|  |  |
| --- | --- |
|  | 2021-22 |
| **School of Computing, Engineering and Mathematics** |
|  |
|  |
|  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | **5001CEM**  **Software Engineering**  **Test Regime Template** |
|  |  |

**Test Regime**

**1 Developer name / SID**

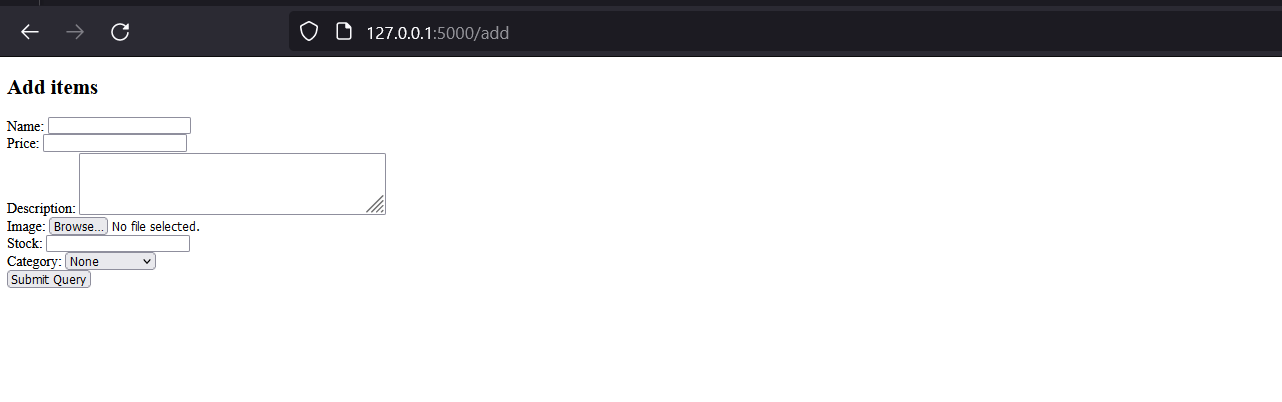
**2 Project**

My project is about implementing an online bookstore using Flask. The system is a simple web application consisting of a storefront, customer login, shopping cart and checkout section. The app was implimentent using Flask, html, css, javacsript and sql for database.

