Project Concept and Content

The objective for this project was to develop a responsive web-based application that allows users to manage their personal growth through core functionalities:

- **User Management**: Create account, log in, and update personal account details.
- **Journaling**: visually track and visualize their thoughts with a journaling system based on a monthly calendar
- **Goal Tracking**: Define goals and monitor their progress in an easy and approachable way
- **Habit Tracking**: Add new habits with customizable monthly goals, and monitor progress through a user-friendly calendar interface.

Clients can manage these components by creating, editing and deleting entries through UI controls. Visual indicators provide real-time feedback on progress and completion status for goals and habits.

Implementation

The development of the application followed a structured approach suggested by the assignment document. The process included initial research of the best front-end and back-end design, and then an iterative loop comprehensive of code, implementation, testing, debugging, and ended with a final documentation.

Before initiating full-scale development, I watched tutorials on YouTube and experimented with locally saved goal systems to understand how similar systems work. Then. during implementation, I used tools such as GitHub Copilot, ChatGPT, and W3Schools for support and reference.

Technical Approach

Front-End Development

Tailwind CSS provided a utility-first framework that enabled responsive design directly with the classes, without writing manually media queries, simplifying the overall process.

Classic CSS was also used to design the UI to deepen my understanding of how stylesheets behave across browsers.

JavaScript handled core functionality, including generating dynamic monthly calendar views for journals and habits, fetching data from the back-end, and updating UI elements in real-time to reflect user interactions and data changes.

Back-End Development

Prior to development, I studied multiple tutorials and read books to learn the fundamentals of PHP and back-end logic.

Authentication was implemented using hashed passwords to ensure secure logic functionality. PHP connected the web-app with database using MySQL.

MySQL and XAMPP virtual server were chosen for familiarity because I already had basic knowledge of how it works, so it was the best option.

The database includes separate tables for users, habits, goals, and journal entries, with appropriate relationships linking data to individual users.

Testing and Debugging

This phase was one of the most time-consuming due to the complexity and interactivity of the features, especially because I wanted to them to work exactly how I planned in the initial documentation. Testing was done incrementally and feature-wise. Isolating and testing each module independently proved to be an effective debugging strategy.

1. Developed the habit tracking system, followed by debugging to ensure proper toggling and rendering. The habit-toggling mechanism was the most time-consuming and the greatest challenge, as it was a new concept for me.
2. Implemented the goal system, with testing to confirm accurate progress updates.
3. Created the journaling feature, then validated calendar integration and data storage.
4. Implemented and verified user authentication, registration, login, and session handling.

Documentation

To meet the project requirements, I updated all the previous documentation. This included:

- Updating the “brief written concept” with the actual technologies and features included and the high-level architecture.
- Test case additions in the PowerPoint for phase 2.

Lessons Learned

This project marked my first experience building a complete web application from scratch. Key takeaways include:

- Early Planning is Crucial: Initially, I underestimated the complexity of integrating front-end and back-end components. Mapping out the structure earlier would have saved significant time and confusion.
- Responsiveness Matters: I delayed the responsive design until late in the process, which made implementation more difficult and required layout restructuring. In future projects, designing mobile-first will be prioritized from the start.
- User Experience Awareness: I initially overlooked the importance of real-time feedback in user interactions. I realized how I, in the first case, expect immediate visual feedback when interacting with something on a website. Therefore, eventually, I prioritized re-rendering components (e.g., habit toggles, progress updates) to enhance responsiveness. As a result, extra effort was made to ensure fast and fluid UI reactions.

Reflections and Future Improvements

I am proud of my final product, which delivered a clean, intuitive interface with all core functionalities working as intended. However, there is room for growth. Future improvements may include:

- Applying more structured error-handling strategies learned during this project.
- Start UI responsiveness and backend integration earlier in the development cycle.

This project laid a strong foundation in full-stack web development, and the lessons learned will be used in my future work and projects.