Library Application Status Documentation

Last updated: 2/27/2017 – dls

**Overall Status: Risk - Behind Schedule**

**Application design: Completed**

**Design Requirements: Completed**

**Database Table Design: ~90% completed**

**Platform Configuration: Completed**

**Java Swing proof-of-concept: ~95% completed**

**Java Swing with MariaDB: Behind Schedule**

**Java Application Code: Risk - Behind Schedule**

**MariaDB tables Configured & populated with data (Doug): Completed**

**PostgreSQL DB create tables(Bhavani): Completed**

**Documentation: ~70% completed**

**Overall Status:** There were numerous problems getting the various application components to work correctly and delays determining the platform to use. We should have completed 1/3 of our application coding, but are still working out the proof of concept (While I can display a database table in the GUI, I still need to get the GUI-button to list the table from the database). This project is at risk of not completing.

**Application design:** Six class diagrams have been designed.

**Design Requirements:** Ten design requirements have been identified.

**Database Table Design:** Six database tables have been designed.

**Platform Configuration:** This step consumed quite a bit of the project time.

**Java Swing proof-of-concept:** I can push a java swing button to display a table from MariaDB, but only if I resize the screen.

**Java Swing** is now operational on the client (desktop) using Cygwin. This is a clunky solution, but the only way I was able to get X-Windows working. This step consumed quite a bit of the project time. Bhavani has a Swing menu working on her laptop.

**Java Swing with MariaDB** is partially working. I have consulted with the Java tutor on 2/17/2018 and 2/20/2018 and am making progress.

**Java Application Code** – Update 4/3/2018

* Model objects completed
* Menus created
* Common DAO component for CRUD operations completed
* DB connection is coded and working as expected
* Add user and add book menus and end to end coding completed.

**MariaDB tables** are all configured and have been populated with data.

**PostgreSQL DB create tables:** Bhavani configure tables on her PostgreSQL DB.

**Documentation:** The User guide is a work in progress.

GitHub Repository: <https://github.com/bputrevu/ICS499_Project_Library/tree/master/DOCS>

Java application code Update: 5/1/2018

Added code for following:

1. Adding a new user to the User List in the Library
   1. <https://github.com/bputrevu/ICS499_Project_Library/blob/master/src/UI/AddUserScreen.java>
   2. Updated with outstanding penalty fees
2. Adding a new book to the Book List in the Library
   1. https://github.com/bputrevu/ICS499\_Project\_Library/blob/master/src/UI/AddBookScreen.java
3. User checks-out a book from the Library
   1. <https://github.com/bputrevu/ICS499_Project_Library/blob/master/src/UI/LoanBooks.java>
   2. Creates transaction
4. User returns a book to the Library
   1. <https://github.com/bputrevu/ICS499_Project_Library/blob/master/src/UI/ReturnBook.java>
   2. Creates transaction
   3. Calculates penalty, if any
5. User holds a book
   1. <https://github.com/bputrevu/ICS499_Project_Library/blob/master/src/UI/HoldBook.ja>
   2. Creates transaction
6. Removing a hold on a book
   1. <https://github.com/bputrevu/ICS499_Project_Library/blob/master/src/UI/RemoveHold.java>
   2. Creates a transaction
7. Renewing a book
   1. <https://github.com/bputrevu/ICS499_Project_Library/blob/master/src/UI/RenewBook.java>
   2. Creates transaction
   3. Calculates penalty, if any
8. Printing user’s transactions
   1. <https://github.com/bputrevu/ICS499_Project_Library/blob/master/src/UI/CreateTransaction.java>
   2. Transaction creation done when book is loaned, renewed or returned.

Modified:

Several models: <https://github.com/bputrevu/ICS499_Project_Library/tree/master/src/Models>

<https://github.com/bputrevu/ICS499_Project_Library/blob/master/src/DBScripts/PostgresDBScripts>

<https://github.com/bputrevu/ICS499_Project_Library/blob/master/src/DAO/PostgresDao.java>