**Core Features:**

Personalized workout plans: This is the most important feature for a gym fitness app. Users expect personalized workout plans that suit their fitness goals and level of fitness.

Exercise library: A comprehensive exercise library is also an essential feature for a gym fitness app. Users want access to a variety of exercises with video demonstrations and detailed instructions.

Tracking and monitoring: Tracking and monitoring features are crucial for users to track their progress, monitor their workout routines, and receive feedback on their performance.

Nutrition tracking: Nutrition tracking is an important feature that allows users to monitor their calorie intake, track their macronutrient intake, and set nutrition goals.

**Extra Features:**

Social integration: Social integration features are useful for users to connect with friends, create workout challenges, and share their progress. This is a good feature to have but not essential for the app's core functionality.

In-app purchases: In-app purchases such as access to advanced workout plans and personalized coaching are not essential features, but they can generate revenue for the app.

Challenges and rewards: Challenges and rewards are also not essential features, but they can help users stay motivated and engaged with the app.

Personalized coaching: Personalized coaching from certified trainers is a great feature, but it is not essential for the app's core functionality.

**Day 1**

**Overview**

On day 1, I worked on creating the basic structure of the Fitness app. The main components created include:

* 5 fragments for different sections of the app
* Bottom Navigation Bar to switch between the different fragments

**Tasks Completed**

Created a new Android Studio project with an empty activity

Created 5 fragments for different sections of the app:

* Home Fragment
* Achivements Fragment
* Socials Fragment
* Exercise Fragment
* Settings Fragment

Implemented a Bottom Navigation Bar to switch between the different fragments

Added the necessary code to handle navigation between fragments on Bottom Navigation Bar item selection

Added necessary UI elements like text and image views, buttons, and layouts for each fragment

**Challenges Faced**

I faced a few challenges while working on day 1. These include:

* Understanding how to create and implement fragments in Android Studio
* Configuring and implementing the Bottom Navigation Bar to switch between fragments
* Figuring out the best layout and design for each fragment

**Day 2**

**Overview**

On day 2, I continued working on the Fitness app by focusing on the Home fragment. I implemented the necessary UI elements for the fragment, added functionalities such as displaying a list of daily workouts and setting and tracking fitness goals, improved the design and layout, and tested and debugged the fragment to ensure proper functionality.

**Tasks Completed**

The main tasks I completed on day 2 include:

• Implemented necessary UI elements for the Home fragment, such as text views, buttons, images, and a RecyclerView to display daily workouts.

• Added functionalities to the Home fragment, such as displaying a list of daily workouts with details such as exercise name, duration, and calories burned. The workout data was fetched from a mock API and displayed using a RecyclerView adapter.

• Added a Lottie animation to the Home fragment to make it more engaging and visually appealing.

• Improved the design and layout of the Home fragment by adjusting font sizes, colors, and spacing.

• Tested and debugged the Home fragment to ensure proper functionality and fixed any issues that arose during testing.

**Challenges Faced**

I faced a few challenges while working on day 2. These include:

• Figuring out how to fetch and display workout data from a mock API using a RecyclerView adapter.

• Configuring and implementing the Lottie animation in the Home fragment.

• Ensuring that the design and layout of the Home fragment was consistent with the overall design of the app.

**Day 3**

**Project Overview**

We worked on designing a database schema for a fitness tracking application. The schema consists of four tables: users, user\_tracking, workout\_plan, and exercise.

**Table Definitions**

* users: This table stores user account information, including user\_id, username, password, and email.
* user\_tracking: This table stores user fitness tracking information, including user\_id, date, weight, height, chest, biceps, waist, thigh, and hips.
* workout\_plan: This table stores user workout plan information, including workout\_name, training\_type, day, exercise1, exercise2, exercise3, exercise4, exercise5, and exercise6.
* exercise: This table stores exercise information, including exercise\_id, exercise\_name, bodypart, sets, reps, weight, and description.

