**Final Project Proposal**

**Name : Diogo Rangel Dos Santos**

**1. Project Title:** Fitness Progress Tracker

**2. Problem Statement**

Tracking fitness progress can be challenging without a structured system to log workouts, monitor improvements, and analyze trends over time. Many fitness enthusiasts rely on scattered notes, memory, or multiple apps, which may lack customization for their specific goals. This project aims to provide a simple, user-friendly program that helps individuals log their workouts, track progress, and visualize improvements over time.

**3. Proposed Solution**

The Fitness Progress Tracker will be a Python-based application that allows users to:

* Log different types of exercises (e.g., weightlifting, running, cycling, etc.).
* Store workout data such as repetitions, sets, weight used, duration, and distance.
* View progress over time with basic statistics and visualizations.
* Export data for further analysis.

The program will include multiple functions to handle data entry, data retrieval, calculations, and data visualization.

**4. Key Features**

* A command-line interface (CLI) for inputting workout details.
* Functions to calculate progress trends (e.g., average weight lifted over time, total distance run in a week).
* Data visualization using the matplotlib library to generate progress graphs.
* Data storage using a CSV file or a simple SQLite database.
* Unit tests for at least two core functions using pytest.

**5. Functions Overview**

* log\_workout(): Allows the user to input workout data.
* view\_progress(): Retrieves and displays historical workout data.
* calculate\_trends(): Computes key metrics such as average weight lifted, total time spent exercising, and improvements over time.
* generate\_report(): Creates a graphical report using matplotlib to visualize progress.
* export\_data(): Saves the workout data in a CSV file for external use.

**6. Tools & Libraries:**

* Python
* csv or sqlite3 for data storage
* matplotlib for data visualization
* pytest for testing core functions

**7. Time Estimate & Plan:**

**Week 1:**

Research and design program structure (2 hours)

Implement data logging and storage functions (4 hours)

Develop basic data retrieval and calculation functions (3 hours)

**Week 2:**

Implement data visualization and reporting (3 hours)

Write test cases and run pytest (2 hours)

Finalize documentation and refine code (2 hours)

**8. Expected Outcom**

By the end of the project, the Fitness Progress Tracker will be a functional Python application capable of logging workouts, tracking progress, and visualizing improvements. The structured approach will ensure code reusability, testability, and clarity.

**9. Potential Challenges:**

* Ensuring proper data validation for user inputs.
* Designing an intuitive CLI experience.
* Efficiently handling and visualizing workout data.

**10. Conclusion**

This project will provide a useful tool for fitness tracking while demonstrating strong Python programming skills, including data handling, function design, and unit testing. It will be a valuable personal project and can be expanded in the future with a graphical user interface or mobile integration.