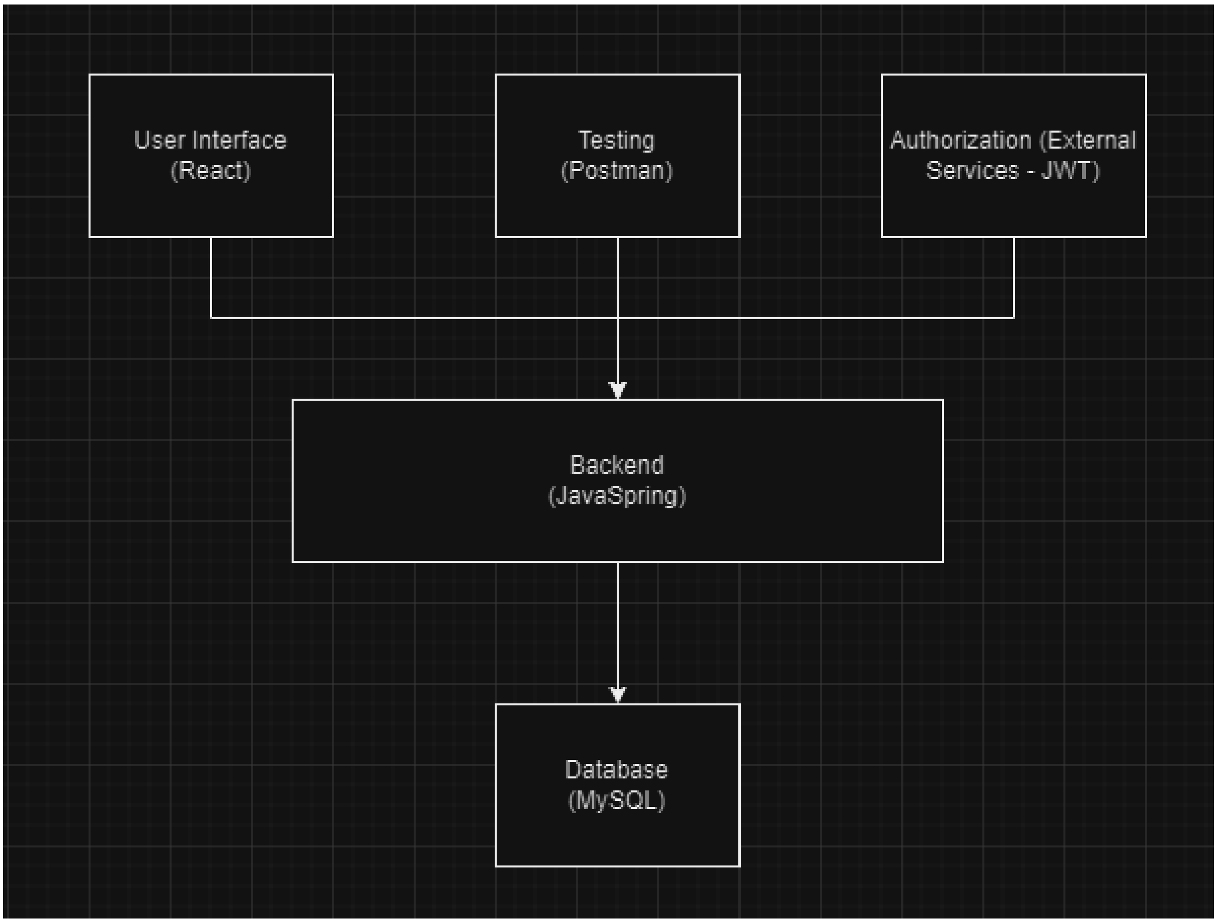
Server Specifications:

- CPU: 6 cores
- Logical processors: 12
- RAM: 32GB
- Storage: 250GB SSD

Load Capacity:

- Concurrent Users: up to 5000
- Requests per Second: up to 500

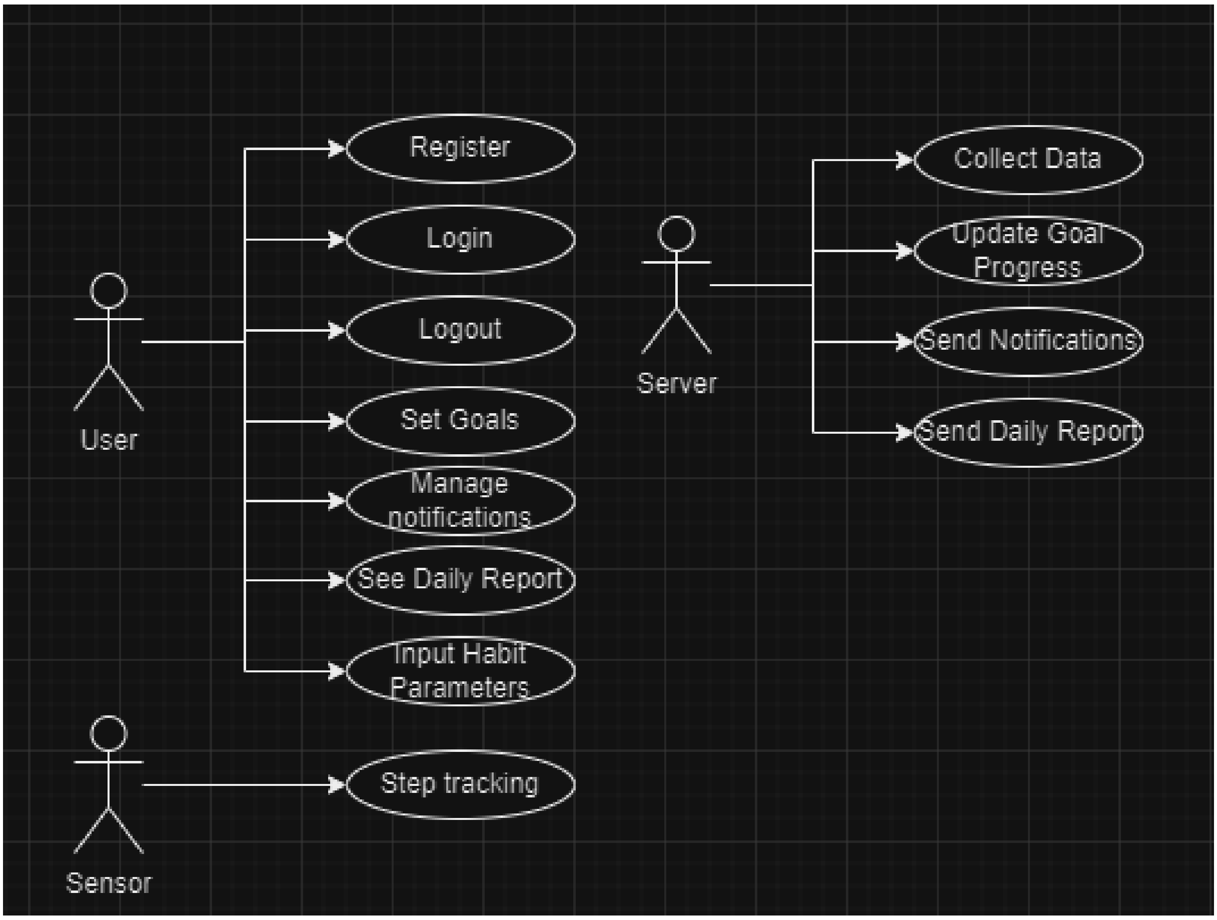
1.8 Application Diagram



The User Interface (React) provides an interface for BalanceBuddy. The user interacts with the application through the UIT, which sends request to the backend API.
External Services are used for functionalities like sending notifications or handling authentication.
Postman is used for testing the API endpoints exposed by the backend. It simulates requests and checks responses to ensure the API is functioning correctly.
The Backend (JavaSpring) handles all the business logic, exposing API endpoints that the frontend can call to perform operations such as logging habits, setting goals, generating reports, and so on. It interacts with the MySQL database to store and to retrieve data.
The Database (MySQL) stores all the persistent data for the application, including user information, habits, goals, reports.

