

# BOOKLAND

**Prepared by**  
Kshiti Patel (16IT080)  
Nidhi Patel (16IT085)

**Under the supervision of**  
Prof. Hemant Yadav

A Report Submitted to  
Charotar University of Science and Technology  
for Partial Fulfillment of the Requirements for the  
Degree of Bachelor of Technology  
in Information Technology

**IT345 Software Group Project-II (5<sup>th</sup> Sem)**

**Submitted at**



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**Chandubhai S. Patel Institute of Technology**

**At: Changa, Dist: Anand – 388421**

**November 2018**

**CERTIFICATE**

This is to certify that the report entitled “**BOOKLAND**” is a bonafied work carried out by **Ms. Kshiti Patel (16IT080)**, **Ms. Nidhi Patel (16IT085)** under the guidance and supervision of **Prof. Hemant Yadav** for the subject **Software Group Project-II(IT345)** of 5<sup>th</sup> Semester of Bachelor of Technology in **Information Technology** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate herself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Under supervision of,

Prof. Hemant Yadav  
Assistant Professor  
Dept. of Information Technology  
CSPIT, Changa, Gujarat.

Prof. Parth Shah  
Head & Associate Professor  
Department of Information Technology  
CSPIT, Changa, Gujarat.

---

---

**Chandubhai S Patel Institute of Technology**

At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat

**TABLE OF CONTENTS**

• <b>Acknowledgement .....</b>	<b>1</b>
• <b>Abstract.....</b>	<b>2</b>
• <b>Chapter 1 Introduction.....</b>	<b>3</b>
1.1 Project Overview .....	3
1.1.1 How to do .....	3
1.2 Scope .....	3
1.3 Objective .....	3
• <b>Chapter 2 System Analysis.....</b>	<b>4</b>
2.1 Tools & Technology (Hardware/ Software Requirements- Minimum requirements to run the system) .....	4
• <b>Chapter 3 System Design.....</b>	<b>5</b>
3.1 Flow of System .....	5
• <b>Chapter 4 Implementation .....</b>	<b>6</b>
4.1 Laravel at a glance.....	6
4.2 Implementation of Laravel in project .....	13
4.3 Coding Standards(Sample code of main module) .....	13
4.4 Snapshots of project .....	18
• <b>Chapter 5 Constraints and Future Enhancement.....</b>	<b>28</b>
• <b>Chapter 6 Conclusion .....</b>	<b>29</b>
• <b>References.....</b>	<b>30</b>

## Acknowledgement

Knowledge in itself is a continuous process. At this moment of our substantial enhancement, we rarely find words to express our gratitude towards those who were constantly involved with us. We take this opportunity to express our profound gratitude and deep regards to our guide Mr. Hemant Yadav for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry us a long way in the journey of life on which we am about to embark. The completion of any inter disciplinary project depends upon coordination, cooperation and combined efforts of several resources of knowledge, creativity, skill, energy and time. The work being accomplished now, we feel our most sincere urge to recall and knowledge through these lines, trying our best to give full credit wherever it deserves. It's our good fortune that we had support and well wishes of many. We are thankful to all and those names which have been forgotten to acknowledge here but contributions have not gone unnoticed.

**“We may not achieve everything we dream, but we cannot achieve anything unless we dream.”**

With Sincere regards,  
Kshiti Patel-16IT080  
Nidhi Patel-16IT085

## Abstract

“**Bookland**” is a PHP application ease the process of buying and selling books. This application software is user friendly and has required options which can be utilized by the user to perform the desired options. The software is developed using HTML, CSS and JavaScript as front end and MySQL as back end in windows environment.

“**Bookland**” is a content management project. It is made using PHP framework named Laravel. It is a platform where the we have facility to add a subject, add a book. Admin can create a user, change his credentials, contact the seller and many such functionalities. Subjects and books can be added to trash (soft deleted) according to the unavailability.

