Fast Pages

Project Report

By

John Hoang 014411072

Western Han 012261964

Aryan Kumar 012162371

Problem Overview:

We believe that there should be an easier way to let students access the books they may need throughout the semester. To make a solution to this our project is to make a website using Spring Boot, Java, AWS, and Docker to make an online book catalog for books that may be needed in a class by SJSU students.

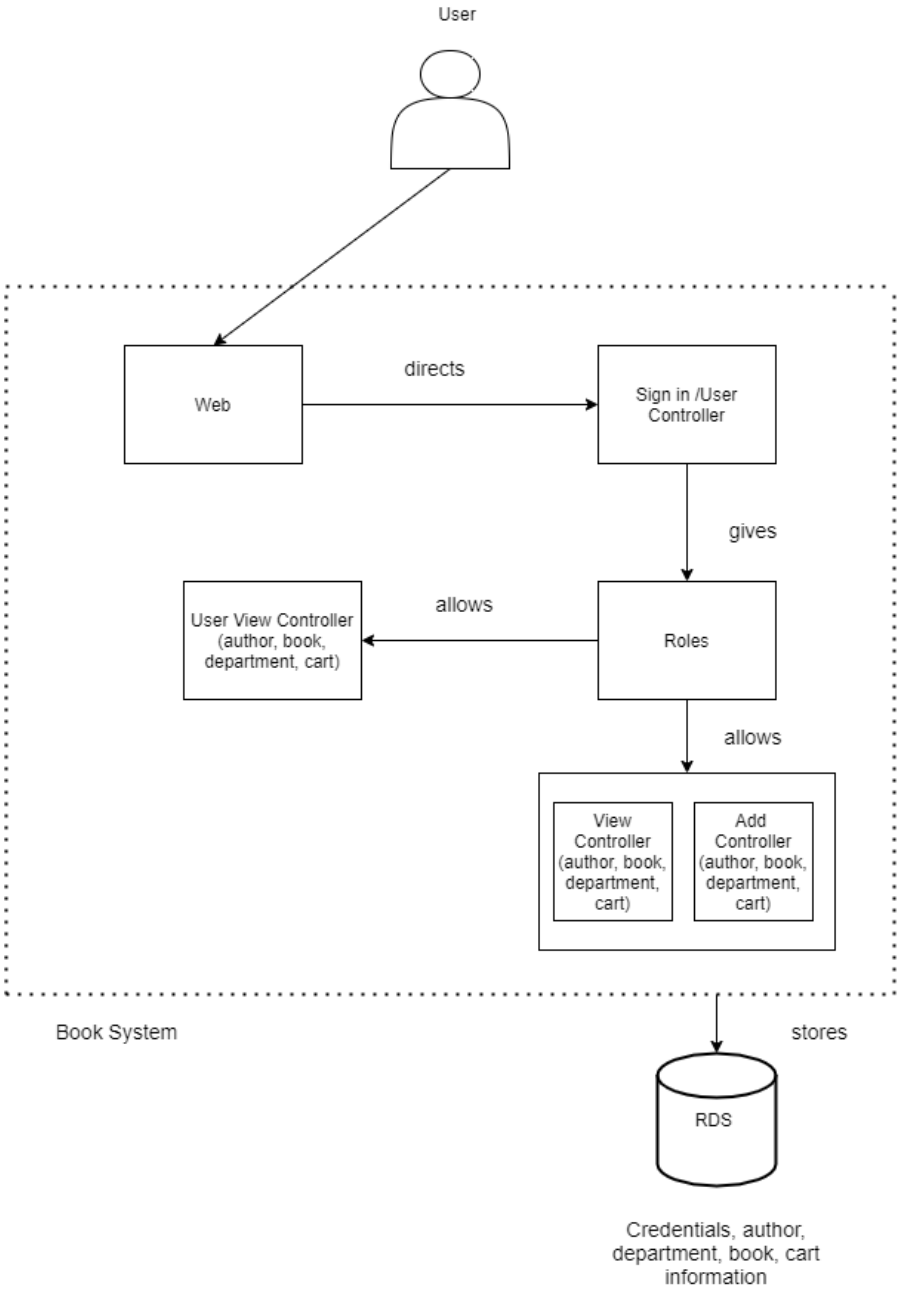
Project Description:

We implemented our solution using a three-tier architecture for presentation, application, and data access. At the presentation layer, we had code to have the user navigate through the site so this is where we stored out HTML, CSS, JS, and other frontend related files for the application.