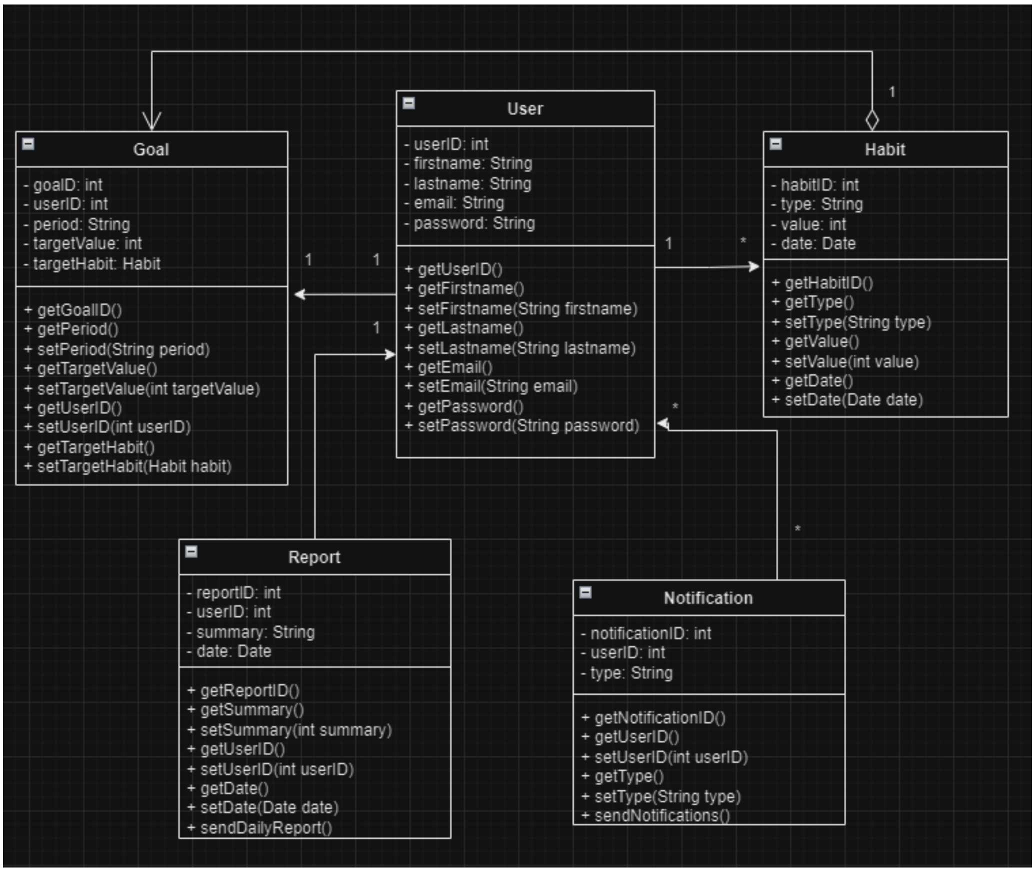
1.9 Architecture Elements

1.9.1 Use Case Diagram



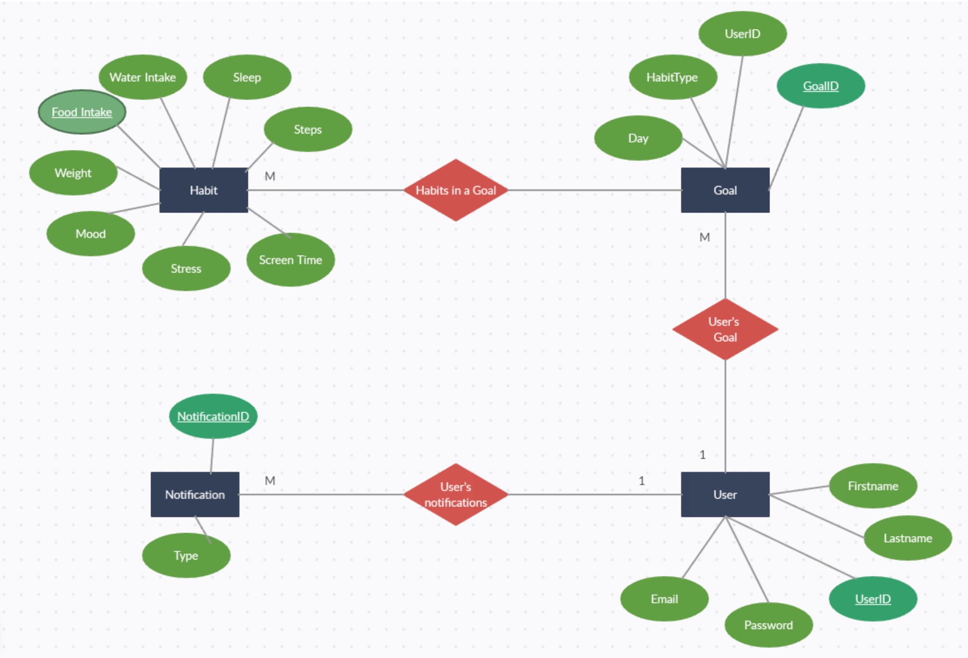
The use case diagram illustrates the actions that each actor of the application can do. The present actors are the User, Sensor and the Server. The User represents the end-users of the BalanceBuddy application who interact with the system to manage their lifestyle habits. The Sensor represents the hardware or software sensors that track user steps. Finally, the Server represents the backend server that processes data, sends notifications and manages user goals and reports. A user is able to register, login, logout, set goals, see the daily report and input habit parameters. The sensor sends the detected number of steps to the server, while the server collects all the data, updates the progress of the user's goal, sends notifications and daily reports.

1.9.2 Class Diagram



This is an orientative class diagram for the application. It provides a clear overview of the structure and interactions between different entities in the application highlighting how user data is managed, how habits and goals are tracked and how reports and notifications are generated and sent.