# CHAPTER 1

## INTRODUCTION

### 1.1 Project Overview

Bookland is a content management project. It is made using PHP framework named Laravel. It is a platform where we have facility to add a subject, add a book. Admin can create a user, change his credentials, contact the seller and many such functionalities. Subjects and books can be added to trash (soft deleted) according to the unavailability.

#### 1.1.1 How to do

The project lies on the foundation of MVC architecture. Models, controllers, migrations, routes work together to maintain the flow of the project.

**Model:** It is responsible for tables within the database. Every module has its model.

**View:** It separates the application logic and the presentation logic. **Views** are stored in resources/views directory.

**Controller:** It acts as a directing traffic between Views and Models.

**Migration:** These are the simple classes that create the tables in database.

**Middleware:** Middleware acts as a bridge between a request and a response. It is a type of filtering mechanism.

**Routes:** It defines routes that are for your web interface

### 1.2 Scope

#### What it can do

- Provides admin and user login
- Admin has the authority to make any user as admin as well as remove one.
- Admin and user both can contact the seller.
- Admin and user both can add subjects and books.
- Books can be trashed and can be restored.
- The user can edit their profile.

#### What it can't do

- The seller cannot see their past activities.
- Spam records cannot be separated from real records, so cannot protect buyers from fake.
- Buyers cannot view the appearance and condition of the book since we didn't provide the upload picture functionality.

### 1.3 Objective

There are platforms for buying and selling of books, but the problem with those sites is we cannot have those books in short time and the delivery charges are also applied. So, we are developing this application for the ease of students of IT department so they can exchange their books easily.

## CHAPTER 2

# SYSTEM ANALYSIS

### 2.1 Tools and Technology

#### Hardware requirements:

Processor	Minimum 2.20 GHz
RAM	Minimum 2GB
Hard disk	20 GB free space

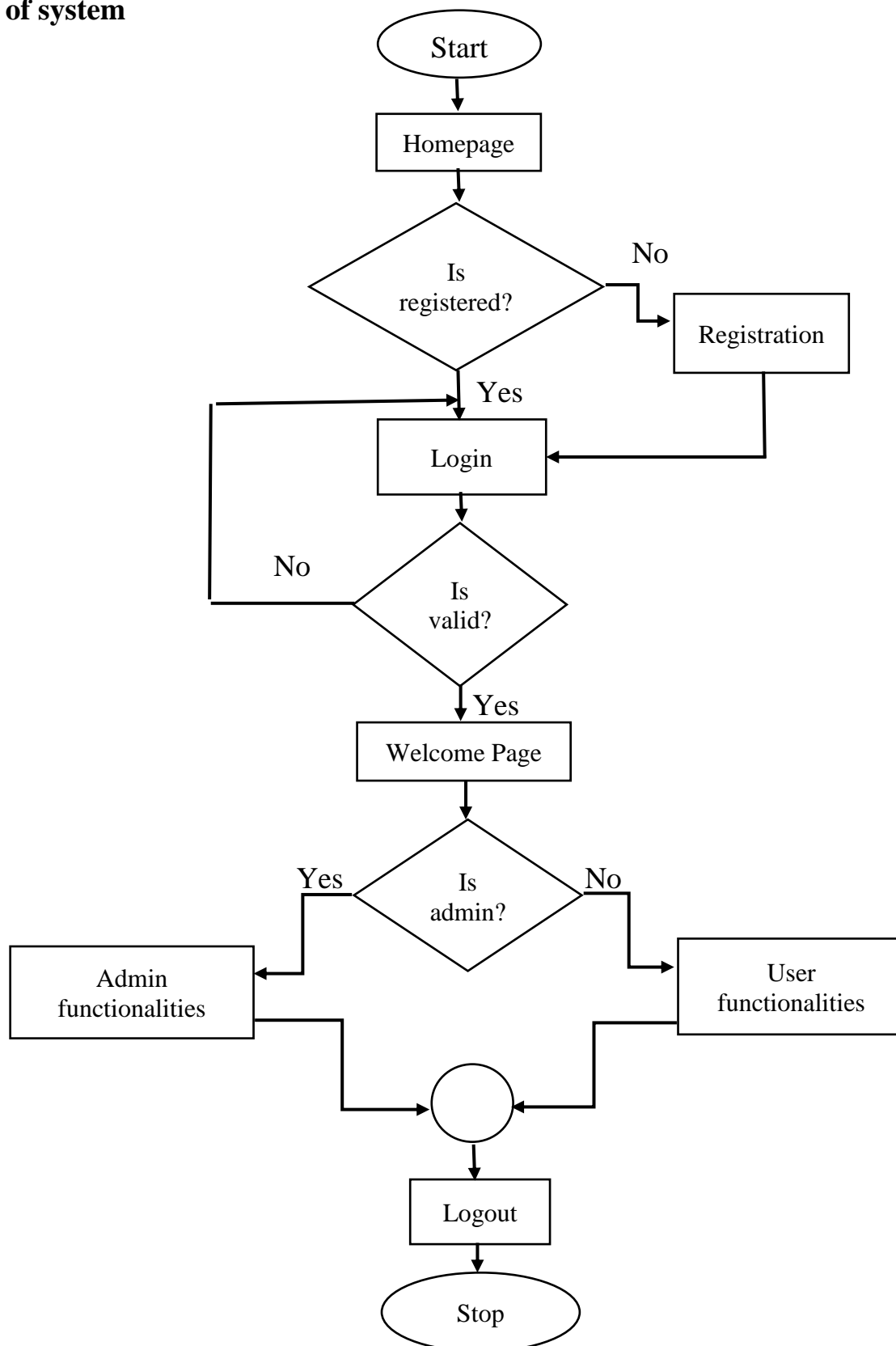
#### Software requirements:

Laragon version-3.4.0
Sublime/ visual studio code

## CHAPTER 3

### SYSTEM DESIGN

#### 3.1 Flow of system





## CHAPTER 4

# IMPLEMENTATION

### Laravel at glance

#### What is Laravel?

Laravel is a PHP-based web framework that is largely based on the MVC architecture. Laravel was created to make it easier for developers to get started on PHP projects. With Laravel, you think less about the setup, architecture, and dependencies of a project and go straight into the meat of the project.

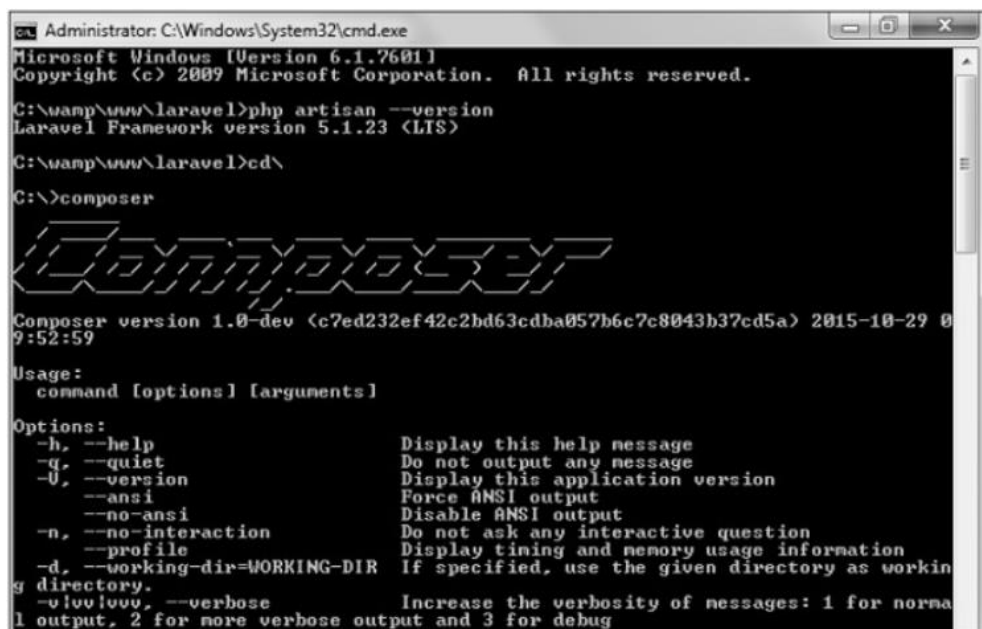
#### Basic Installation steps

For managing dependencies, Laravel uses **composer**. Make sure you have a Composer installed on your system before you install Laravel.

