

Javeria Mohammed Nasir  
[jmohamme@hawk.iit.edu](mailto:jmohamme@hawk.iit.edu)  
ITMD-562-Final

### Project Description:

This is a Book Inventory project created in Laravel Framework. It is basically a CRUD operations implemented using Laravel. With the help of this project user/admin can add new book, edit book, update book and delete book. Only admin can perform the entire CRUD operations. This project uses Sqlite database.

This project contains one controller:

**Books Controller:** This contains all the CRUD methods in it. Like: index, edit update, create, show and delete methods.

**Books Controller has five views:**

**create.blade.php:** On click on add new book button, will redirect to create view.

Which takes all the inputs for fields from the admin. On click of create button on this page will store the data in SqlLite database.

**edit.blade.php:** This view basically lets the admin to edit the added book and this will update the existing record in database.

**index.blade.php:** This view shows up the entire data stored in database. This is the main page after user logs in.

**show.blade.php:** On click of view button on index page will redirect to show page where admin can view a particular record selected to view.

**login.blade.php:** This is a login page for admin. If the username and password is incorrect the user is redirected to the same page. Once user enters the right username and password, then on click of submit button it redirects the user/admin to index page.

All the pages has logout link. On click of logout the session destroys.

### Development Environment:

This project is developed under Laravel framework.

System : windows 10

PHP version : 5.5

Laravel version : 5

Sqlite 3

Wamp server

### Installation Instructions:

Unzip the folder place the folder under wamp/www/.

The Username and password used for logging into the application is show below

**Username:** admin

**Password:** password

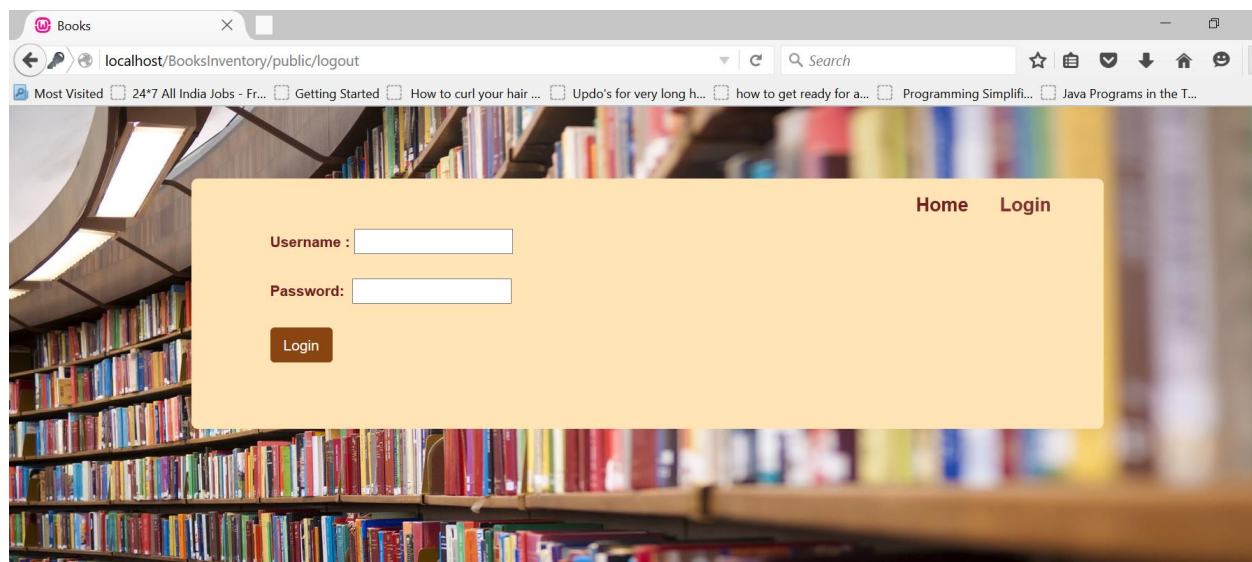
### Insights and Results:

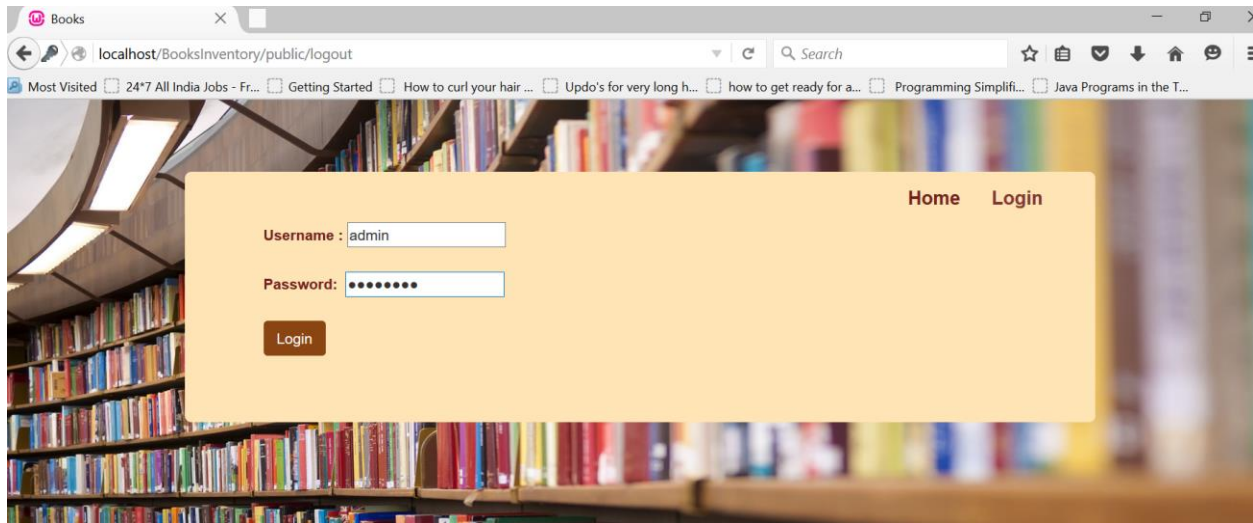
It was very challenging to understand Laravel and use the framework to design a project. Initial process started with installation of Laravel framework. Luckily there were no hurdles while installing. It went through easily. The actual struggle started when tried to create a new page using controller, model and views. Started facing lots of issues in Routing. Tried solving those and many other small issues like returning to views, lot of other exceptions but managed to complete the CRUD using Laravel which internally uses MVC framework. The main challenge started was while implementing login page using basic authentication. After lot of tries it finally worked well. So this project was completed using Laravel with CRUD operations along with Basic authentication.

Below is the screenshots for the Outputs of each operations:

### Login Page:

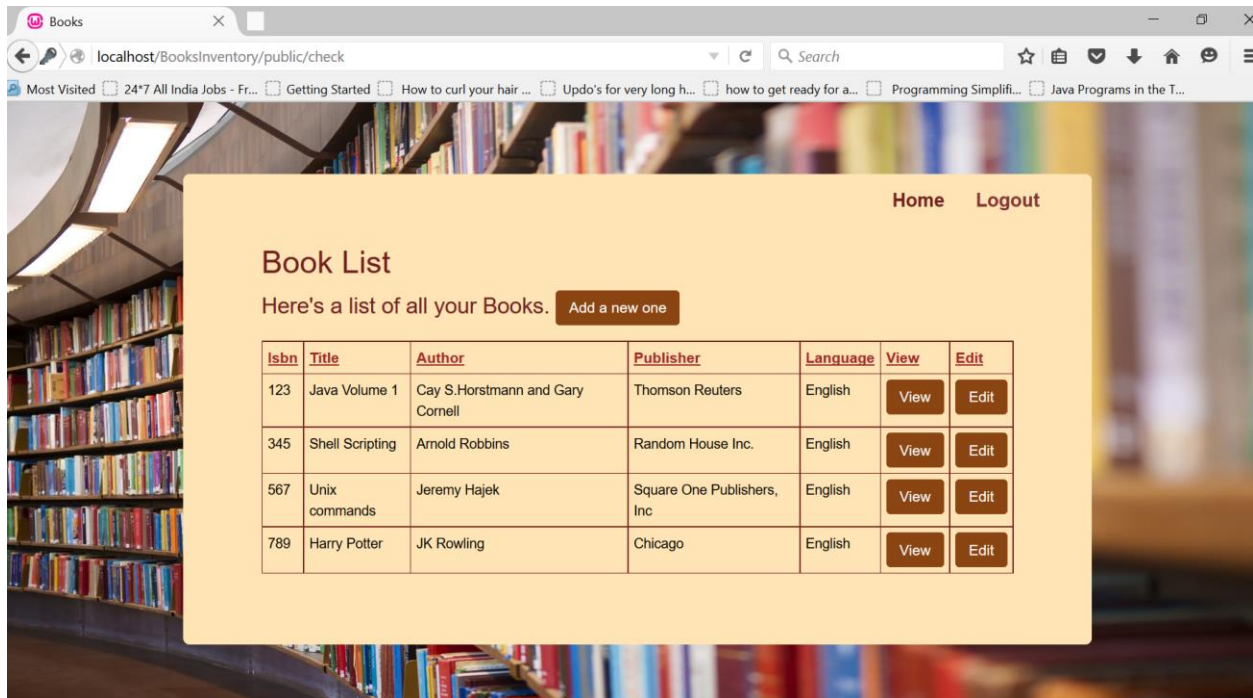
Url used: <http://localhost/BooksInventory/public/>





