_doxudio-

Priyanshu Luhar | David Ayeni | Solomon Anagha

Table of Contents

User Profiles *3*

Client 3

Admin 3

Features *3*

Initial 3

Dropped 4

Potential 4

Relational DB Schema *4*

Application Architecture *5*

Style Guide *7*

Program Flow Diagram *8*

User Profiles

While there can be multiple user bases, such as, students, teachers, librarians, advisors, amateur writers, authors, publishers, etc., they all generally fall under clients. There are two user profiles in practice: Client and Admin.

Client

Any person who doesn't have access central server, "odin" in our case, would be considered a client or an end-user. Their features are most diverse as the application is based around them. Each user will also be rated based upon how much contribution they have made to the database with books, audiobooks, reviews, valid error reports, etc.

Admin

Contrary to above, anyone who has access to the server where all temporary files are stored for transfer and the transaction database along with the review database resides, is called an admin. Their side of the application is generally for observation and correction purposes and is purely functional using the toolset utilized in the development of this application.

Features

Initial

- PDF/Epub file viewer
- PDF-Epub convertor
- PDF tools to isolate a chapter for goal binding
- peer-to-peer file sharing
- Book(File) review system
- Centralized database option to make a book public
- File sharing via websockets through odin(server)
- Audiobook option
- Persistent notes option
- API calls for information on any book that is not available
- Dual database system, local for personal features and global for server features
- Related Books using graph theory

Dropped

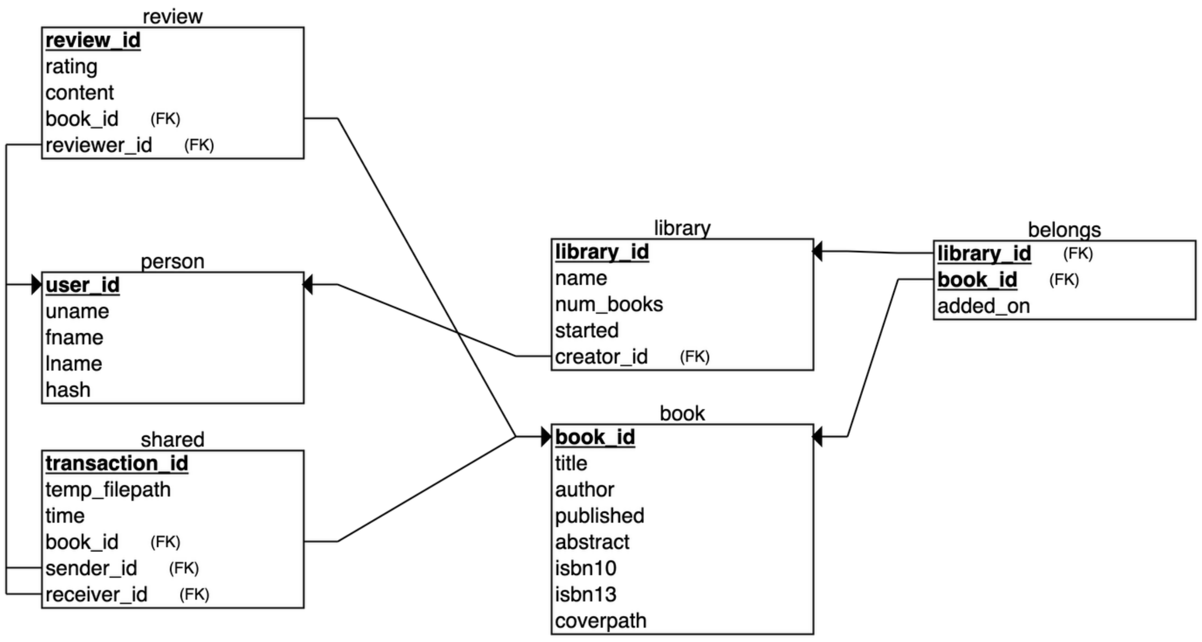
- PDF tools to isolate a chapter for goal binding
- peer-to-peer file sharing
- Audiobook option
- Related Books using graph theory

Potential

- Have note sharing between books and people

Relational DB Schema

Server side:



Application Architecture

```
BEGIN Program

  CALL InitializeGlobals()

  // --- LOGIN SEQUENCE ---
  DISPLAY LoginWindow

  WHILE NOT authenticated DO
    WAIT FOR UserInput(username, password)
    result ← CALL attempt_login(username, password)

    IF result IS success THEN
      SET CURRENT_USER ← result.user_data
      SET authenticated ← TRUE
      CLOSE LoginWindow
    ELSE
      DISPLAY "Invalid credentials. Please try again."
    ENDIF
  ENDWHILE

  // --- MAIN APPLICATION WINDOW ---
  CALL InitializeMainWindow()

  SET current_view ← "home"
  DISPLAY TopNavigationBar [Home, Library, Profile]

  LOOP
    SWITCH current_view
      CASE "home":
        CALL DisplayHomePanel()
      CASE "library":
        CALL DisplayLibraryPanel()
      CASE "profile":
        CALL DisplayProfilePanel()
    ENDSWITCH

    WAIT FOR NavigationInput()
    SET current_view ← selected_navigation_option
  ENDLOOP

END Program

// --- FUNCTION: DisplayLibraryPanel ---
PROCEDURE DisplayLibraryPanel()
  DISPLAY SearchBar
```

-*
*-

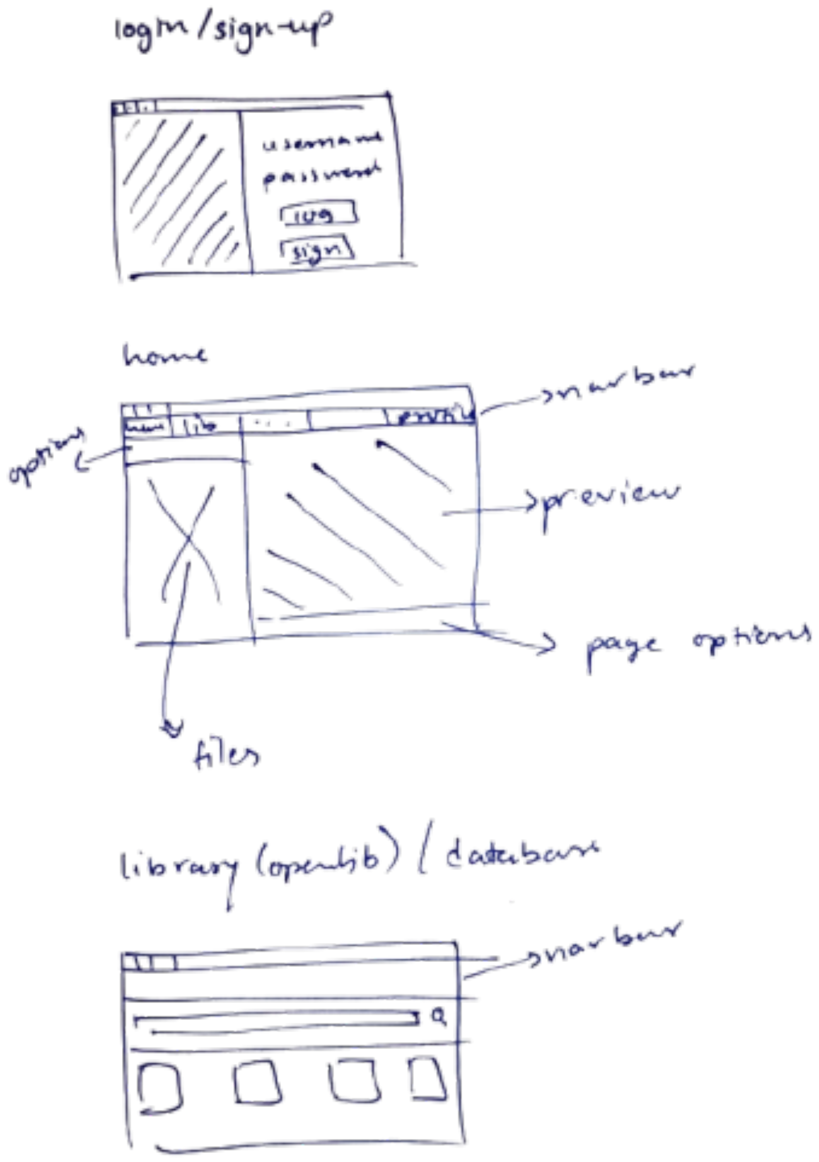
```
  WAIT FOR BookSearchInput(query)
  results ← CALL FetchBooksFromOpenLibrary(query)
  DISPLAY BookCards(results)

  WAIT FOR BookSelection(book_id)
  book_details ← CALL FetchBookDetails(book_id)
  DISPLAY BookDetails(book_details)

  reviews ← CALL get_reviews(book_id)
  DISPLAY ReviewSection(reviews)

  IF AddReviewButtonClicked THEN
    WAIT FOR ReviewInput(rating, content)
    CALL add_review_to_server(book_id, rating, content)
    updated_reviews ← CALL get_reviews(book_id)
    DISPLAY ReviewSection(updated_reviews)
  ENDIF
ENDPROCEDURE

// --- FUNCTION: DisplayProfilePanel ---
PROCEDURE DisplayProfilePanel()
  DISPLAY UserProfile(CURRENT_USER)
  OPTIONAL: DISPLAY UserReviews(CURRENT_USER)
ENDPROCEDURE

// --- FUNCTION: attempt_login (in server_utils.py) ---
FUNCTION attempt_login(username, password)
  user ← CALL get_user_by_credentials(username, password)

  IF user EXISTS THEN
    RETURN success WITH user_data
  ELSE
    RETURN failure
  ENDIF
ENDFUNCTION

// --- FUNCTION: get_reviews (in database.py) ---
FUNCTION get_reviews(book_id)
  QUERY reviews WHERE reviews.book_id = book_id
  RETURN reviews
ENDFUNCTION

// --- FUNCTION: add_review_to_server ---
FUNCTION add_review_to_server(book_id, rating, content)
  INSERT INTO reviews (book_id, rating, content, reviewer_id)
  RETURN success
```

