MAD PROJECT(MAP MY FITNESS):

PROJECT MEMBERS

20SW027
20SW071

Real world problem identification:

The " MapMyFitness " fitness tracking app is a mobile application designed to promote good health and well-being while advancing gender equality in the realm of fitness and health. The app achieves this by providing essential functionalities that empower users to take control of their fitness journeys while fostering inclusivity and accessibility. " MapMyFitness " aligns with the United Nations' Sustainable Development Goals (SDGs), particularly SDG 3 (Good Health and Well-being) and SDG 5 (Gender Equality).

PROPOSED SOLUTION

1. User Registration and Authentication:

- " MapMyFitness " enables users to create secure accounts, ensuring data privacy and protection. The implementation of password recovery and reset options enhances the user experience.

2. User Profile:

- Users can create and manage personalized profiles, including vital information such as name, age, weight, height, and fitness goals. The app promotes inclusivity and gender equality by catering to users of all genders.

3. Home/Dashboard:

- The app features a central hub, the dashboard, where users can access daily or weekly summaries of their fitness data. This empowers users to monitor their progress towards healthier lifestyles.

4. Workout Plans:

5. PLAN YOUR DAY

6.VIEW PROGRESS:

7. Accessibility:

DATA STORAGE:

Use SQFlite databse for plan your Day function beacuse
Using SQLite (often implemented in Flutter through the `sqflite` package) for a fitness app's "Plan Your Day" feature can be justified for several reasons:

1. Data Persistence: SQLite is a lightweight, embedded database that allows you to persistently store structured data on the user's device. In a fitness app where users need to save and access daily tasks, this is crucial for retaining user data even when the app is closed or the device is rebooted.

2. Structured Data: SQLite is a relational database management system, which means you can structure your data in tables and define relationships between them. This makes it suitable for organizing and managing daily tasks efficiently.

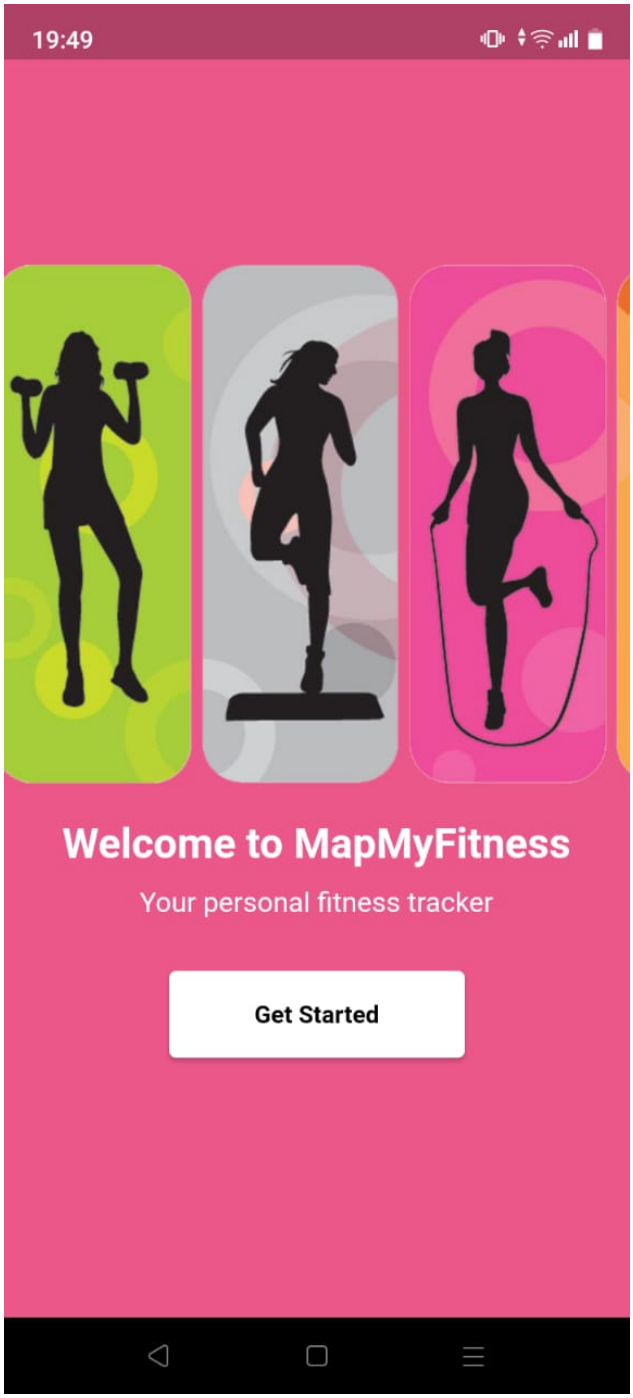
3. Offline Access: SQLite enables your app to function offline, ensuring that users can access, add, delete, or update their daily tasks even without an internet connection. This is essential for a fitness app, as users may not always have an internet connection while working out.

