1.1 Identify the Core Purpose

Questions to Consider:

What is the main goal of your app?

* Keep track of activities through the weeks and using as a foundation for expansion on activity tracking and provide analytics on the activities

1.2 Determine Key Features

Questions to Consider:

What are the essential features that must be included in the first version of the app?

The core features will include activity tracking, a progress dashboard and goal setting

1.3 Understand Your Target Audience

Questions to Consider:

The target audience is fitness enthusiasts and beginners who want to track their physical activities and monitor their progress.

1.4 Define User Stories

Questions to Consider:

User stories:

As a user, I want to log my daily bike rides, gym activities.

As a user, I want to set and track my monthly bike skills goals.

As a user, I want to see a summary of my monthly activities in a graph by day.

1.5 Prioritize Features

Questions to Consider:

MVP must have activity tracking and progress report with metric display and basic goal setting. Nice to have would be further data analytics like exertion and calories and export features.

1.6 Define Technical Requirements

The app will be developed for Android using Java/Kotlin, with local storage using SQLite. Future versions may include cloud integration with AWS and support for wearable devices.

1.7 Sketch the User Interface

Tasks to Complete:

Create wireframes or mockups of the app’s main screens

Home screen with activity summary

Activity logging screen

Progress tracking screen

Settings and goal-setting screen

Tools:

Use design tools like Figma, Sketch, or even paper and pencil to draft initial designs.

Example:

Draft wireframes for the home screen showing daily activity summary, an activity logging screen with input fields for metrics, and a progress tracking screen with graphs and charts.

1.8 Plan the Development Timeline

Questions to Consider:

What is your timeline for development?

How long will each phase take (planning, development, testing, deployment)?

Are there any specific deadlines you need to meet?

Example:

Plan a development timeline that includes:

2 weeks for detailed planning and design

4 weeks for core feature development

2 weeks for testing and bug fixing

1 week for initial deployment and feedback collection