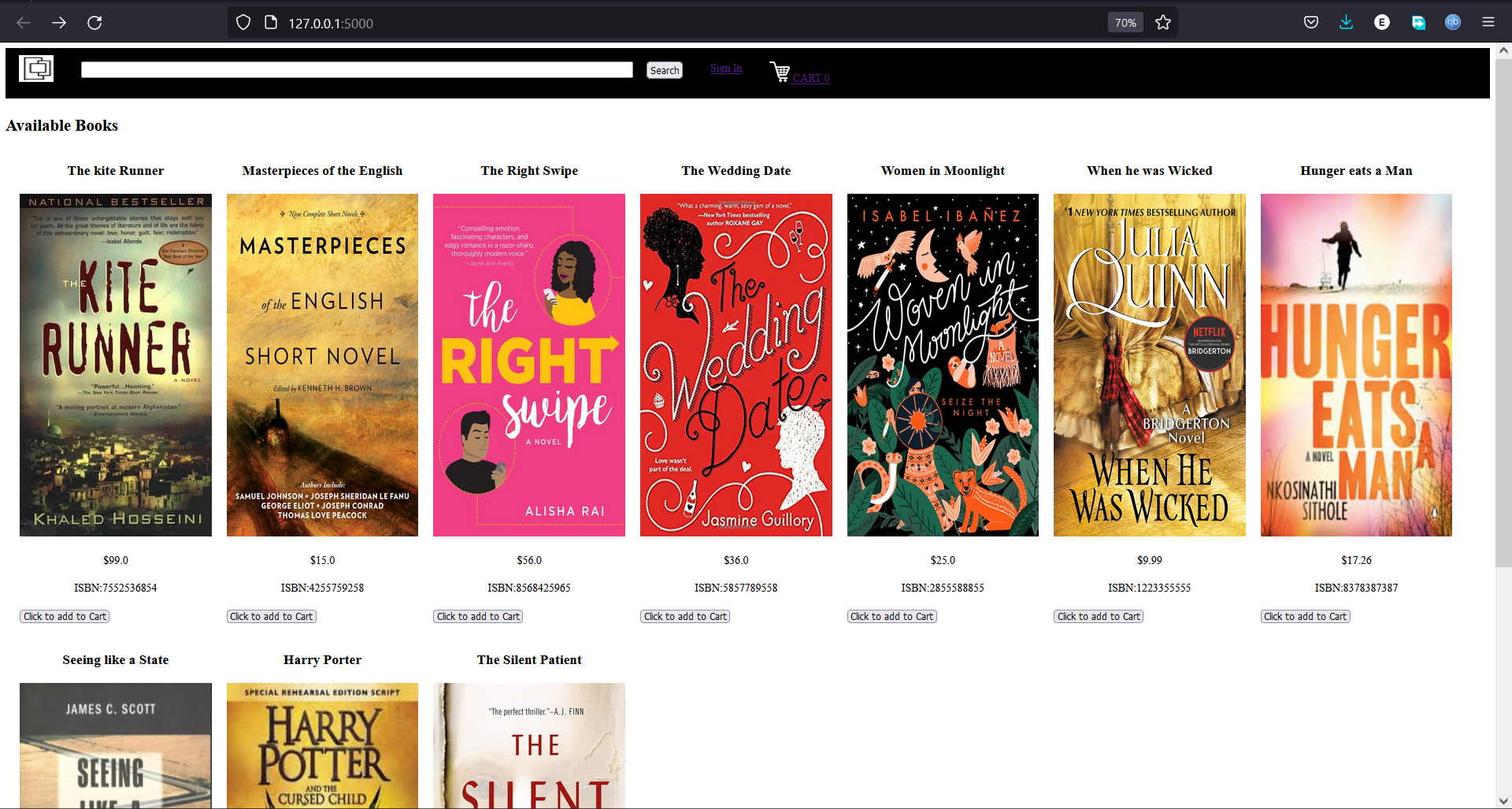
For security feature I used password hasshing and authentification logings for the customers.

Here is the program screenshots.