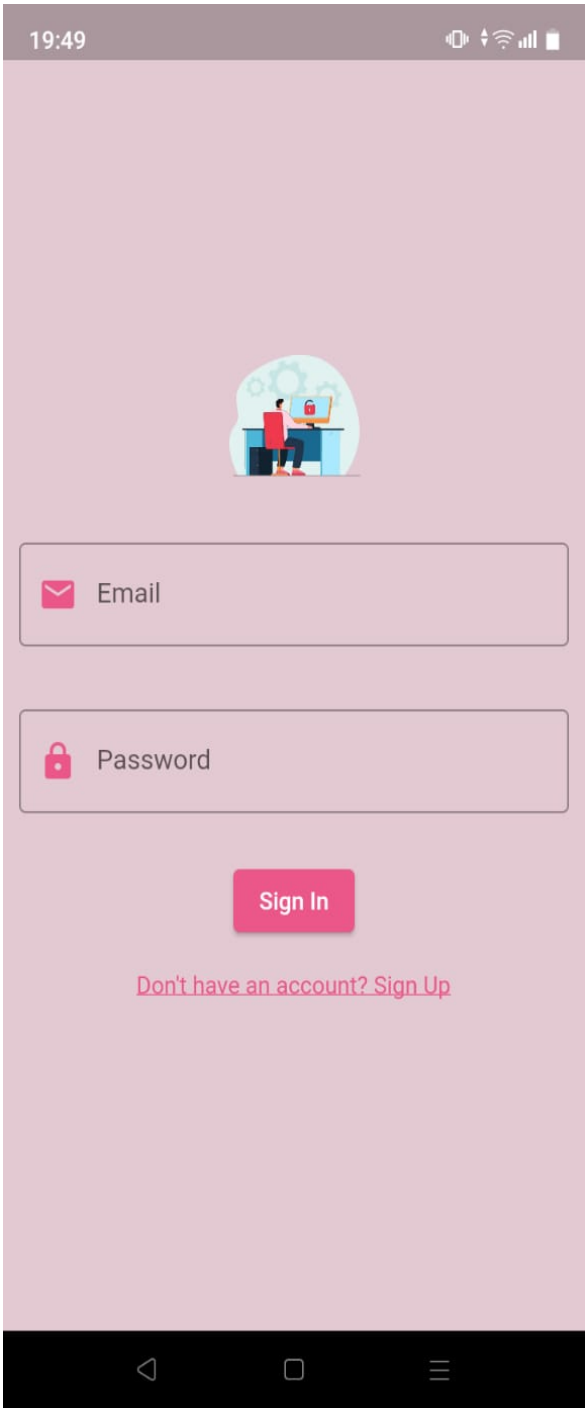
4. Efficient Data Queries: `sqflite` provides a SQL-like interface for querying and manipulating data. You can easily retrieve specific tasks, filter them, or perform complex queries to retrieve insights about a user's daily routine. This flexibility can be valuable in a fitness app.

5. Performance: SQLite is known for its performance and can handle a considerable amount of data efficiently. For a fitness app with potentially large amounts of data, this can ensure smooth and responsive user experiences.

Uses Firebase for authentication

RESPONSIVE UI:


