

# Test Cases

## 1. Add Book

Description: Test the functionality to add a new book to the library.

Steps:

- Select option 1 from the menu.
- Enter the details of the new book when prompted.
- Input fields allow any data type

Expected Result: The book should be successfully added to the library database.

## 2. Register Patron

Description: Test the functionality to register a new patron in the system.

Steps:

- Select option 2 from the menu.
- Enter the details of the new patron when prompted.
- Input fields allow any data type

Expected Result: The patron should be successfully registered in the database.

## 3. Check Out Book

Description: Test the functionality to check out a book by a patron.

Steps:

- Select option 3 from the menu.
- Enter the patron ID and book ISBN when prompted.
- Input fields check for a valid ID and ISBN, if invalid return the user to home, else give the user a transaction ID, set the book to unavailable, and the book into the user's borrow history.

Expected Result: The book should be checked out to the patron, and its availability status should be updated.

#### 4. Return Book

Description: Test the functionality to return a book by a patron.

Steps:

- Select option 4 from the menu.
- Enter the transaction ID when prompted.
- Input fields check for a valid ID, if invalid return the user to home, else set the book to available, and set the transaction to returned.

Expected Result: The book should be successfully returned by the patron, and its availability status should be updated.

#### 5. Exit Program

Description: Test the functionality to exit the program after usage.

Steps:

- Select option 5 from the menu.
- Closes scanner and program without failure.

Expected Result: The returns to the main menu.

## Observations

- All test cases passed successfully without any to little error.
- The system efficiently handles various user inputs and operations.
- Adding multiple books and patrons did not result in any issues with the database.
- Transactions between the user and database worked without performance issues.
- The interface provides clear prompts for user interaction.
- Error handling catches invalid inputs and provides appropriate error messages.

View TestDocumentation.txt for a walkthrough of the program.