The application layer is used to connect web elements to web servers so this is where we have our Spring Boot, Java, AWS, and Docker elements.

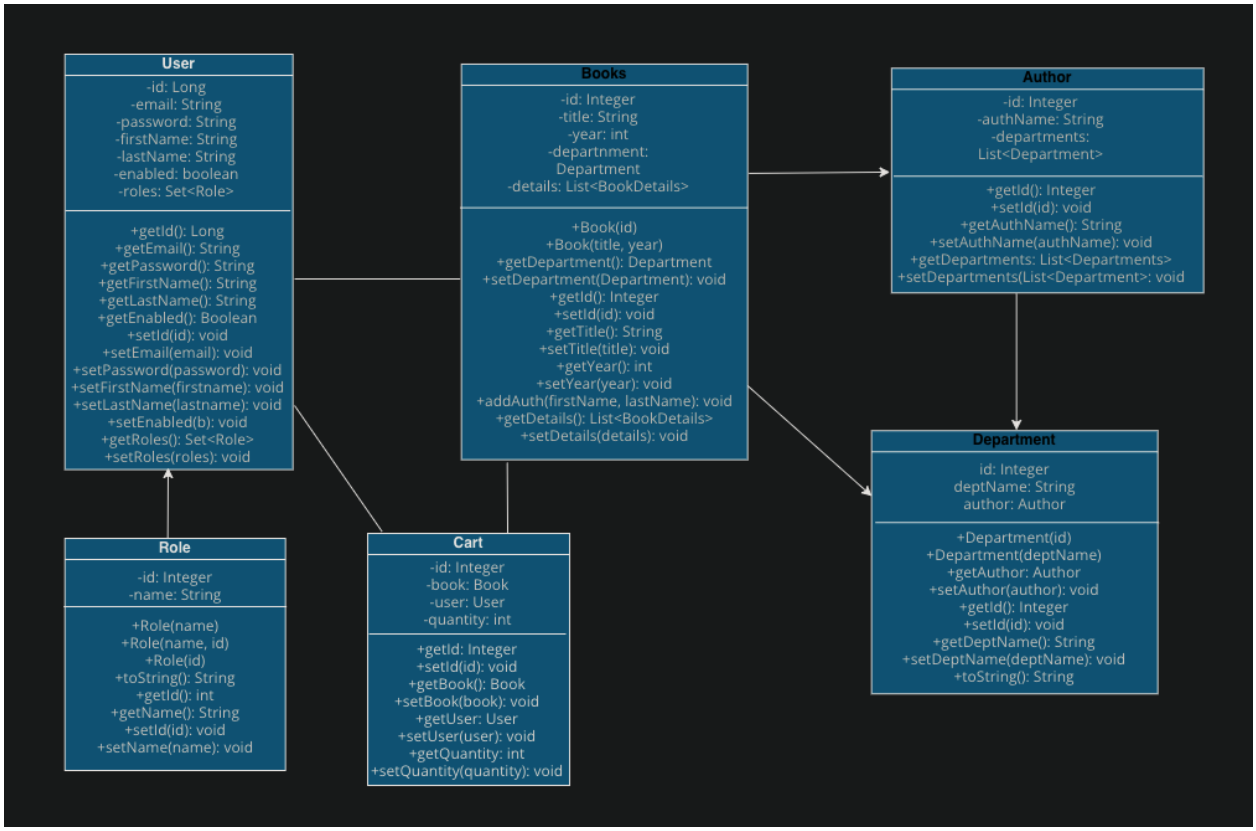
Finally, the data-access layer is where we hold the information related to our user and the data related to the books. In our application, the data layer is managed by Amazon RDS and MySQL.