**Step 1** – Visit the following URL and download composer to install it on your system.

<https://getcomposer.org/download/>

**Step 2** – After the Composer is installed, check the installation by typing the Composer command in the command prompt as shown in the following screenshot.



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\uamp\www\laravel>php artisan --version
Laravel Framework version 5.1.23 (LTS)

C:\uamp\www\laravel>cd\

C:\>composer

Composer version 1.0-dev (c7ed232ef42c2bd63cd8a057b6c7c8043b37cd5a) 2015-10-29 09:52:59

Usage:
  command [options] [arguments]

Options:
  -h, --help                Display this help message
  -q, --quiet               Do not output any message
  -U, --version              Display this application version
  --ansi                    Force ANSI output
  --no-ansi                 Disable ANSI output
  -n, --no-interaction      Do not ask any interactive question
  --profile                 Display timing and memory usage information
  -d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.
  --verbose                 Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug
```

Figure 4.1

**Step 3** – Create a new directory anywhere in your system for your new Laravel project. After that, move to path where you have created the new directory and type the following command there to install Laravel.

**composer create-project laravel/laravel --prefer-dist**

**Step 4** – The above command will install Laravel in the current directory. Start the Laravel service by executing the following command.

**php artisan serve**

**Step 5** – After executing the above command, you will see a screen as shown below –

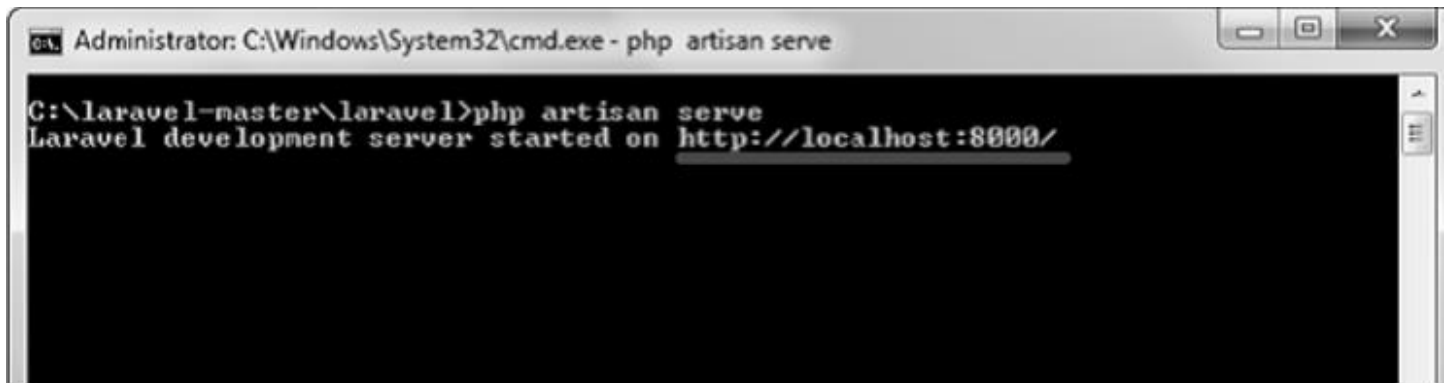


Figure 4.2

**Step 6** – Copy the URL underlined in grey in the above screenshot and open that URL in the browser. If you see the following screen, it implies Laravel has been installed successfully.



Figure 4.3

## Laravel with Laragon Installation steps

1. Navigate to the Laragon site. This is where you will find and download the installer for Laragon.
2. Run Laragon.
3. Press Start All.
4. Press the Settings cog and navigate to Services and Ports and check PHP Server . ...
5. Go to Menu > Laravel > Create Project > Laravel5 and give your project a name.
6. Wait for a terminal window to open up.

## Features of PHP Laravel Framework

### 1. Template Engine:

Laravel framework is highly acknowledged for its built-in lightweight templates which can be used to create wonderful layouts using dynamic content seeding. In addition to this, it provides multiple widgets incorporating CSS and JS code with robust structures

### 2. MVC Architecture Support:

Laravel supports MVC architecture pattern which ensures separates business logic and presentation layers. MVC pattern of Laravel has a lot of built-in functions, improves application performance and increases security as well as scalability.

### 3. Eloquent ORM (Object Relational Mapping):

Laravel offers Eloquent Object Relational Mapping (ORM) which includes a simple PHP Active Record implementation. This allows web application developers to write database queries with PHP syntax rather than writing SQL code. An ORM is relatively faster than other PHP frameworks.