1.9.3 Entity Relationship Diagram



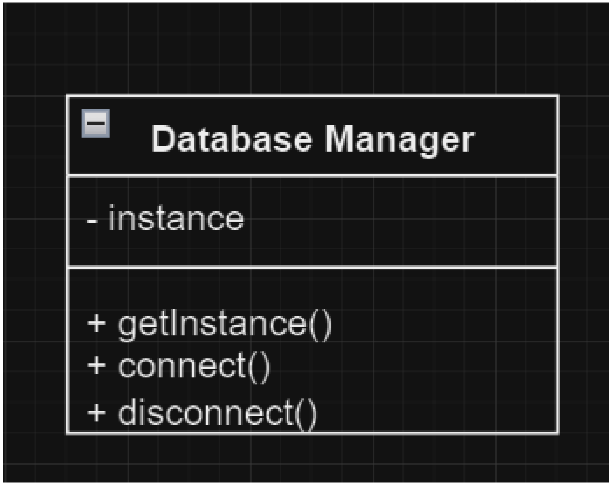
1.9.4 Lifecycle of objects

1. User object
   - **Creation:** When a user registers, a new User object is created.
   - **Usage:** The User objects is used for login, setting goals, inputting habits, viewing reports and managing notifications.
   - **Deletion:** When a user account is deleted, the User objects is removed form the database.
2. Habit object
   - **Creation:** When a user logs a new habit, a Habit object is created.
   - **Usage:** The Habit object is used to store and track habit data such as water intake, food consumption and so on.
   - **Deletion:** When a habit entry is deleted by the user, the Habit object is removed from the database.
3. Goal Object
   - **Creation:** When a user sets a new goal, a Goal object is created.
   - **Usage:** The Goal object is used to track user goals and progress towards achieving them.
   - **Deletion:** When a goal is deleted or achieved by the user, the Goal object is removed from the database.
4. Report Object
   - **Creation:** When a daily report is generated, a Report object is created.
   - **Usage:** The Report object is used to summarize the user's daily activities and progress.
   - **Deletion:** Reports are stored for the user dashboard/history system, but can be deleted based on user preferences.
5. Notification Object
   - **Creation:** When a notification is triggered, a Notification object is created.
   - **Usage:** The Notification object is used to manage and send reminders and alerts to the user.
   - **Deletion:** Notifications are removed after they are sent.

1.10 Design Patterns

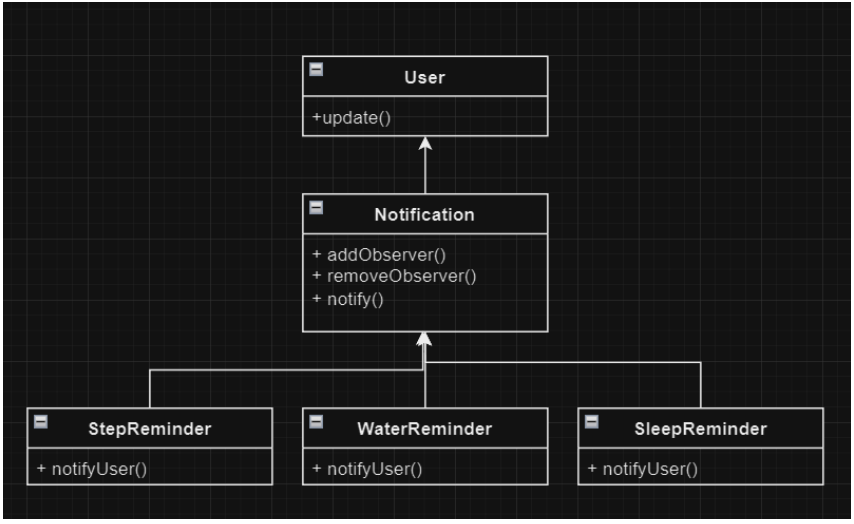
1.10.1 Singleton Pattern

The Singleton pattern is used for managing the database connection and to ensure there is a single instance of the connection used across the application.



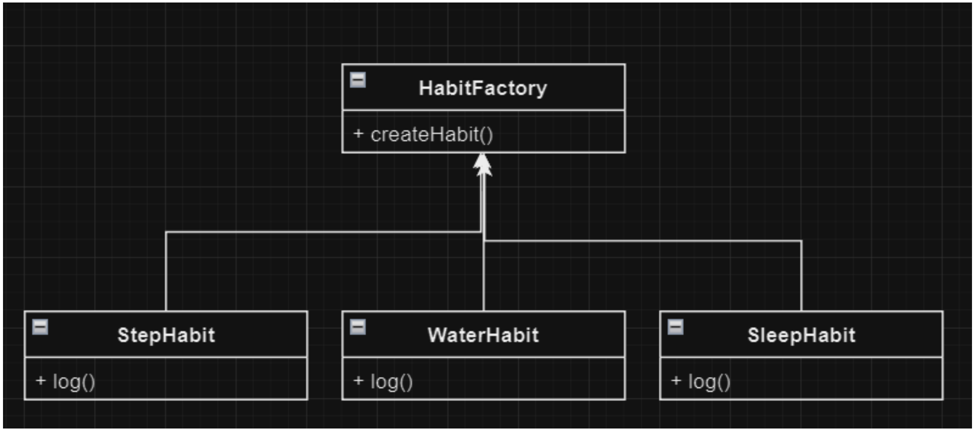
1.10.2 Observer Pattern

The Observer pattern is used for implementing the notification system where users can subscribe to certain habit reminders.