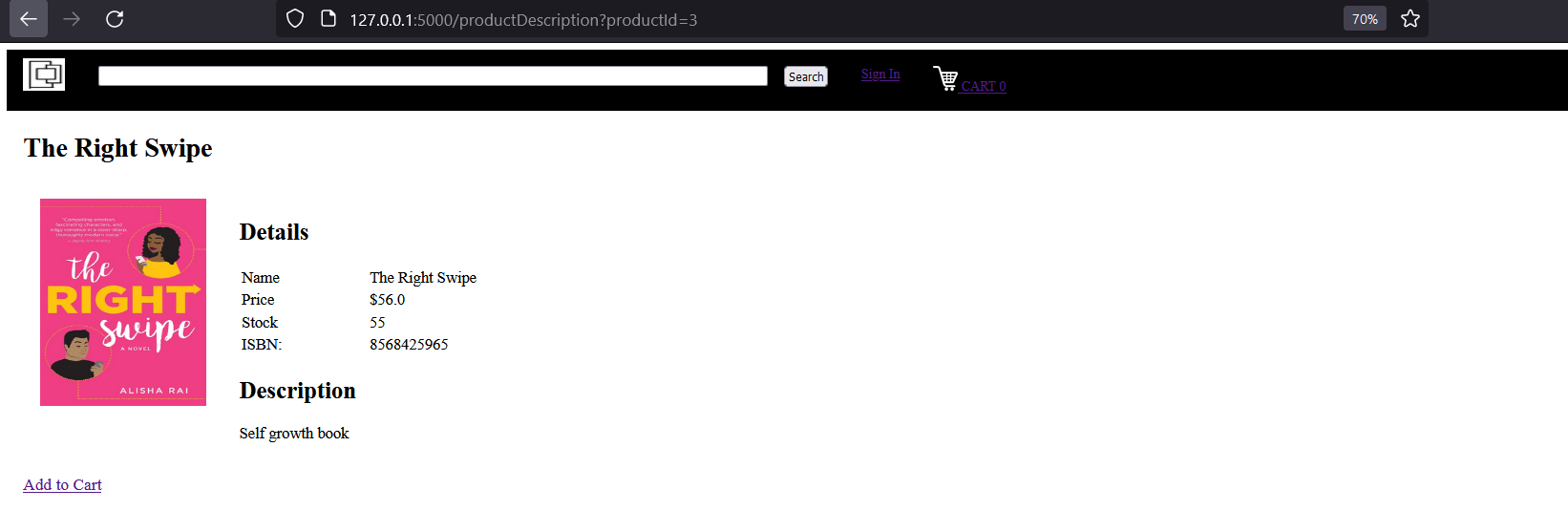
**Admin**



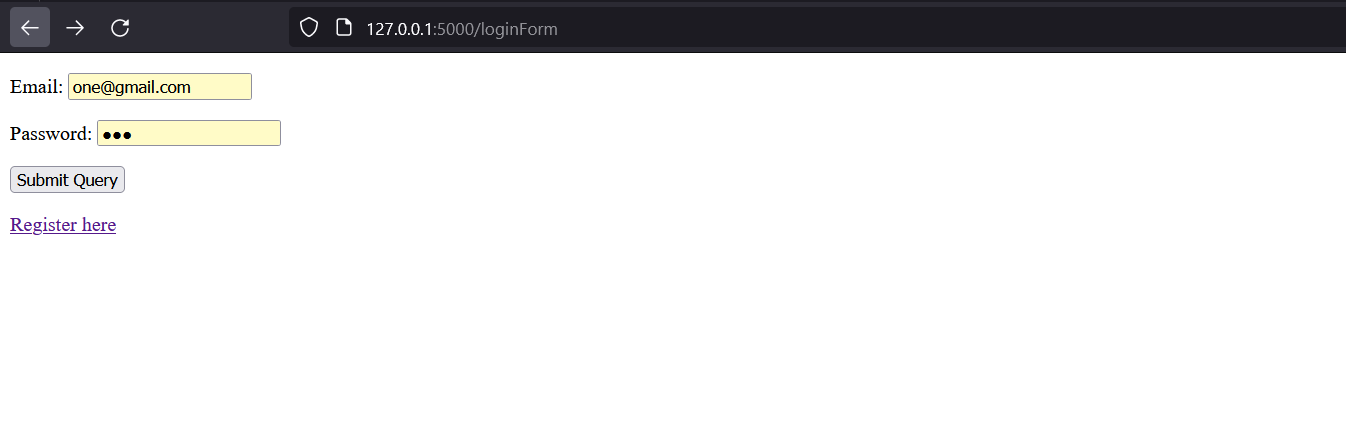
**Shopfront**



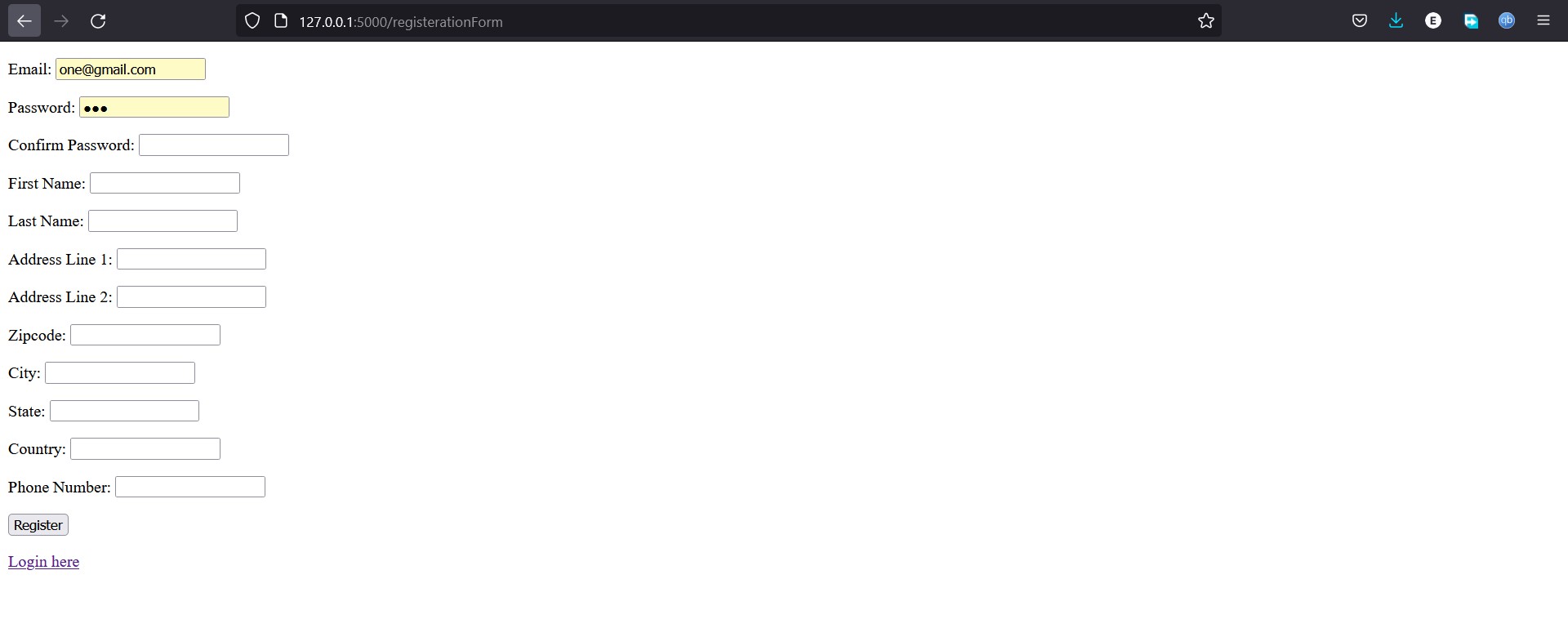
**Add to cart**



**login**



**Registration**



**3 Test scope**

The follong tests were performed to test the app basic features.

**Registration**

I tried to registeser a new user and the operation was sucesful.

**Login**

I logged in the new user created and was succesfully loged in.

**Shopfront**

The shop front was displaying all the books feature and also the search buttong was present and funtioning correctly.

**Cart**

Inside a users account I tested adding an item to cart and the items were added succesfully.

**Check out**

After adding items to cart there is a button to direct the user to check out where there is summary details about his or her oders and total amount payment.

3.1 out of scope

Security of the system was tested where I tried to retieve the user password from the database but the passwords were hashed.

**4 Test time**

# how long the total set of tests should take

**5 Test regime**

# here we define the set of tests

# the tester should be provided a means of reporting findings simply and effectively

# the following are examples only

4.1 Registration (10 mins)

# relevant tests

4.2 Login

# relevant tests

4.3 Shopfront (20 mins)

.1 System deals with n-sized data

Test 1: inspect number of database records and cross reference to display. These should match

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Yes | Fail | No | Comments | All the items matched |

Test 2: use alternative database with expanded records and repeat Test 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Yes | Fail | No | Comments | The items were updated |

.2 Images are correctly displayed

Test 1: ensure that images are correctly resized and fit bounding box

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Yes | Fail | No | Comments | All image thumbnail were accurate |

.3 Currencies are correctly displayed

Test 1: interface should display correct symbol for currency ($, £ etc.)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Yes | Fail | No | Comments | $ sign was used |

.4 Prices are correctly displayed

Test 1: Amounts should agree with database and be formatted in the same way (2 decimal points)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Yes | Fail | No | Comments | The price amount matched correctly |

4.4 Shopping basket

.1 Items can be added and removed

Test 1: Add item

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Yes | Fail | No | Comments | Item was added to cart |

Test 2: Remove item

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Yes | Fail | No | Comments | Items were added to cart |

4.5 Checkout

.1 Payment is accepted

Test 1: Pay for items in basket; payment accept or cancel screen displays

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | No | Fail | Yes | Comments | Payment summary was displayed but there was no option to accept user payment. |