-*
*-

ENDFUNCTION

Style Guide

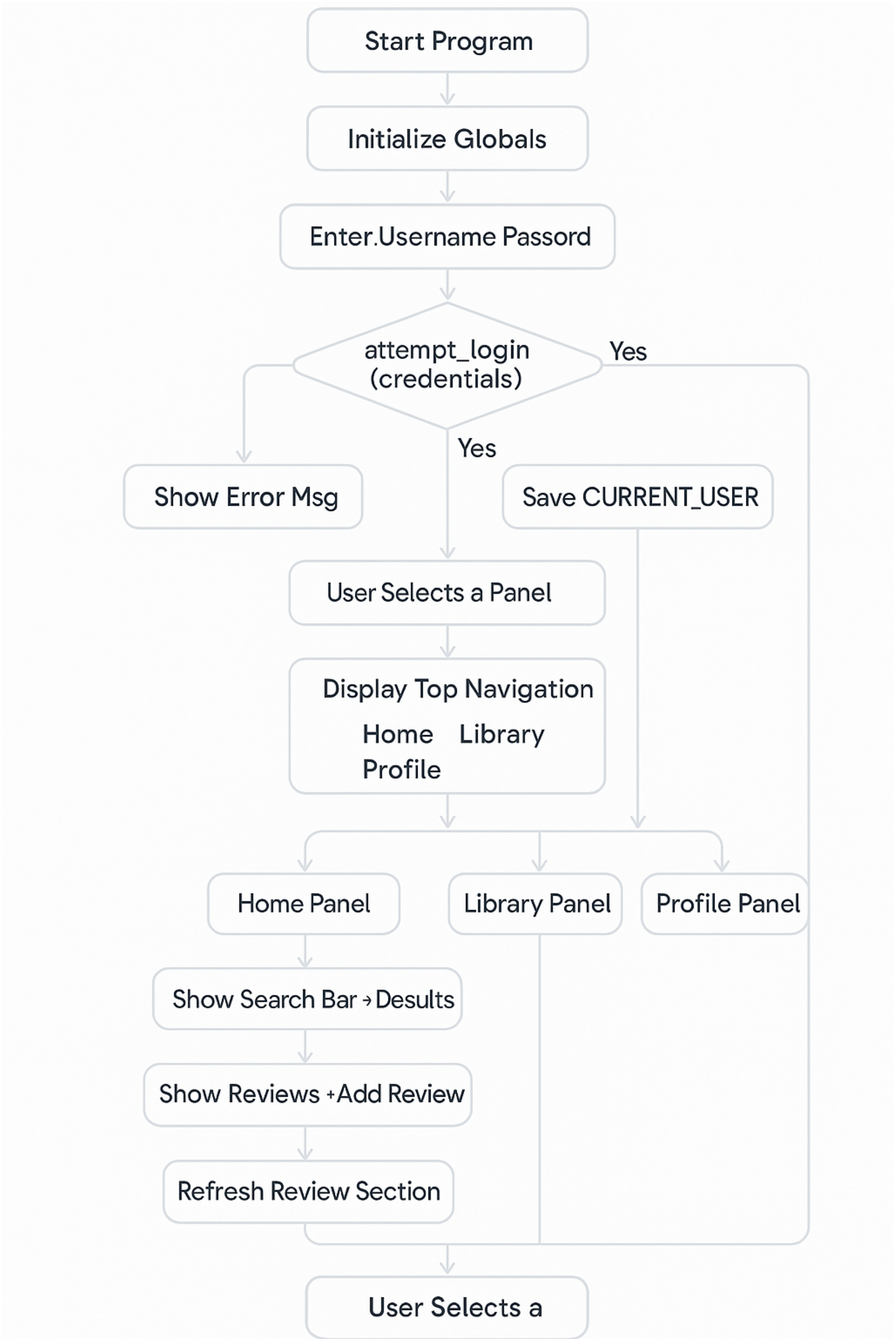
Our style guides were generally drawn on paper and then discussed and implemented. Fortunately, we never pivoted from our original guide which looked something like the following:



Program Flow Diagram

```
+-------------------+
|   Start Program   |
+-------------------+
         |
         v
+---------------------+
| Initialize Globals  |
+---------------------+
         |
         v
+---------------------+
|  Show Login Window  |
+---------------------+
         |
         v
+-------------------------------+
| Enter Username & Password    |
+-------------------------------+
         |
         v
+-----------------------------+
| attempt_login(credentials) |
+-----------------------------+
         |
         v
   +-----------------------+
   | Credentials Correct?  |
   +-----------------------+
     |             |
     | No          | Yes
     v             v
+------------------+    +--------------------------+
| Show Error Msg   |    | Save CURRENT_USER        |
+------------------+    +--------------------------+
     |                  | Close Login Window       |
     |                  +--------------------------+
     |                          |
     +--------------------------+
                                v
                     +------------------------+
                     | Launch Main App Window |
                     +------------------------+
                                |
                                v
             +--------------------------------------+
             | Display Top Nav (Home, Library, etc) |
             +--------------------------------------+
                                |
                                v
                 +-------------------------------+
                 |     User Selects a Panel      |
                 +-------------------------------+
                                |
       +-----------+-----------+------------+
       |           |                        |
       v           v                        v
+------------+ +---------------+    +-------------------+
| Home Panel | | Library Panel |    | Profile Panel     |
+------------+ +---------------+    +-------------------+
                 |                              |
                 v                              v
     +----------------------------+   +--------------------------+
     | Show Search Bar & Results |   | Show CURRENT_USER Info  |
     +----------------------------+   +--------------------------+
                 |
                 v
     +----------------------------+
     | Select Book → Show Details|
     +----------------------------+
                 |
                 v
     +----------------------------+
     | Show Reviews & Add Review |
     +----------------------------+
                 |
                 v
     +----------------------------+
     | Refresh Review Section    |
     +----------------------------+
```

Here is an image of the above flow (look at next page)


Start Program
Initialize Globals
Enter.Username Passord
attempt_login (credentials)
Yes
Yes
Show Error Msg
Save CURRENT_USER
User Selects a Panel
Display Top Navigation
Home Library
Profile
Home Panel
Library Panel
Profile Panel
Show Search Bar → Desults
Show Reviews +Add Review
Refresh Review Section
User Selects a

_ _