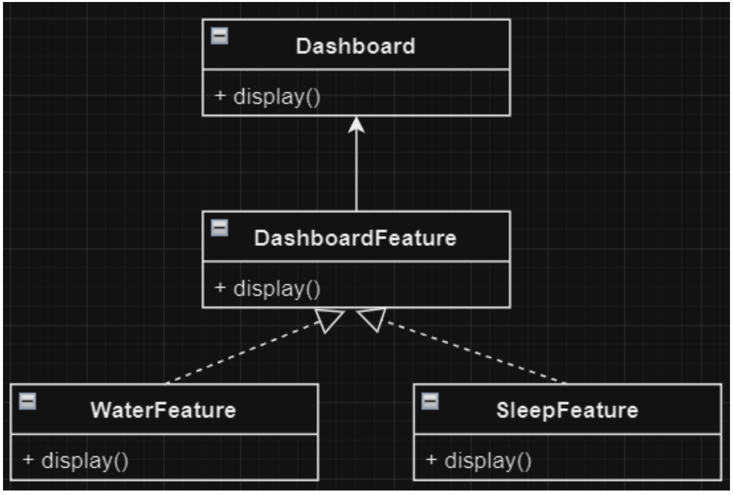
1.10.3 Factory Pattern

The Factory pattern is used for creating different types of habit log entries, for example steps, water, coffee, food, stress level, sleep hours, etc.



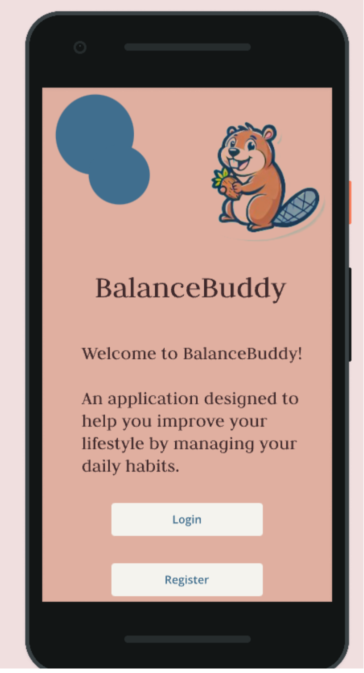
1.10.4 Decorator Pattern

The Decorator pattern is used for adding additional features to the user dashboard dynamically.



1.11 Prototype

1.11.1 Landing Page



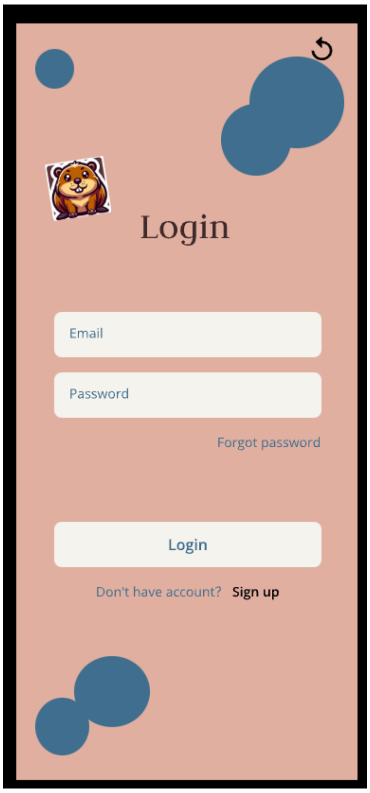
Welcome to BalanceBuddy!

An application designed to help you improve your lifestyle by managing your daily habits.

Landing Page Description
The landing page is the first interface that users encounter when they access the BalanceBuddy application. It provides a welcoming overview. The primary elements on this page include:

- A brief introduction to the app and its benefits.
- Button to log in.
- Button to register
- Visual elements to engage users and make navigation intuitive.

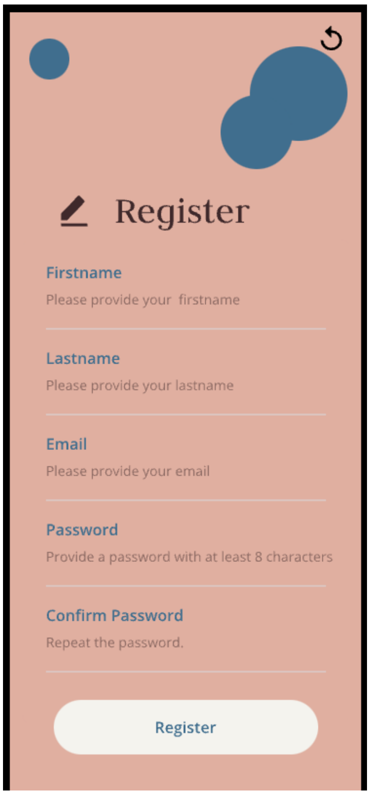
1.11.2 Login Page



Login Page Description
The login page allows users to access their personal BalanceBuddy account. It includes:

- Fields for entering email and password.
- A link to reset the password if the user has forgotten it.
- A button to submit the login form.
- A link to the registration page for new users.
- A button for going back to the landing page.