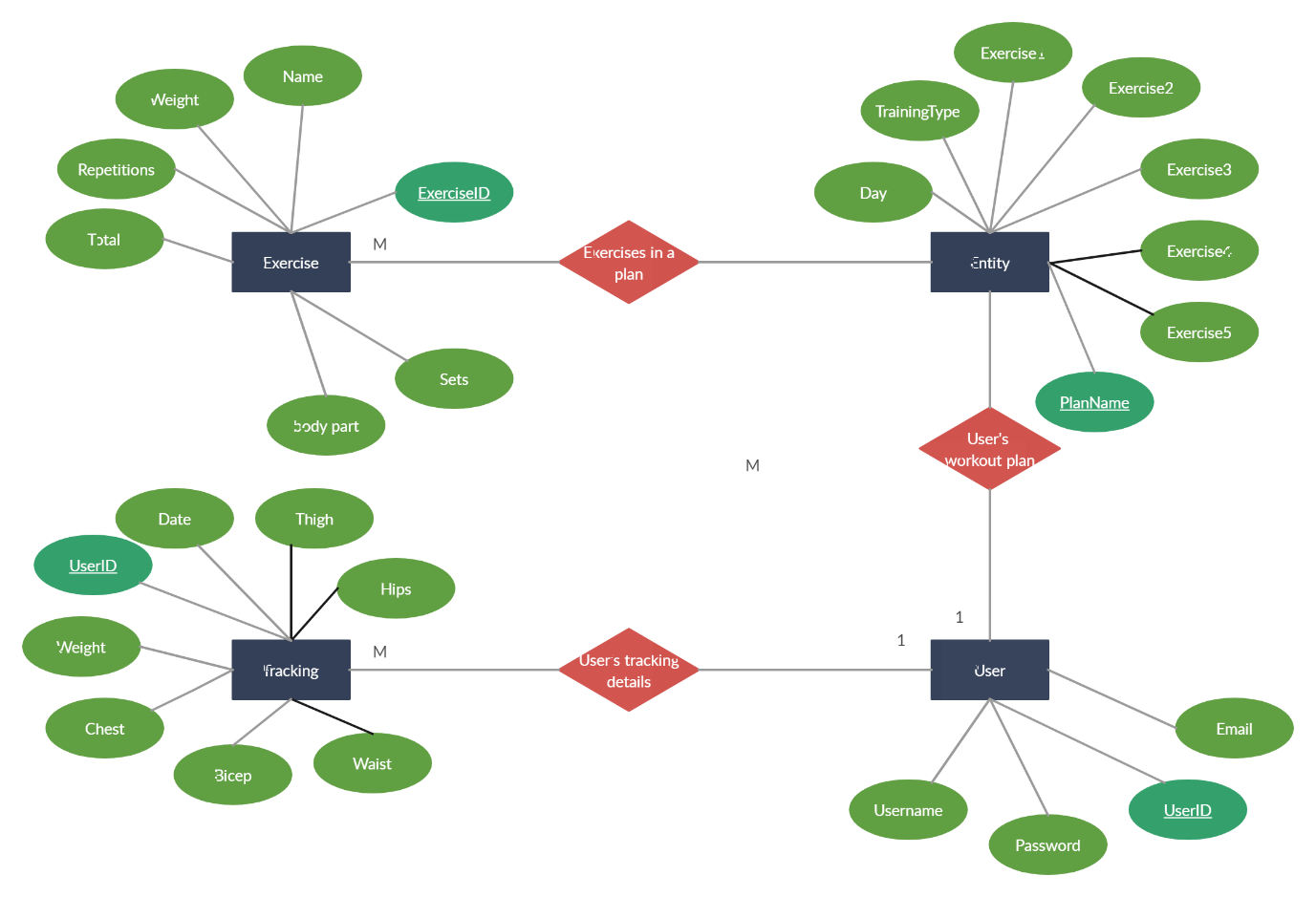
**SQL Queries**

* CREATE TABLE queries were used to create the four tables in the database schema.
* INSERT INTO queries were used to insert dummy data into the users, user\_tracking, workout\_plan, and exercise tables.

**Conclusion**

We successfully designed a database schema for a fitness tracking application and created dummy data for the tables using SQL queries. This database schema can be used as a foundation for developing a fitness tracking application.

**ER Diagram:-**

****

**Other Work**

To implement a login feature in the fitness app, allowing users to securely login to their accounts.

**Tasks Completed**

1. Created a MySQL database to store user login information.
2. Created a PHP API to handle login requests and validate user credentials.
3. Added the Volley library to the Android Studio project to make network requests.
4. Implemented a login activity in the Android app with two EditText views for the user to input their username and password.
5. Implemented a button to initiate the login process.
6. Added logic to check if the EditText fields were empty before making a network request.
7. Added a ProgressDialog to display a loading indicator while the network request is being made.
8. Implemented logic to save the user's login credentials using SharedPreferences if the "Remember Me" checkbox is checked.
9. Added a TextView with an onClickListener to navigate to the registration activity if the user does not have an account.
10. Created documentation summarizing the work completed.

**Challenges Faced**

Had to learn how to create a MySQL database and write SQL queries to interact with it.

Had to debug network issues when making requests to the API.

Had to learn how to use the ProgressDialog and how to dismiss it after the network request was completed.

**Results**

The login feature was successfully implemented and tested with sample user credentials. The ProgressDialog and "Remember Me" functionality provided a smoother user experience.