**Software Implementation and Testing Document**

**For**

**Group 17**

Version 1.0

**Authors**:

Bryce Beck

Mason Hartle

Phuong Nguyen

# Programming Languages (5 points)

Python – The sole programming language used for the App driver and the database initialization file. We chose Python because our team is comfortable with the language and are familiar with Flask for webpage development.

Other supporting computer and database languages/applications:

CSS - Used for providing styling to our web pages and for <div> functionality. Chosen for its ability to work with HTML and our increased comfort with this over JavaScript

SQL - Used for the database component of our project and utilized with sqlite3 library within Python. Chosen for familiarity and ease of use.

HTML – Used for the template web pages which are utilized by Flask. We also chose HTML for our team’s familiarity with the language and how it works alongside Python and CSS.

# Platforms, APIs, Databases, and other technologies used (5 points)

Google Books API - We are using the Google Books API to retrieve all book data and information.

SQLite database - We are using SQLite database to manage and store crucial data such as user information.

**Although #3, 4, and 5 are not required for Increment #1, we answered them as best we could for this increment:**

# Execution-based Functional Testing (10 points)

We performed execution-based testing by running the application when new code is being written to ensure basic functionality and check for errors. We test things such as attempting to Log in and ensure that the user stays logged in when changing pages as well as when the user is logged out they are viewing the website as a guest. We are also able to keep the data of the users after log out and sign in again, and reset the password through reset and test this through repeated runs of the program without resetting the database.

# Execution-based Non-Functional Testing (10 points)

So far we have tested that the website UI is user friendly and easy to use. We test this by logging in and out, navigating to different pages, and changing the size of the browser window. As a result of not having implemented the functionality on which our other non-functional requirements rely, we are unable to test for them.

# Non-Execution-based Testing (10 points)

We contact each other through Discord to inform each other about what we’ve done and test each other’s source code to detect errors. Additionally, we perform walkthroughs by reading one another’s commits to the GitHub repository and use commit messages and merge comments to communicate what is being achieved with each commit.