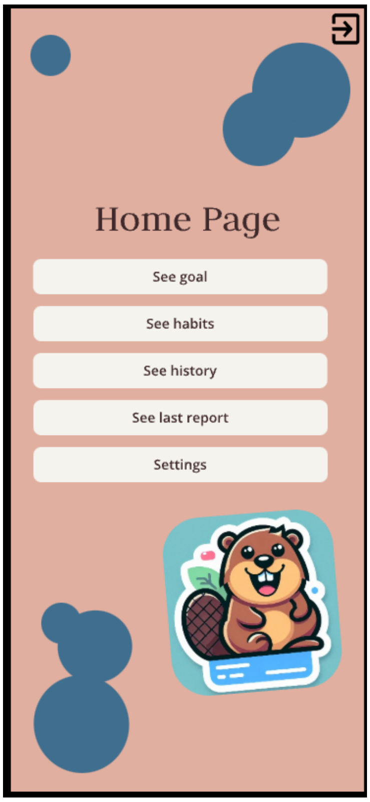
1.11.3 Register Page


Register
Firstname
Please provide your firstname
Lastname
Please provide your lastname
Email
Please provide your email
Password
Provide a password with at least 8 characters
Confirm Password
Repeat the password.
Register

Register Page Description
The register page allows new users to create a BalanceBuddy account. It includes:

- Fields for entering personal information such as name, email, and password.
- A button to submit the registration form.
- A button for going back to the landing page.

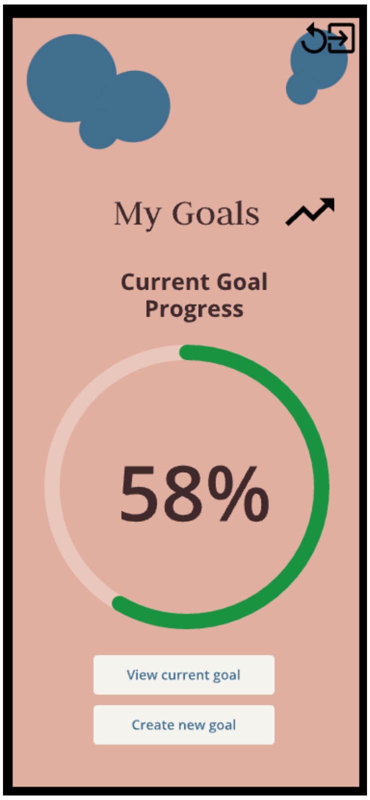
1.11.4 Home Page



Home Page Description
The home page provides users with an overview of the application pages. It includes:

- A button directing the users to the goal page.
- A button directing the users to the habits page.
- A button directing the users to the history page.
- A button directing the users to the settings page.
- A button directing the users to the report page.
- A button for logging out.

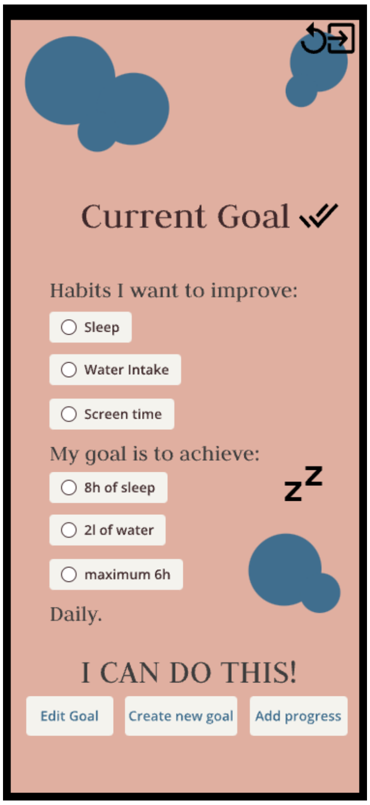
1.11.5 Goals Page



Goals Page Description
The goals page allows users to view and manage their lifestyle goals. It includes:

- The progress of the current goal.
- Option to view the current goal.
- A button to create new goals.
- Buttons for logging out and going back to the last accessed page.

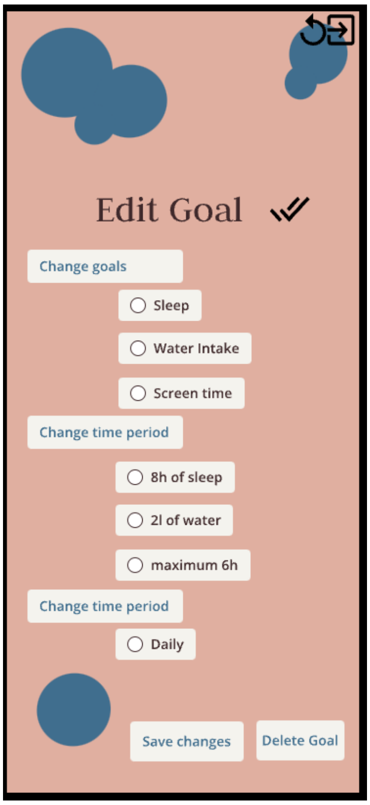
1.11.6 Current Goal Page



Current Goal Page Description
The current goal page provides details of a specific goal. It includes:

- The target habit and its parameters.
- Summary of the goal.
- Option to edit the goal.
- A button to create new goals.
- A button to add the progress to the goal.
- Buttons for logging out and going back to the last accessed page.