19:49
Welcome to MapMyFitness
Your personal fitness tracker
Get Started


19:49
Email
Password
Sign In
Don't have an account? Sign Up

19:50

Fitness App Dashboard

WELCOME TO THE FITNESS APP DASHBOARD

START WORKOUT

VIEW PROGRESS

PLAN YOUR DAY

19:50

Fitness App Dashboard

Plan A

Description of Plan A

Plan B

Description of Plan B

Plan C

Description of Plan C

Plan D

Description of Plan D

Plan E

Description of Plan E

Plan F

Description of Plan F

Favorites

19:50

Fitness App Dashboard

Plan D

Description of Plan D

Plan E

Description of Plan E

Plan F

Description of Plan F

19:50

BMI Calculator

MALE

FEMALE

Height

0 cm

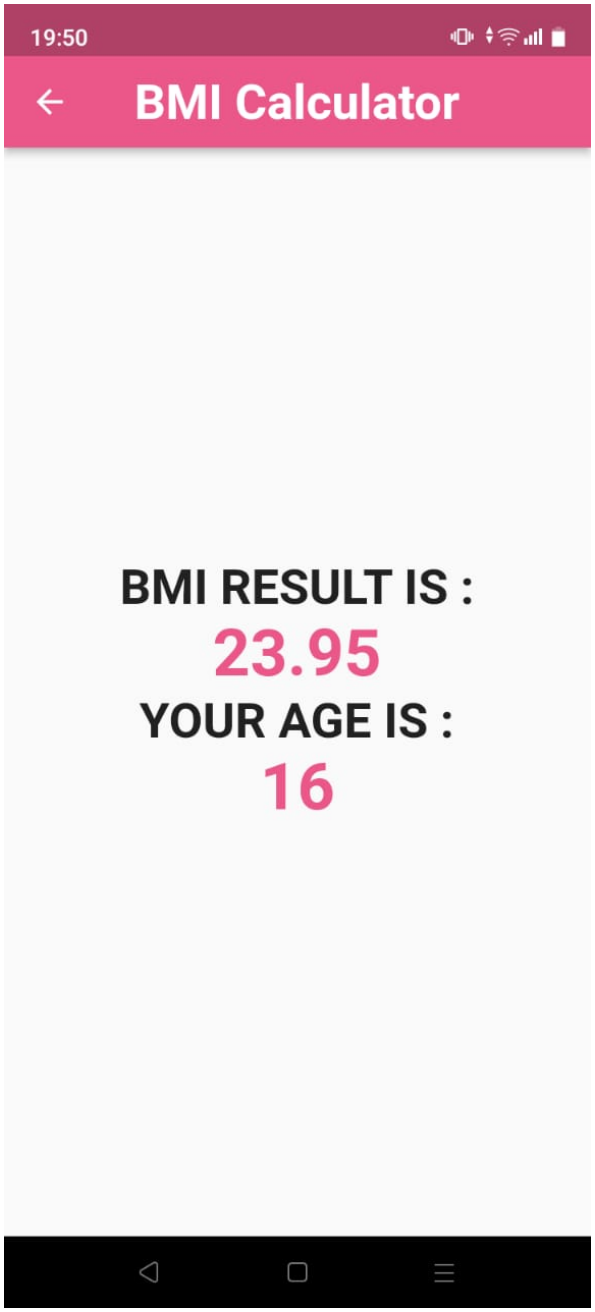
Weight

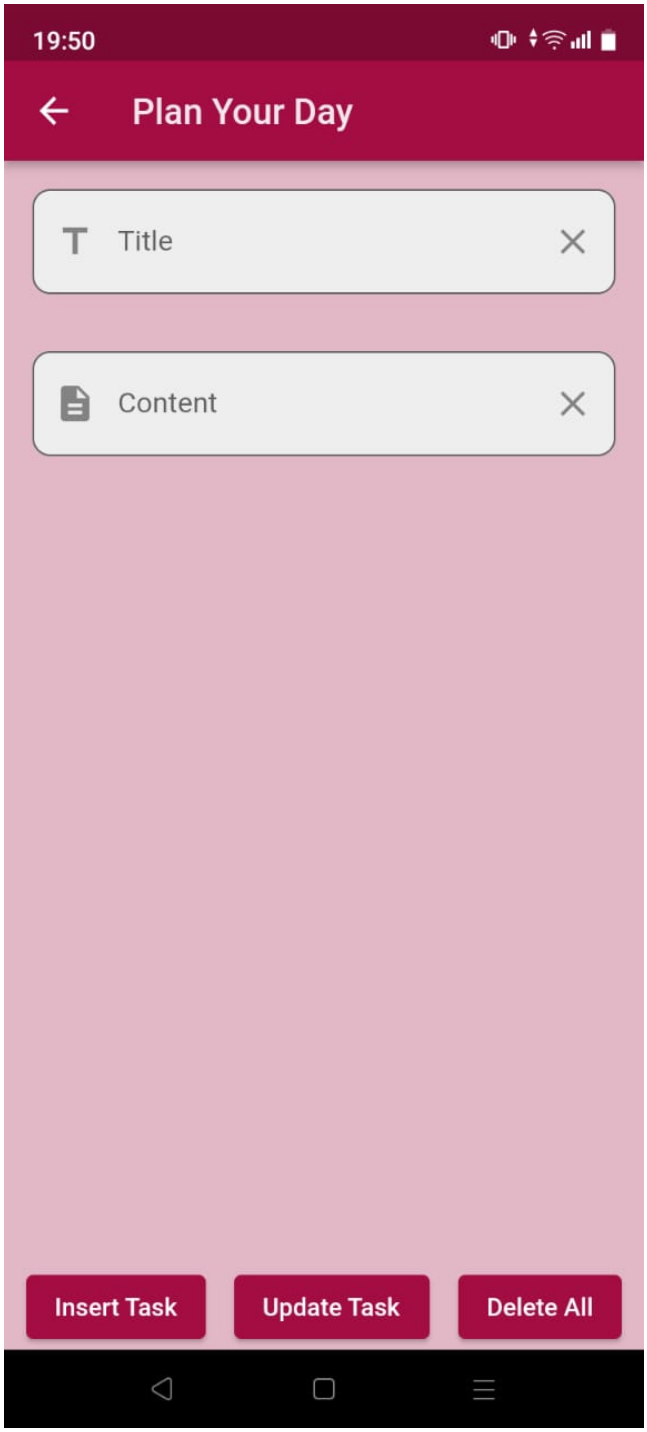
40

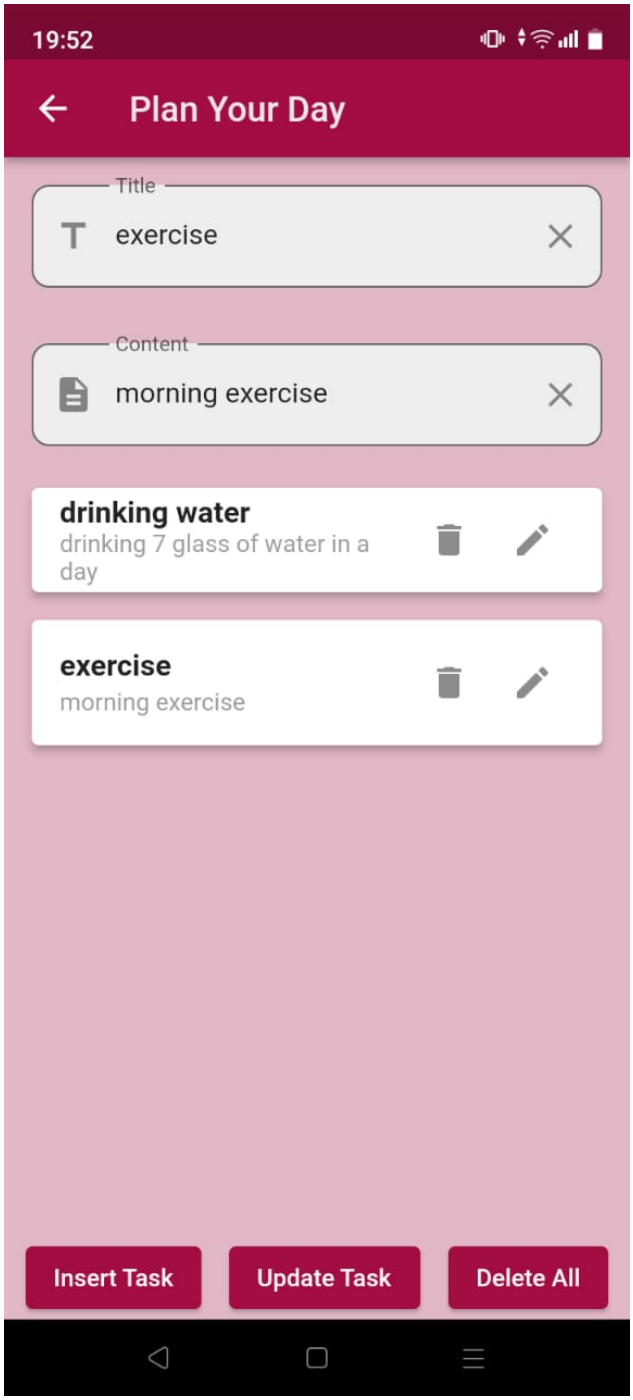
Age

10

CALCULATE


19:50
BMI Calculator
BMI RESULT IS :
23.95
YOUR AGE IS :
16


19:50
Plan Your Day
Title
Content
Insert Task
Update Task
Delete All



BUGS AND ERRORS

Error

1.In firebase authentication

2.In SQFlite database connectivity

3.Adding picture in provider code when user select exercise so picture picture is not showing on second screen
4.when user successfully login so redirect to dashboard
5.combining all screens