### 4. Security:

Laravel framework offers very strong web application security. It uses hashed and salted password mechanism so the password would never be saved as plain text in the database. It also uses “Bcrypt Hashing Algorithm” for generating an encrypted password. Additionally, this PHP web development framework uses prepared SQL statements that prevents SQL injection attacks.

### 5. Artisan:

Laravel framework offers a built-in command line tool called Artisan which helps automating the majority of tedious repetitive programming tasks. These artisans can also be utilized to create the database structure, a skeleton code, and manage migration so it is pretty easy-to-manage database system. In addition, it can generate basic MVC files through command-line and manage those assets as well as their respective configurations. Artisan even helps developers to create their own commands and use them as required.

### 6. Libraries & Modular:

Laravel comes with pre-installed Object-Oriented and Modular libraries which are not available in many other PHP frameworks. For example, an Authentication library which is easy-to-implement and has features such as checking active users, Bcrypt hashing, password reset, CSRF (Cross-site Request Forgery) protection, and encryption. Furthermore, this framework is divided into individual modules adopting modern PHP principles facilitating responsive and modular web applications development.

### 7. Database Migration System:

Laravel migration system helps to expand the web application database structure without re-creating everytime when there is a change in code. Because of this feature, the risk of losing data is very minimal. It not only offers the facility to change the database structure but also helps to do using PHP code instead of SQL. Laravel Schema Builder helps creating database tables and inserts indices or columns promptly.

### 8. Unit-Testing:

Laravel is a preferred framework for web application developers today due to the way it facilitates unit testing. This framework is capable of running multiple unit tests to ensure that new changes done by the developer do not unexpectedly break the web application. Ideally, Laravel based web applications are capable of stable releases as it is careful of the known failures. It also make it easy for the developers to write unit-tests.

### M-V-C

MVC is an acronym for 'Model View Controller'. It represents architecture developers adopt when building applications. With the MVC architecture, we look at the application structure with regards to how the data flow of our application works.

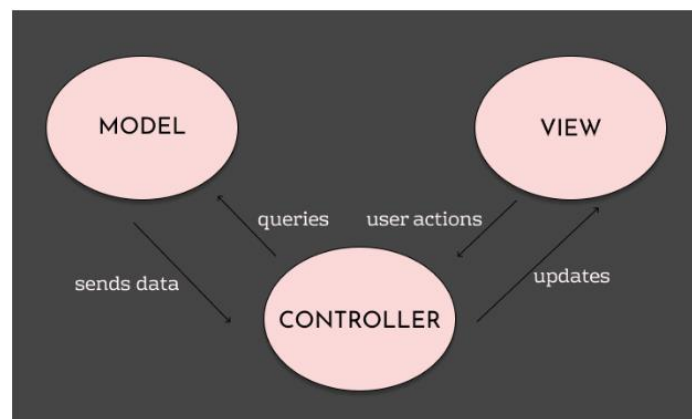


Figure 4.4

A Model is a representation of a real-life instance or object in our code base. The View represents the interface through which the user interacts with our application. When a user takes an action, the Controller handles the action and updates the Model if necessary.

## How Laravel request works

When you create a new Laravel project, (you can create one by running the command `laravel new project-name`), the project has the following structure:

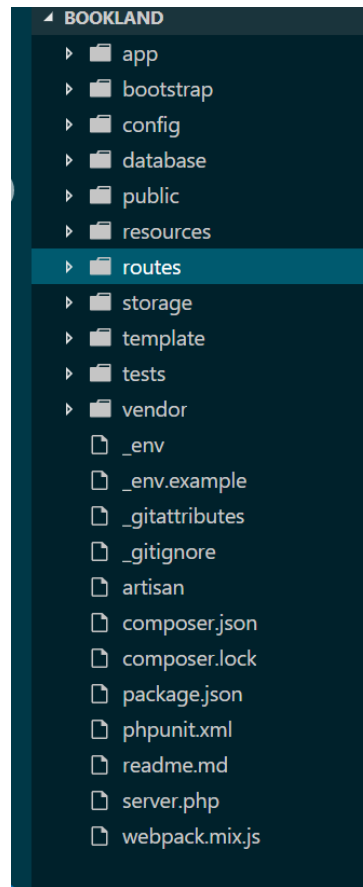


Figure 4.5

There is a file in the `routes/` directory called `web.php`. That file is where you handle the requests when users visit your app. In this file, you can route URLs to controllers in your application. The file looks like this:

```
web.php x
1  <?php
2
3  /*
4  |-----
5  | Web Routes
6  |-----
7  |
8  | Here is where you can register web routes for your application. These
9  | routes are loaded by the RouteServiceProvider within a group which
10 | contains the "web" middleware group. Now create something great!
11 |
12 */
13
14 use App\Subject;
15 Route::get('test',function(){
16 |     return App\User::find(1)->profile;
17 | });
```

Figure 4.6

## MVC implementation in Laravel applications

To create a new project

```
$ laravel new project-name
```

To see the starter app at work, update the APP\_URL in your .env file to http://localhost:8000 then run the following command in your terminal:

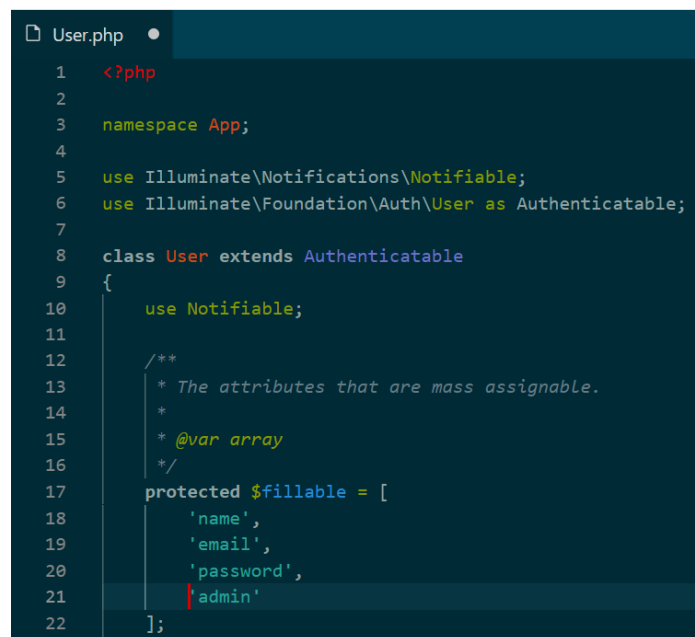
```
$ php artisan serve
```

### Model

In Laravel, the Model is usually a class with properties that match the columns in the database.

To create a model in Laravel, run the command in your terminal:

```
$ php artisan make:model Product
```



```
User.php
1  <?php
2
3  namespace App;
4
5  use Illuminate\Notifications\Notifiable;
6  use Illuminate\Foundation\Auth\User as Authenticatable;
7
8  class User extends Authenticatable
9  {
10     use Notifiable;
11
12     /**
13      * The attributes that are mass assignable.
14      *
15      * @var array
16      */
17     protected $fillable = [
18         'name',
19         'email',
20         'password',
21         'admin'
22     ];
```

Figure 4.7

### Migration

Migration lets developers make and undo changes to a project's database. Migrations can be used to make managing databases easy and predictable. To create a migration, run the following in your terminal:

```
$ php artisan make:migration table_name_table
```

```
2018_10_12_062054_create_subjects_table.php x
1  <?php
2
3  use Illuminate\Support\Facades\Schema;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Database\Migrations\Migration;
6
7  class CreateSubjectsTable extends Migration
8  {
9      /**
10       * Run the migrations.
11       *
12       * @return void
13       */
14     public function up()
15     {
16         Schema::create('subjects', function (Blueprint $table) {
17             $table->increments('id');
18             $table->string('subject_name');
19             $table->timestamps();
20         });
21     }
22 }
```

Figure 4.8

## Controller

Controllers are responsible for completing user actions and the managing the business logic of our applications.

To create a resource controller in Laravel, run the following command:

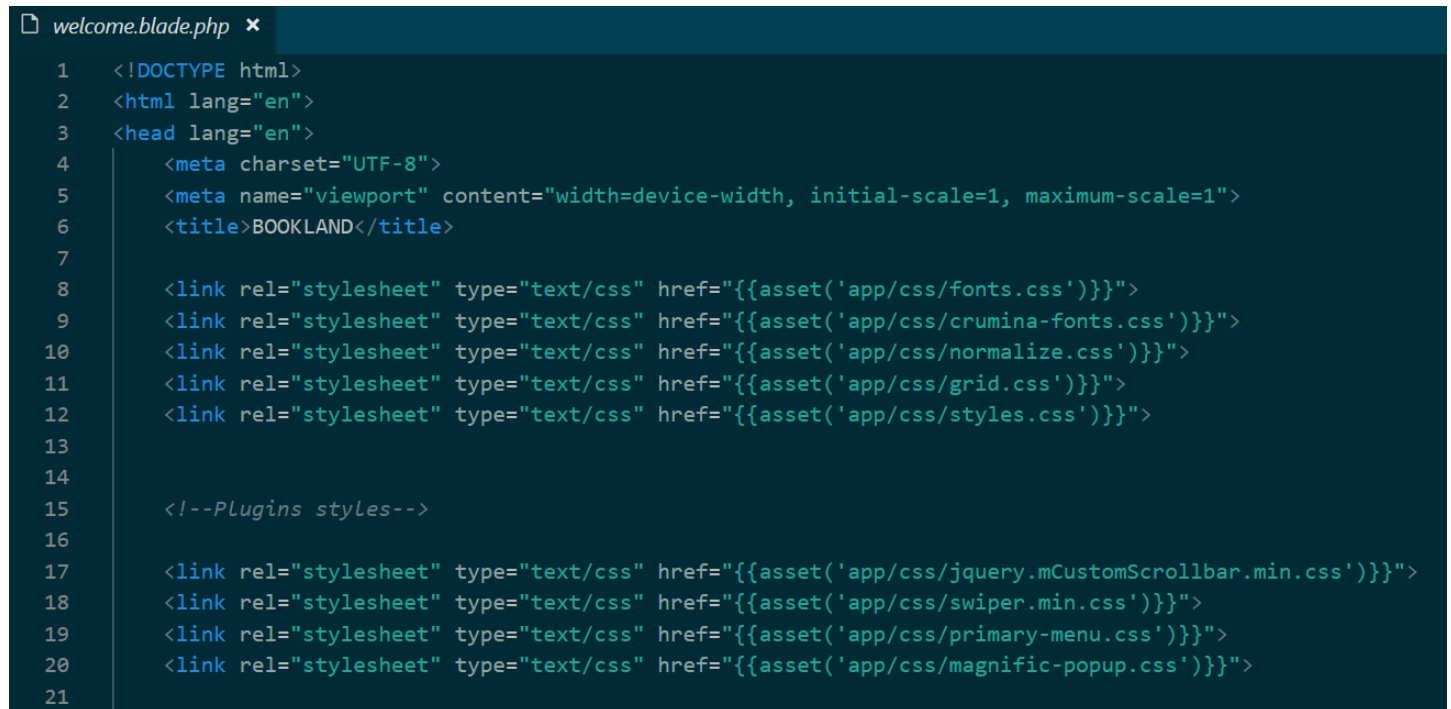
```
$ php artisan make:controller ProductController -r
```