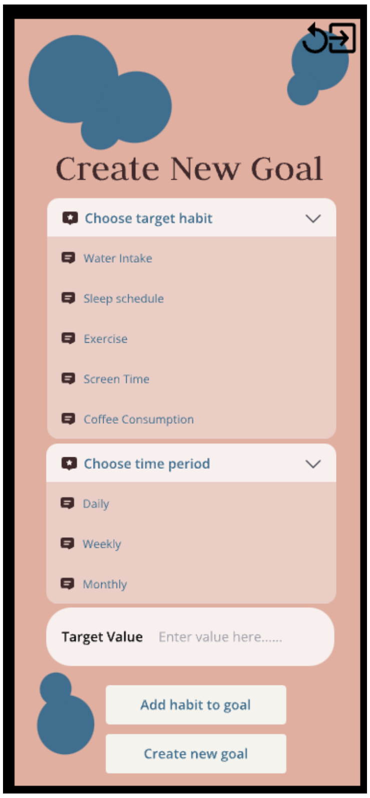
1.11.7 Edit Goal Page



Edit Goal Page Description
The edit goal page allows users to modify an existing goal. It includes:

- Fields to update the goal parameters such as target habit, value, and period.
- A button to save the changes.
- A button to delete the goal.
- Buttons for logging out and going back to the last accessed page.

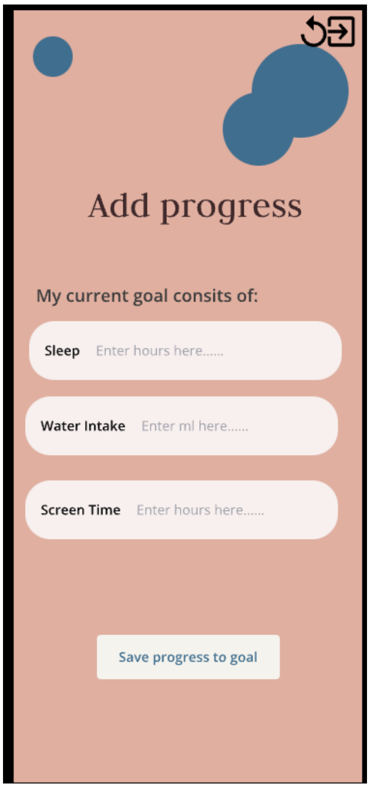
1.11.8 Create New Goal Page



Create New Goal Page Description
The create new goal page allows users to set up a new goal. It includes:

- Fields to specify the target habit, value, and period.
- A button to create the new goal.
- A button to add the habit to the goal that is in creation.
- Buttons for logging out and going back to the last accessed page.

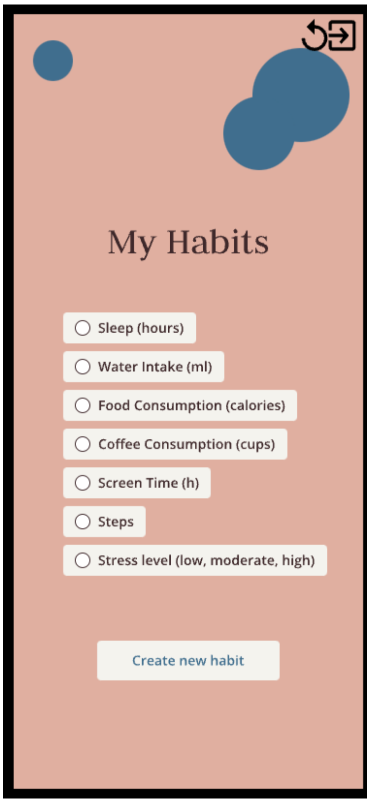
1.11.9 Add Progress Page



Add Prorgress Page Description
The add progress page allows users to input the habit parameters for their current goal. It includes:

- Fields to enter the progress made towards the current goal.
- A button to save the progress.
- Buttons for logging out and going back to the last accessed page.

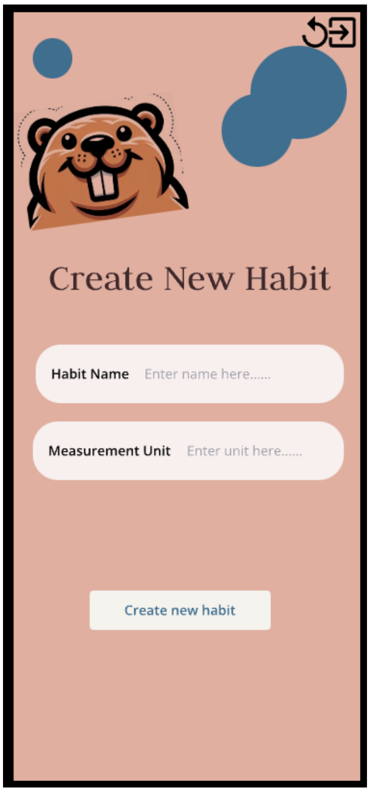
1.11.10 Habit Page


My Habits
○ Sleep (hours)
○ Water Intake (ml)
○ Food Consumption (calories)
○ Coffee Consumption (cups)
○ Screen Time (h)
○ Steps
○ Stress level (low, moderate, high)
Create new habit

Habits Page Description
The habit page allows users to track and manage their daily habits. It includes:

- A list of habits that can be tracked by the user.
- Options to create a new habit.
- Buttons for logging out and going back to the last accessed page.