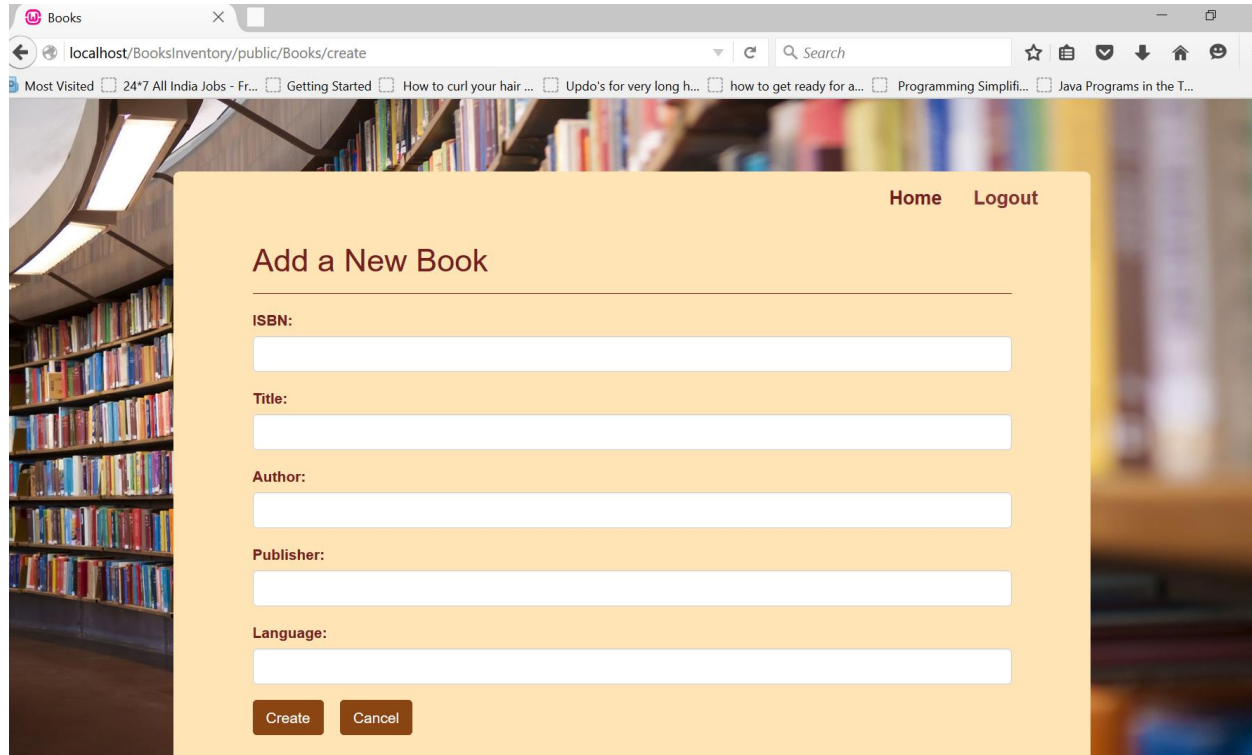
### Index/Show page:

Url used : <http://localhost/BooksInventory/public/Books>



Create page:

Url used : <http://localhost/BooksInventory/public/Books/create>



The screenshot shows a web browser window with the address bar displaying `localhost/BooksInventory/public/Books/create`. The page features a background image of a library with bookshelves. Overlaid on this is a light orange form titled "Add a New Book". In the top right corner of the form, there are links for "Home" and "Logout". The form contains five input fields, each preceded by a label: "ISBN:", "Title:", "Author:", "Publisher:", and "Language:". At the bottom of the form, there are two buttons: "Create" and "Cancel".

