Dynamic Library Map

Final Project Report

"Computer Science Projects"

Spring 2018

(Prof. Hanh Pham)

Anthony DiNardi
State University of New York at New Paltz

Project Type: On-campus

Class Key: p-s18-10

TABLE OF CONTENTS

1. Problem Description

1.1 Business Context and Goals

This software will be an improvement of the current software being used at Sojourner Truth Library at SUNY New Paltz. It can be used by students and faculty alike. Users will be able to use this software at anytime and from any computer connected to the school's network.

Currently, this software is able to search the call number of a book in the library's database and be shown the book's location in the library. The software shows the user the floor and bookshelf that the book is located on.

The software will be developed further in order to be more dynamic so that it will display the layout of the library floors and bookshelves based on already existing data that can be added to, edited, or deleted (i.e. moving of bookshelves). As of now, the layout of each floor is hard-coded into the HTML.

BUSINESS PROBLEM:

The library wants to be able to easily change the bookshelf layout on the software so this project needs to:

1. dynamically draw the locations of the bookshelves
2. automatically change the layout when changes are made to the database

Some things the software can do:

- User can login to administrator page to add, edit, or remove data
    - Input = Username & Password
    - Output = Admin page

- User can input a book call number to see the location of that book
    - Input = Book call number
    - Output = Location of book

- Admin can add book or bookshelf locations to the database through the admin page
    - Input = Shelf information
    - Output = New shelf is displayed on the map

1.2 Technical Requirements

- ENVIRONMENT: Software runs in a web browser:
    - Browser PLATFORM: Any modern browser (Chrome, Firefox, Safari, etc)
    - CONNECTIONS: Centralized but connects to school's server => must have Internet connection
    - MEAN OF INTERACTION: keyboard, mouse, touchscreen

- SYSTEM COMPONENTS:
    - User Interface:
        - Basic computer-window display
        - Touch-screen
    - Processing:
        - Login mechanism for admin:
            - Input = username/password => input with keyboard
            - Process = compare inputted username/password with original user info in database
            - Output = (i) fail when it's not matched, asks user if they want to retry (ii) redirects user to admin page
        - Dynamic mapping:
            - Input = shelf information from database
            - Process = iterate through shelf location data and generate HTML code for layout
            - Output = visual representation of library floor based on data

        - Data:
            - Format of Data = MySQL database

- Tables: bookLocations, shelfLocations, permStructLocations, FeedBack, login

1.3 My Responsibilities

- Develop a way to make the layout of the library dynamically draw itself based on the existing data