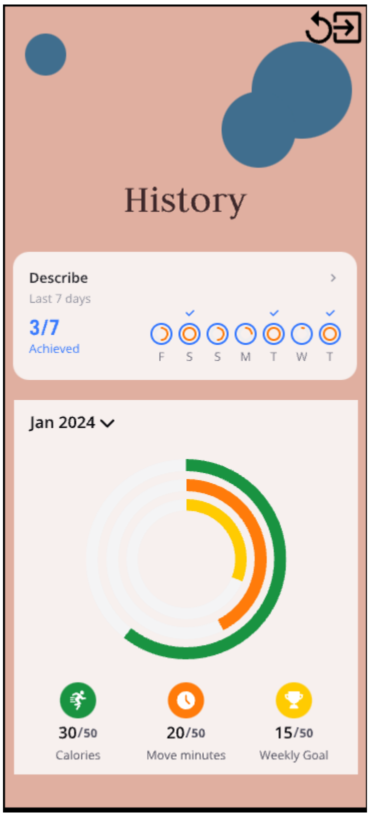
1.11.11 New Habit Page



Creat New Habit Page Description
The new habit page allows users to add a new habit to be tracked. It includes:

- Fields to specify the habit name and the measurement unit of the habit.
- A button to save the new habit.
- Buttons for logging out and going back to the last accessed page.

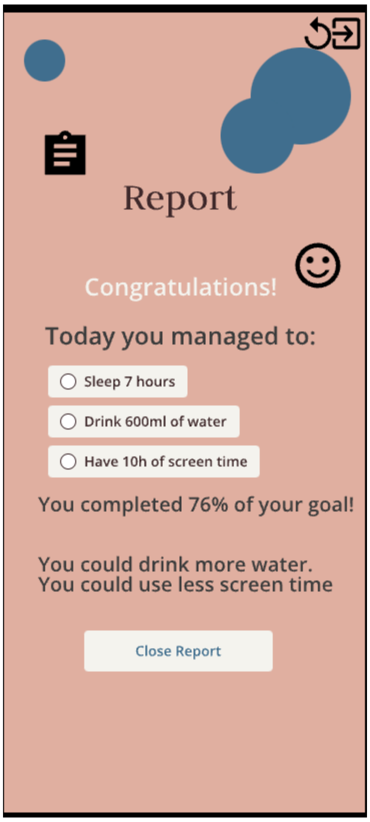
1.11.12 History Page



History Page Description
The history page provides users with a record of their tracked habits and goals. It includes:

- A chronological list of past habits and goals.
- Summary statistics and progress reports.
- Buttons for logging out and going back to the last accessed page.

1.11.13 Report Page



Report

Congratulations!

Today you managed to:

○ Sleep 7 hours

○ Drink 600ml of water

○ Have 10h of screen time

You completed 76% of your goal!

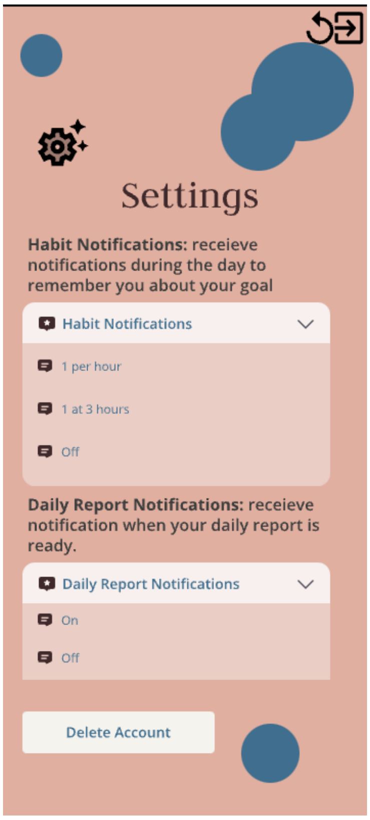
You could drink more water.
You could use less screen time

Close Report

Report Page Description
The report page provides users with detailed reports of their progress. It includes:

- Summaries of the user's habit tracking and goal progress.
- Button to close the report.
- Buttons for logging out and going back to the last accessed page.

1.11.14 Settings Page



Settings

Habit Notifications: receieve notifications during the day to remember you about your goal

Habit Notifications

1 per hour

1 at 3 hours

Off

Daily Report Notifications: receieve notification when your daily report is ready.

Daily Report Notifications

On

Off

Delete Account

Settings Page Description
The settings page allows users to customize their BalanceBuddy experience. It includes:

- Settings to configure notifications and reminders.
- Button for deletion of account.
- Buttons for logging out and going back to the last accessed page.