System Diagram:

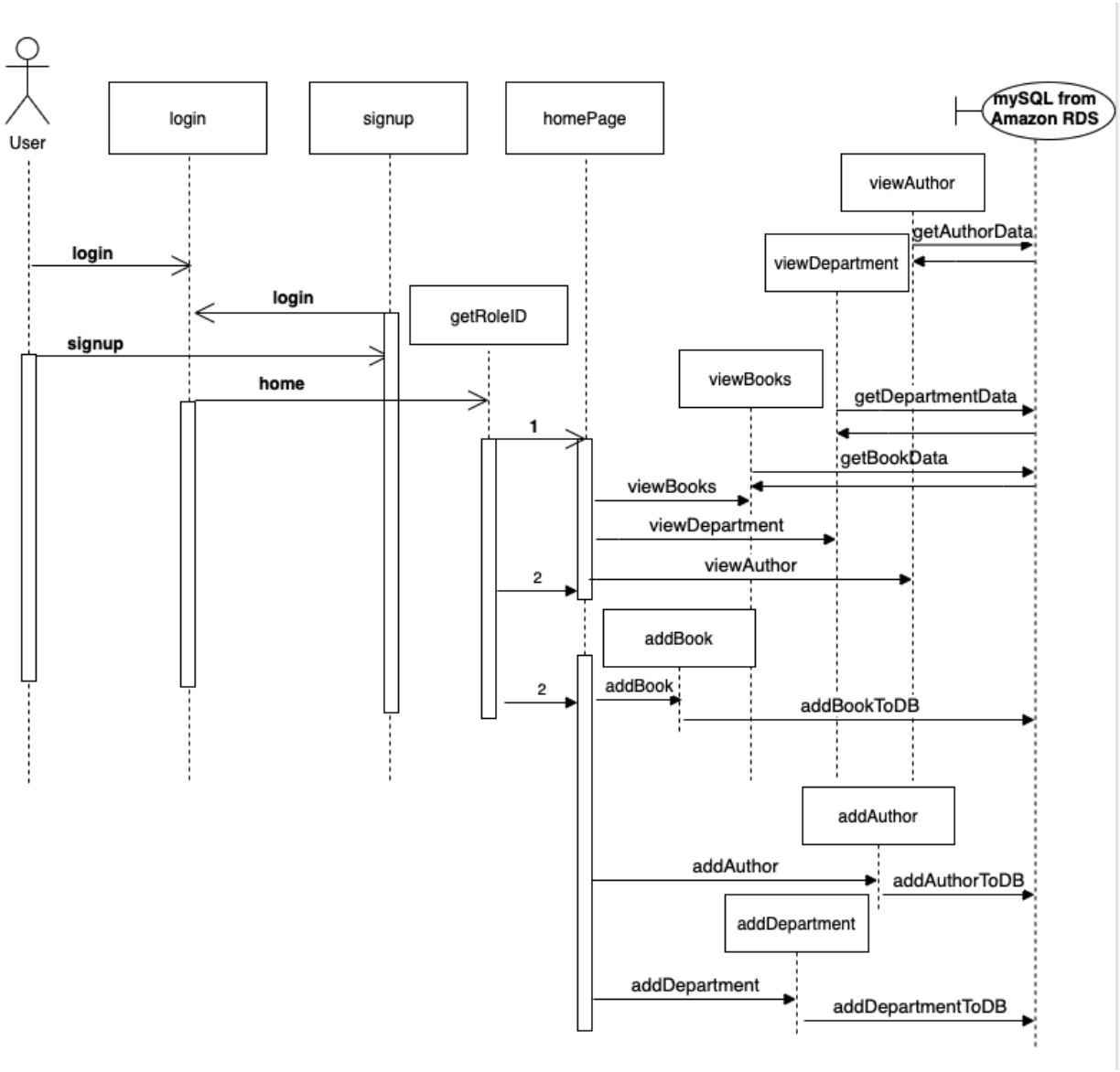


Class, Sequence, and interaction diagrams:

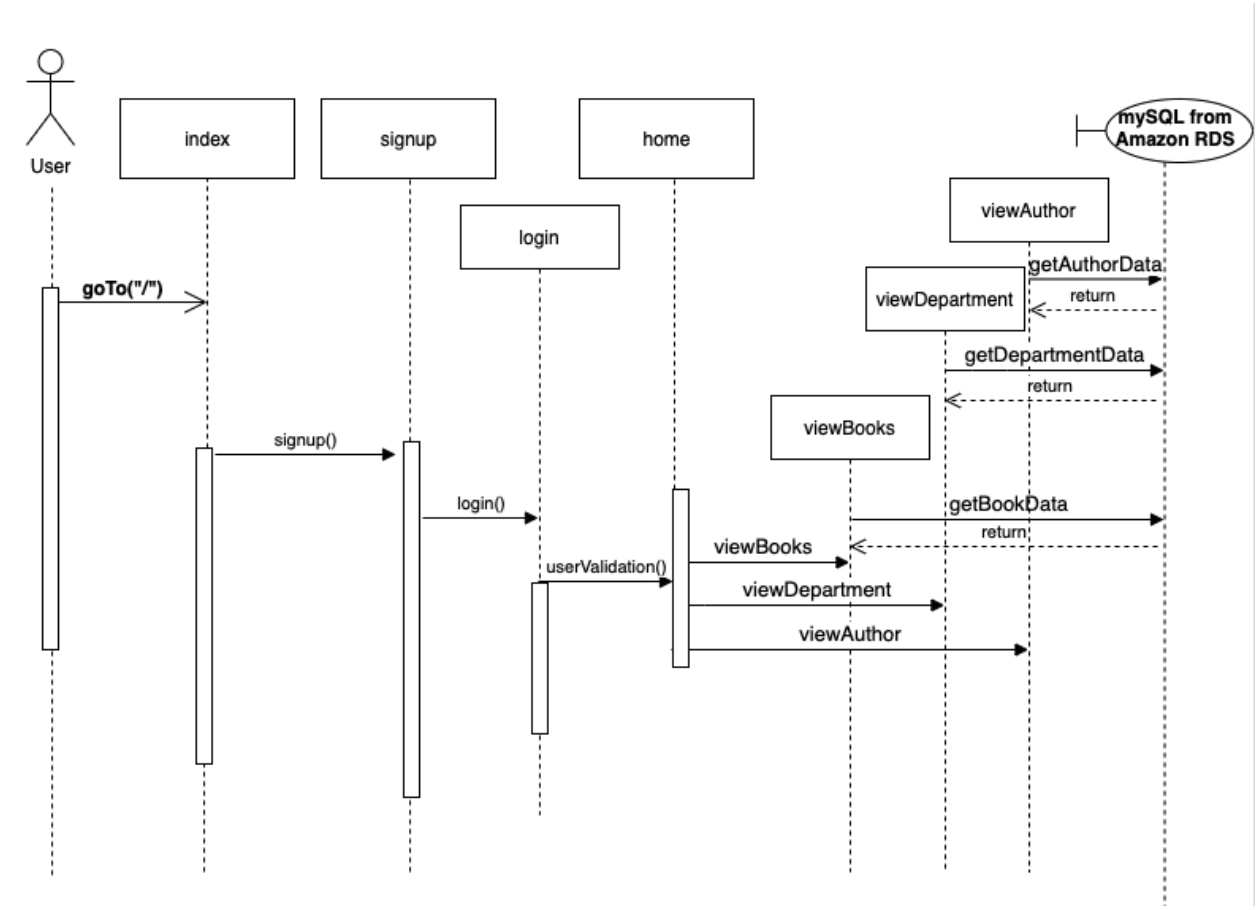
Class Diagram:



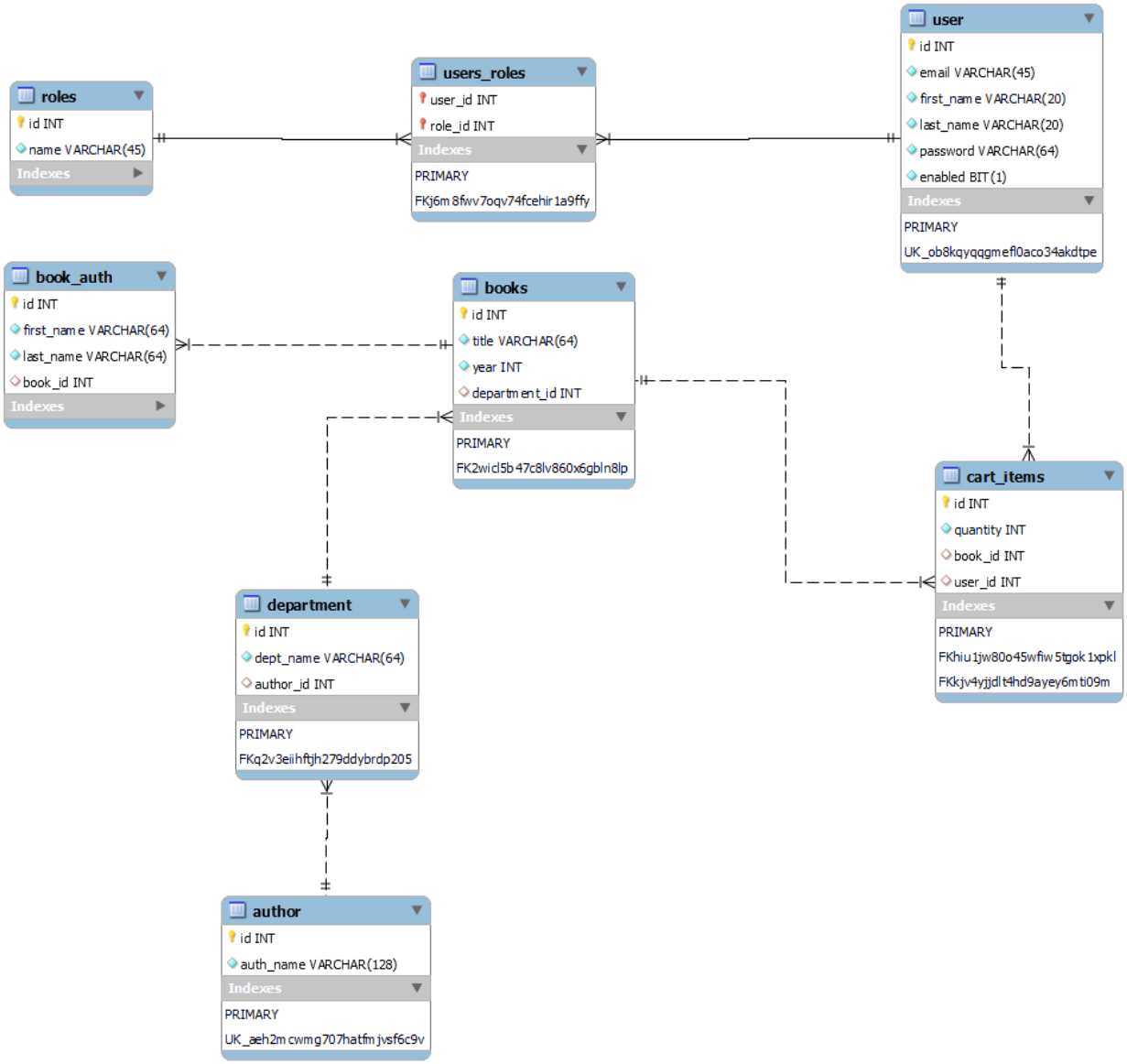
Sequence Diagram:



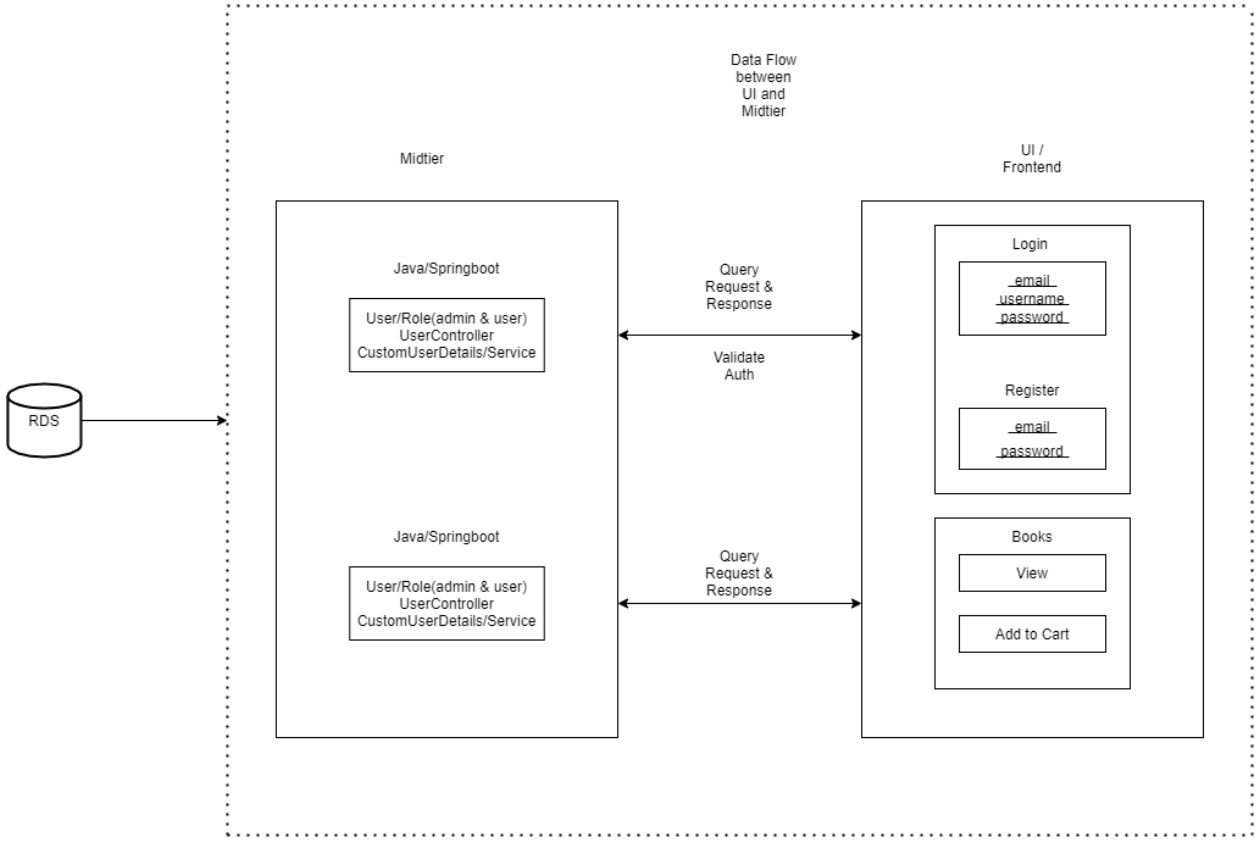
Interaction Diagram:



Database Schema diagrams, highlighting the interactions between the different tables



UI data transport



Team Member Contributions:

John	Aryan	Western
-Database Schema and Tables -Table Mapping, HTML mappings -Login/Logout/Register -Add/Remove feature -Add to Cart feature	-Admin role - Cart Features -Add/Remove feature	-UI/UX -Database population -Profile

Next Steps/Lessons Learned:

With CMPE 172 lessons, we were able to learn to connect the three-tier architecture. We learned to utilize Spring Boot Java (mid-tier) with HTML, JS, and CSS (presentation), and connect with the AWS RDS database (data-access) to store user and admin information. It was in some way similar to building a web with python flasks but a lot more complicated since it was different.

With a deadline extension, we would have sufficient time to populate our database with more books, allow users to search up their desired books with certain filters, build a dashboard for admin analytics, and have a more fleshed-out UI. We could use that extra time to learn how to scale our database to have room to store more data like profile pictures, book checkout logs, and more. One major flaw we failed to implement was error handling and error mapping to check for invalid data entries or in case of site errors.

Relevant Links:

Links to the GitHub upload

https://github.com/jawnhoang/FastPages

Public URL to the application

http://fastpages.us-west-1.elasticbeanstalk.com/

Test account credentials

-Register a new user account

ADMIN ACCOUNT

User: admin@gmail.com

Pass: password

Public Video link

https://youtu.be/7Jbep9NJlQU