Object Oriented and Functional Programming with Python (DLBDSOOFPP01)

EROS FORTEA

Phase 1, Conception:

Introduction

This document outlines the technical foundation for a habit tracking application. The concept covers the essential components, their interactions, and user flow within the application. The goal is to ensure a comprehensive understanding of the system before moving into implementation, avoiding potential oversights that could negatively impact the project.

System Overview

The habit tracking application allows users to create, manage, and track their habits. Users can define new habits, mark them as complete, and view progress over time. The system comprises several key components, each responsible for specific functionality.

Core Components

1. **Habit Class**: Represents a habit with attributes like name, description, creation date, frequency, and completion status.
2. **Database**: Stores user and habit data, ensuring persistence and retrieval.
3. **User interface:** The user will be able to interact with the app through the terminal.

Detailed Component Description

Database

SQLite for simplicity and ease of use.

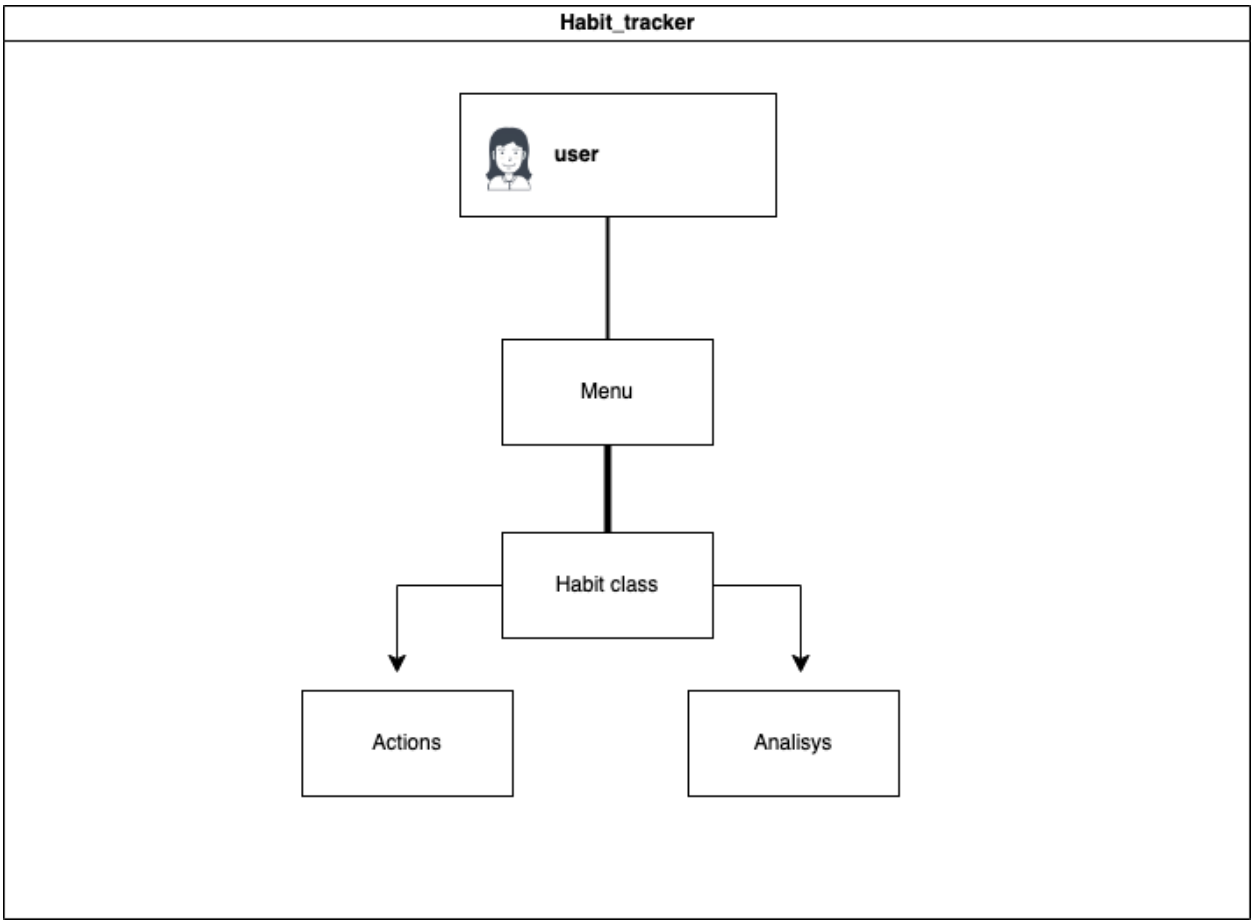
User Flow

4. **Dashboard**: A menu will be shown to the user and will be able to choose which functionality wants to use.
5. **Habit Management**: Users can create new habits, which are added to their list.
6. **Tracking Progress**: Users can mark habits as complete, updating their progress.

Tools and Technologies

- **Backend**: Python, SQLite.
- **Version Control**: Git, GitHub.

Diagram



Conclusion

This conceptual document provides a comprehensive overview of the habit tracking application's design. The proposed structure ensures that all aspects of user interaction and data management are covered, laying a solid foundation for the subsequent implementation phase.