Brionna Morris and Angel Jackson

SDEV 265

Shelvd – User Database for Book Tracking

**SOFTWARE INTRODUCTION**

The purpose of this project is to create a library system of books, called SHELVD, for users to keep track of the content they read. The web application will allow users to add the books they read and provide a review as well as a star rating on a one to five star system. The application is like what Letterboxd is for movies, but for books. Users’ input will be categorized in a way that allows them to see what they have read over the year and review their history from past years.

**SYSTEM REQUIREMENTS**

· Python

· Organization of genres

· Debugging

· Section for reviews

· Section for ratings

· Data library for books

· Functional system using Python

· Graphical Design

· Visual Studio Code IDE and Sublime Text

**SYSTEM ARCHITECTURE**

The system will be built in an iterative process where developers will focus on one section or module of the project at a time and test as we go along. The system will rely on Python for backend development, while using HTML & CSS for frontend and design. Changes will be made as testing is implemented and will use Visual Studio Code and Sublime Text for coding and organization of files.

**SYSTEM EVOLUTION**

As the platform grows, there will be a continuous need to adapt and evolve. Feedback from users will be instrumental in guiding the evolution of SHELVD. Future iterations might include features like book recommendations, social sharing, and integration with other platforms.

**SYSTEMS DEVELOPMENT LIFECYCLE**

Planning: Define the scope of the project and gather detailed requirements.

Analysis: Understand user needs and refine project objectives.

Design: Create a blueprint of the software architecture and design the user interface.

Implementation: Begin coding, adhering to the design specifications.

Testing: Rigorously test each module for bugs and ensure it meets user requirements.

Deployment: Launch the application for users.

Maintenance: Regularly update the application based on user feedback and emerging technologies.

**APPENDICES**

**Software Components:**

Tools and platforms used in the development process.

Trello: Web-based, kanban-style list-making application used for project planning and documentation.

Slack: Cloud-based instant messaging service for team communication and collaboration.

GitHub: Platform for version control, code storage, and collaborative coding.

VS Code: Primary code editor for development, integrated with GitHub for streamlined coding and version control.

Salesforce: (Hypothetical) Customer relationship management solution for gathering user feedback and insights.

AWS: Cloud computing platform for deploying the application and managing data storage.

Splunk: Software for searching, monitoring, and analyzing machine-generated data.

Split: (Potential) Feature flagging and experimentation platform for testing and integrating new features.

**Hardware Components:**

Physical equipment required for development.

Standard PC & Mac: Used for development and testing of the application.

**Software Language:**

Programming languages employed in the project.

Primary Language: Python for backend functionalities.

Frontend Languages: HTML & CSS for designing the user interface.