Design Document – Library Portal

Kavin Kuppusamy – kxk190026

DATABASE DESIGN CS 6360

SYSTEM ARCHITECTURE

The Library Management System has been designed for Librarians to carry out various services in a library. Each functionality has been explained as follows:

1) Search for a Book

a. The Librarian can enter either an ISBN, Title Name or Author Name to search for a book.

b. Substring matching is performed to obtain all possible matches.

c. Availability of book is also mentioned with the respective book.

d. If either nothing or a space is entered in the text box, the search results in all the books that are present in the library.

2) Check Out Books

a. The Librarian can check out books for a borrower.

b. A borrower cannot check out books if the book is unavailable or he/she has already issued 3 books or he/she has a fine pending.

c. Once checked out, the system automatically sets the due date to be 14 days from the issue date.

3) Check In Books

a. The Librarian can check in books for a borrower.

b. To search the book loan entry, the librarian can enter either the ISBN, Card Id or part of the borrower's name to check-in the book.

4) Fines

a. Given the Card Id of the borrower, the system helps the librarian retrieve the history of fines for that borrower.

b. It provides option to display all the fines that are due for each book issued by the borrower, the current total fine the borrower has to pay and the fines that have been paid by the borrower in the past.

c. Every time a fine is paid, the system will update the data base indicating that that particular fine has been paid.

5) Add new Borrowers

a. The system allows the Librarian to enter new borrowers into the system.

b. Basic details (Full Name, Ssn, Address, Phone) are entered to generate a new Card Id for the borrower.

c. A person is allowed to have only one Card Id.

Design Overview
This application is a website that can run on local web server inside a library and possibly on internet. The front-end is developed in HTML and CSS along with Bootstrap, JQuery and Javascript. The backend is developed in python using Django. MYSQL database is used as a DBMS. Django uses MVC framework.

Architecture
Django has MVC architecture:

The **model(M)** is a model or representation of your data. It's not the actual data, but an interface to the data. The model allows you to pull data from your database without knowing the intricacies of the underlying database. The model usually also provides an abstraction layer with your database, so that you can use the same model with multiple databases.
The **view(V)** is what you see. It's the presentation layer for your model. On your computer, the view is what you see in the browser for a Web app, or the UI for a desktop app. The view also provides an interface to collect user input.
The **controller(C)** controls the flow of information between the model and the view. It uses programmed logic to decide what information is pulled from the database via the model and what information is passed to the view. It also gets information from the user via the view and implements business logic: either by changing the view, or modifying data through the model, or both.