Books

localhost/BooksInventory/public/Books/create

Search

Most Visited 24\*7 All India Jobs - Fr... Getting Started How to curl your hair ... Updo's for very long h... how to get ready for a... Programming Simplifi... Java Programs in the T...

Home Logout

### Add a New Book

ISBN:

Title:

Author:

Publisher:

Language:

Create Cancel

Books

localhost/BooksInventory/public/Books/create

Home Logout

## Add a New Book

ISBN:

100

Title:

Php final project

Author:

Javeria

Publisher:

Laravel

Language:

Php and Html

Create Cancel

### View Page:

Url used: <http://localhost/BooksInventory/public/Books/15>

Books

localhost/BooksInventory/public/Books/20

Home Logout

## Edit Book

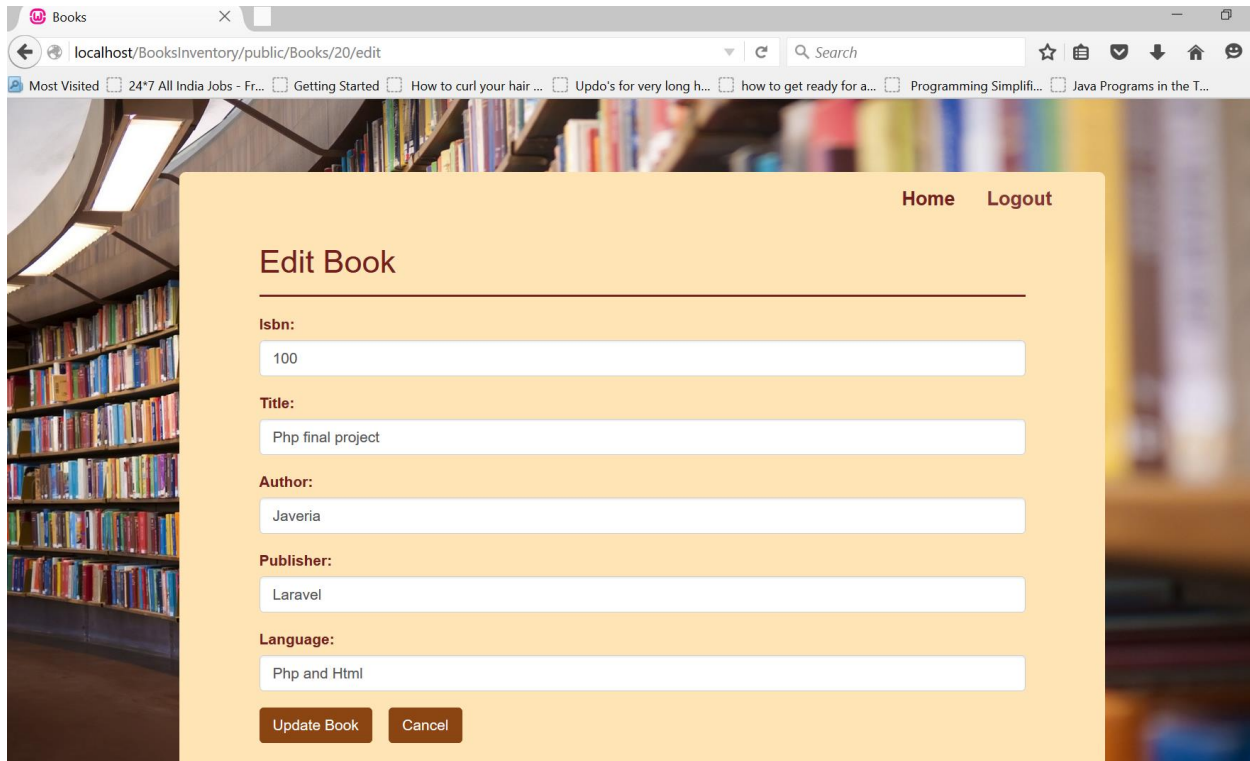
Isbn :	100
Title :	Php final project
Author :	Javeria
Publisher :	Laravel
Language :	Php and Html