2. Technologies

2.1 Related Technologies

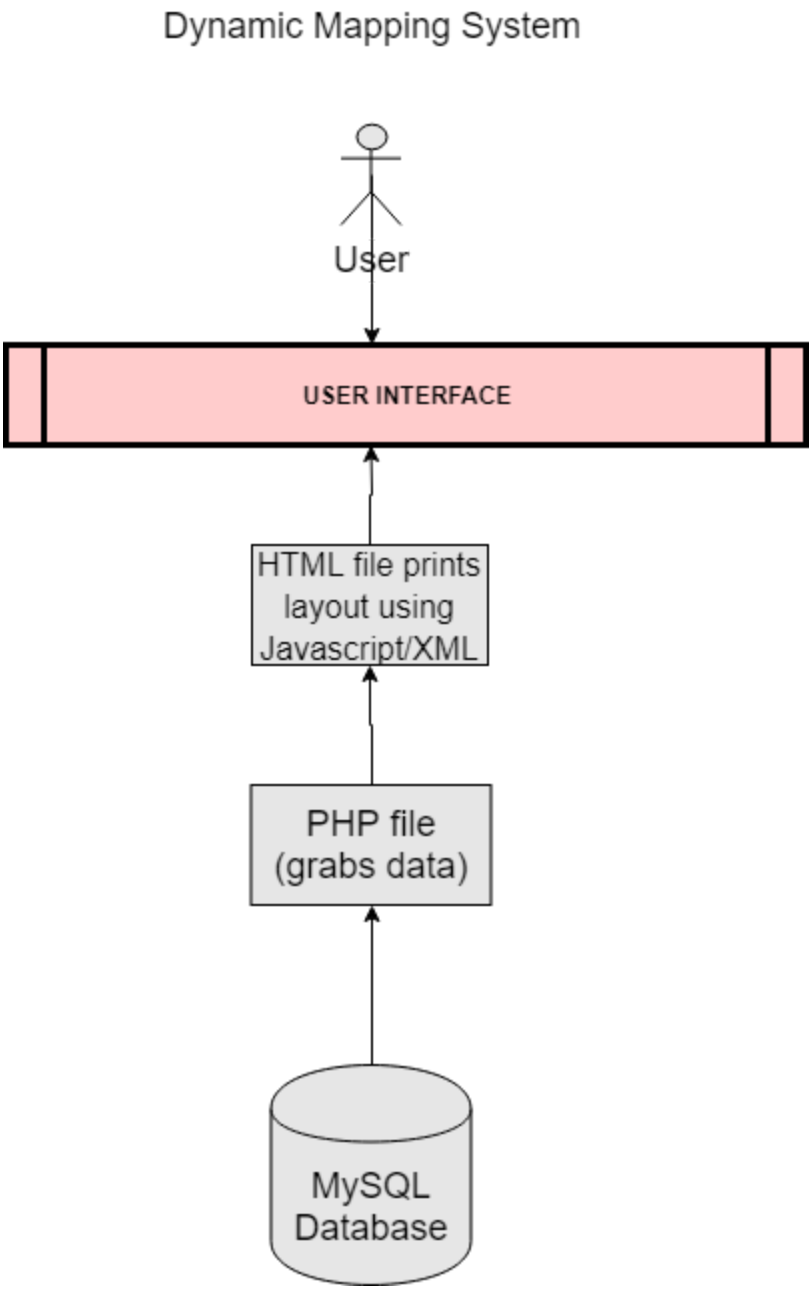
- HTML
- CSS
- Javascript

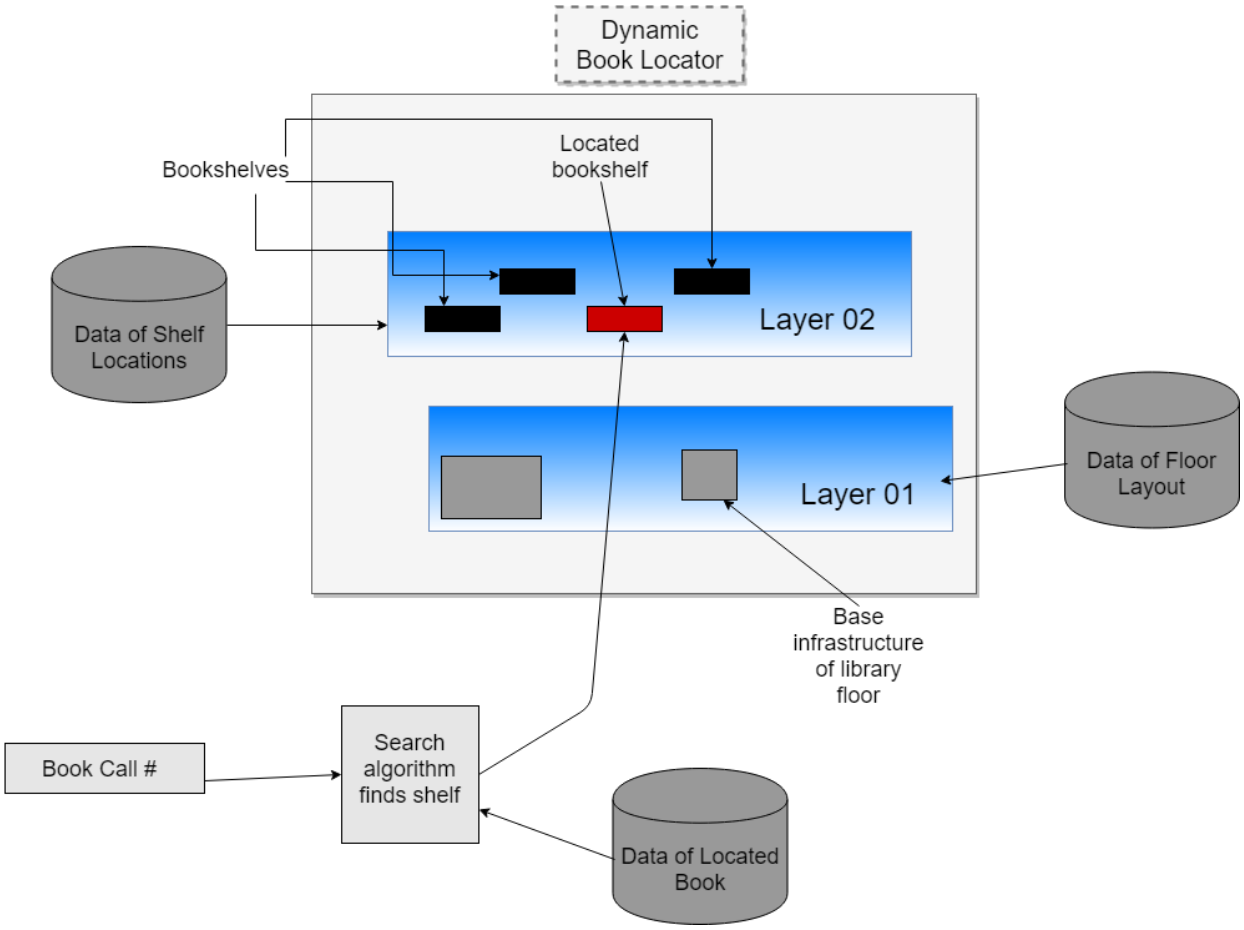
2.2 Newly Learned Skills/Technologies

- PHP
- MySQL
- XML
- SVG

3. Design

3.1 System Architecture





3.2 Programming/Coding Components

- **index.html** - main webpage, Javascript that searches for the book location and calls the PHP scripts