Libraries Used
Frontend : HTML, CSS, Bootstrap, JQuery, Javascript
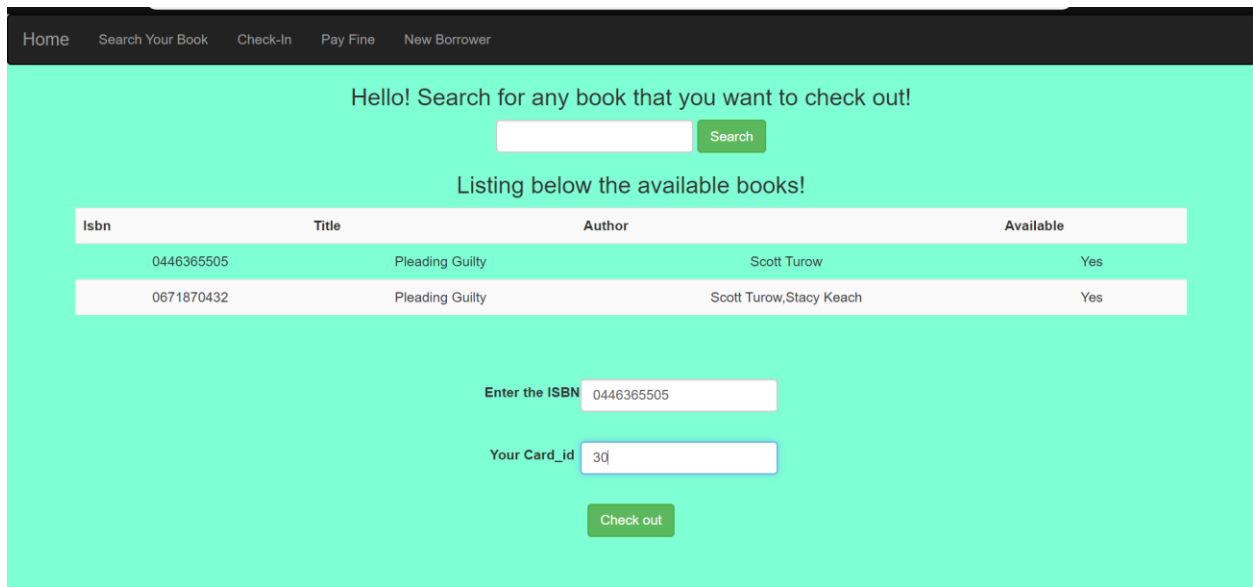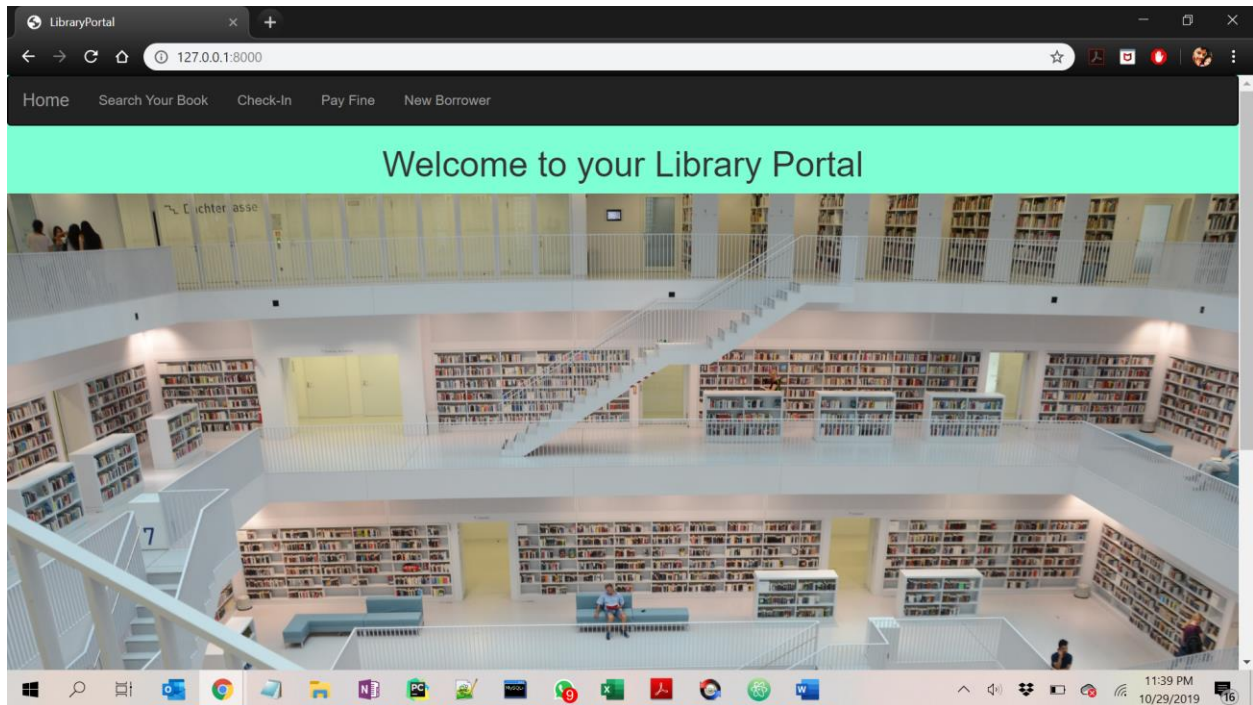Backend : Django, pymysql, mysqlclient, pandas
Database: MySQL

Data Design and Assumptions
'Availability'(0 or 1) column is added in 'Book' schema to keep track of availability of books.
Only ISBN10 has been considered for the whole processing.

Screenshots:

Hello! Please search below for the book to check-in!

31 [Search]

Here's what we found for you!

Isbn	Title	Author	Card Number	Borrower Name	Ssn	Check In
0446365505	Pleading Guilty	Scott Turow	31	Kavin	111-11-1111	Check-In

---

Wanna pay the fine? Enter your Cardno or Name or SSN

[Search]

Heres what we found

Card Number	Ssn	Full Name	Fine Amount	Pay
31	111-11-1111	Kavin	2.00	Pay

Hey! Do you want to create a new Borrower? Fill out correct Info below!

Full Name Kavin

SSN 111-11-1111

Address 955,W.President George Bus

Phone 3344974225

[Submit it!]