Back to all Books Edit Book Delete Book



### Edit Page:

Url used: <http://localhost/BooksInventory/public/Books/15/edit>

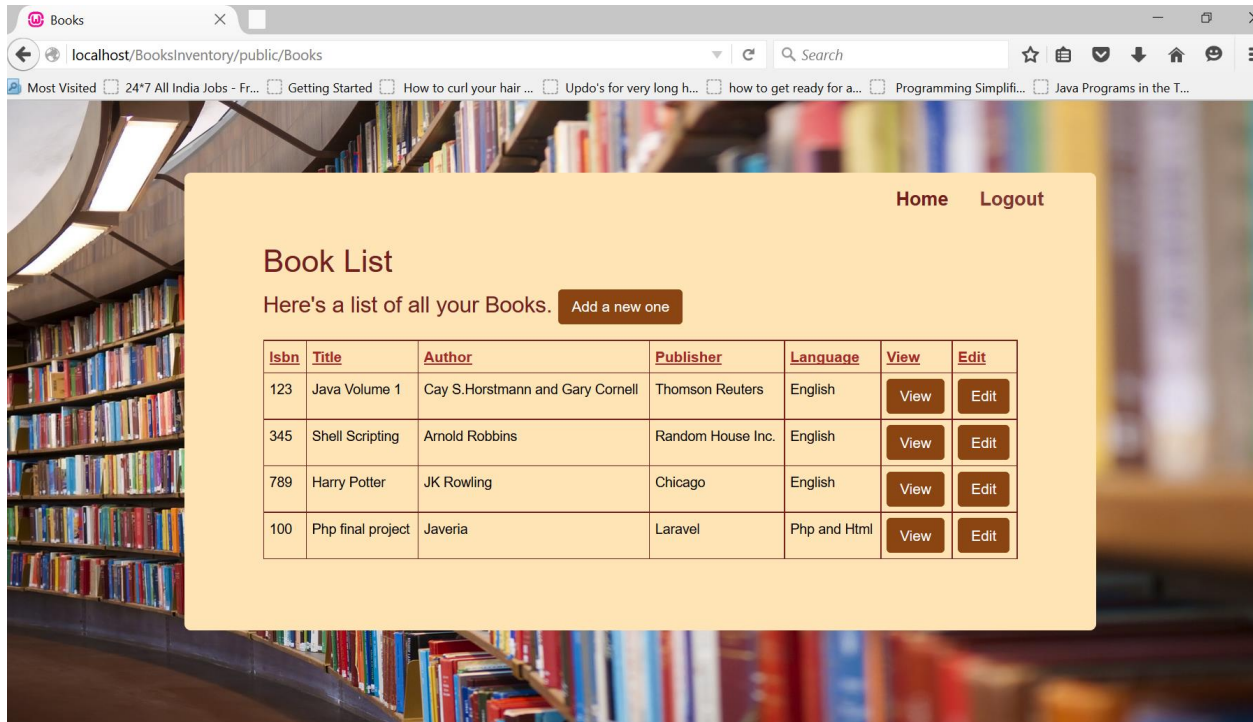


The screenshot shows a web browser window with the address bar displaying 'localhost/BooksInventory/public/Books/20/edit'. The browser's tab is labeled 'Books'. The background of the page is a blurred image of a library with bookshelves. Overlaid on this is a light orange modal form titled 'Edit Book'. In the top right corner of the modal, there are links for 'Home' and 'Logout'. The form contains five text input fields, each with a label to its left: 'Isbn:' with the value '100', 'Title:' with 'Php final project', 'Author:' with 'Javeria', 'Publisher:' with 'Laravel', and 'Language:' with 'Php and Html'. At the bottom of the form are two buttons: 'Update Book' and 'Cancel'.

### Delete page:

This on click of delete button deletes the record.

For demo I can show the existing record deleted in index page.



### Validation added to create and edit page.