- **mainFloorDisplayer.php/concourseDisplayer.php/groundFloorDisplayer.php** - pulls shelf locations from MySQL database and echos each row of data as a string containing a rectangle SVG element with x, y, width, length attributes

- **bookLocations.php** - pulls the book location data from the database and echos the data as a long string so the search algorithm can parse it

4. Software/System Description

4.1 Displaying Bookshelves

The instant the webpage is loaded, a Javascript function is called three times, once for each map. The function is passed the div where the bookshelf SVG elements will be placed and the name of the PHP file that will display that floor. An XML HTTP request is made and opens the passed PHP file name. When the PHP file is executed, it makes a connection to the database using a seperate file in the Admin folder. The PHP file for all shelf locations for that specific map. A while loop is used to iterate through the rows in the database. If a row is not missing a value, then the values are used to echo a rectangle SVG which is sent back to the Javascript XML function which places the SVG in the specified div that was passed through the function call. This is repeated for every row in the database for each map.

4.2 Book Searching

To search for a book, the user can type in a call number. When the search button is clicked, a Javascript function is called which creates an XML HTTP request and opens the book locations PHP file. The PHP file connects to the database through the external file in the Admin folder and queries for all the book locations. A while loop then iterates through each row and assigns every value to a variable that will be passed back to the XML request for parsing. The algorithm splits the raw data and creates shelf objects for each row and assigns the data values accordingly.

Afterwards, the algorithm parses the call number that the user entered and compares that string to each shelf object that was just created to find which shelf the entered call number belongs to. When the shelf is located, a pop-up box is displayed, telling the user which floor and shelf number the book is on. The located bookshelf is also changed to a red color on the map.

5. Test Results/Observations

5.1 Performance

The software shows no signs of hiccups, lag, or long wait-times. Each web page is fast and responsive. The maps are all on the same HTML file but are just hidden or shown depending on what the user wants. Quick transitions between the three maps are achieved by setting the display attribute for that map to “block” to show it, or “none” to hide it.

The software also has great performance in terms of fetching data. It interacts with the database once for each map (3), once for every search function, when the admin adds, edits, or deletes an entry, and when the data is displayed on the admin page. The bookshelves on each map are displayed instantaneously.

The login system also works very quickly and logs the user in immediately after submitting a valid username and password. The admin page is loaded instantly and if the user clicks the logout button, they are brought back to the main map page in a fraction of a second.

5.2 Quality

The overall quality of the software is very good. The UI is simple, visually pleasing, and easy to use. The maps are clear and easily show the user where the located book is. The homepage has an approachable, almost friendly, style. There is even an About section of the homepage that gives instructions on how to use the book locator.

The admin page could use some more polishing. Compared to the homepage, the admin page is more jagged and rugged looking. I suppose this is fine because only a select few people will be using that page. However, I believe it still could use some more styling. Besides that, it is fully functioning and has no bugs.

6. Professional and Career Benefits

I believe that this project will help me get an internship/job in web development or general software engineering. I would prefer to do software engineering but at the beginning of my career, I will take any experience I can get my hands on, whether it is paid or unpaid. I would be able to show this project to a company and give them a demo on how it works, especially since it is on the live server. I put a link to the project on my resume and LinkedIn so employers would be able to see my demonstrated skills.

7. Conclusion

I'm very proud of the progress I have made on this project. It provided a huge learning experience for me in terms of using new technologies, how a full-scale project works, and how to improve on pre-existing work. I am glad that I have some experience using databases and programming back-end services like the bookshelf displayer. Having this project on my resume will definitely help me begin my professional career. I'm so happy that I got to work on something that a lot of people will be using in the future.

8. References

H. Pham, Projects. [Online]. Available: http://cs.newpaltz.edu/~phamh/apc/. [Accessed: 09-May-2018].

D. Sklar, Learning PHP 5. Sebastopol, CA: OReilly, 2004.

"W3Schools," W3Schools Online Web Tutorials. [Online]. Available: https://www.w3schools.com/. [Accessed: 09-May-2018].

"Where Developers Learn, Share, & Build Careers," Stack Overflow. [Online]. Available: https://stackoverflow.com/. [Accessed: 09-May-2018].