Summary of the habit tracker

This abstract provides a brief overview of the content and concept of the project, highlighting the technical approach and lessons learned. It serves as a quick report to summarize the project and present key insights to the manager.

The project involved the development of a habit tracking application using object-oriented programming principles. Initially, I embarked on a learning journey to understand the fundamentals of object-oriented programming by watching videos and reading relevant resources. One particular video, which is referenced in the project's credits, served as a valuable starting point.

Building upon this foundation, I proceeded to develop the application, expanding its functionality significantly. The application enables users to define habits, track their progress, and analyze various aspects of their habits, such as streaks, ratings, and progress over time. Additionally, I incorporated a coach feature to provide users with reminders and support.

During the concept phase of the project, I recognized the importance of having a clear understanding of the application's structure and functionality. To aid in this process, I leveraged Lucidcharts, a free diagramming tool, to create a UML schema. This allowed me to visually represent the relationships between different components, classes, and their interactions within the application. The UML schema provided a comprehensive overview of the project, guiding the implementation process and ensuring a well-organized and modular design. The use of Lucidcharts and UML greatly contributed to the overall success of the project by providing a solid foundation for implementation and reducing the risk of misunderstandings or misalignments.

One aspect that I acknowledge as a mistake in the project's execution is that I postponed writing the tests until after completing the implementation. This approach resulted in additional effort and made it challenging to catch bugs or introduce changes without breaking existing functionality. In the future, I would prioritize writing tests alongside the development process to ensure more robust code and facilitate easier maintenance.

Despite this setback, the project has several noteworthy features that add value to the overall product. The analysis capabilities, such as calculating longest streaks, plotting habits by ratings, and identifying the easiest and hardest habits, provide users with valuable insights into their habits' progress and difficulty levels. The coach functionality, implemented as an email service, offers users reminders and support, enhancing their habit tracking experience.

In conclusion, the project successfully achieved its objective of developing a habit tracking application using object-oriented programming principles. Lessons learned include the importance

of writing tests throughout the development process and the significance of incorporating analysis and supportive features to enhance the user experience. Moving forward, I intend to apply these lessons in future projects to improve code quality, maintainability, and user satisfaction.

github link: https://github.com/Phips92/Habit_Tracker_IU