```
UsersController.php x
11 class UsersController extends Controller
12 {
13     public function __construct(){
14         $this->middleware('admin');
15     }
16     /**
17      * Display a listing of the resource.
18      *
19      * @return \Illuminate\Http\Response
20      */
21     public function index()
22     {
23         return view('admin.users.index')->with('user');
24     }
25
26     /**
27      * Show the form for creating a new resource.
28      *
29      * @return \Illuminate\Http\Response
30      */
31     public function create()
32     {
33         return view('admin.users.create');
34     }
35
36     /**
37      * Store a newly created resource in storage.
38      *
39      * @param \Illuminate\Http\Request $request
```

Figure 4.9

## Views

Views usually store the HTML of your page and are the presentation layer of the MVC architecture.



```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head lang="en">
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1">
6     <title>BOOKLAND</title>
7
8     <link rel="stylesheet" type="text/css" href="{{asset('app/css/fonts.css')}}">
9     <link rel="stylesheet" type="text/css" href="{{asset('app/css/crumina-fonts.css')}}">
10    <link rel="stylesheet" type="text/css" href="{{asset('app/css/normalize.css')}}">
11    <link rel="stylesheet" type="text/css" href="{{asset('app/css/grid.css')}}">
12    <link rel="stylesheet" type="text/css" href="{{asset('app/css/styles.css')}}">
13
14
15    <!--Plugins styles-->
16
17    <link rel="stylesheet" type="text/css" href="{{asset('app/css/jquery.mCustomScrollbar.min.css')}}">
18    <link rel="stylesheet" type="text/css" href="{{asset('app/css/swiper.min.css')}}">
19    <link rel="stylesheet" type="text/css" href="{{asset('app/css/primary-menu.css')}}">
20    <link rel="stylesheet" type="text/css" href="{{asset('app/css/magnific-popup.css')}}">
21
  
```

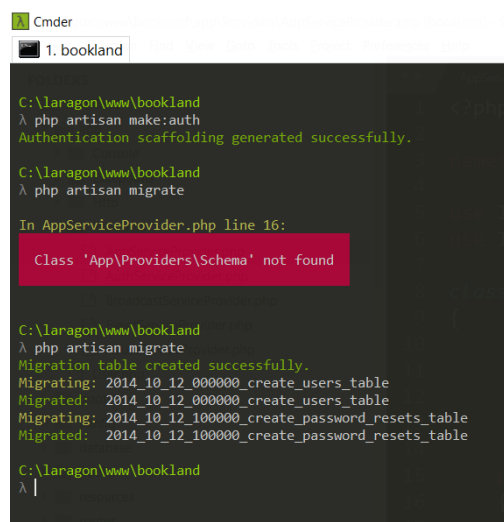
Figure 4.10

## Implementation of Laravel in project

Following are the different unique features used in our project:

### 1. Authentication system

Laravel makes implementing authentication very simple. In fact, almost everything is configured for you out of the box. The authentication configuration file is located at config/auth.php which contains several well documented options for tweaking the behavior of the authentication services.



```

C:\laragon\www\bookland
λ php artisan make:auth
Authentication scaffolding generated successfully.

C:\laragon\www\bookland
λ php artisan migrate

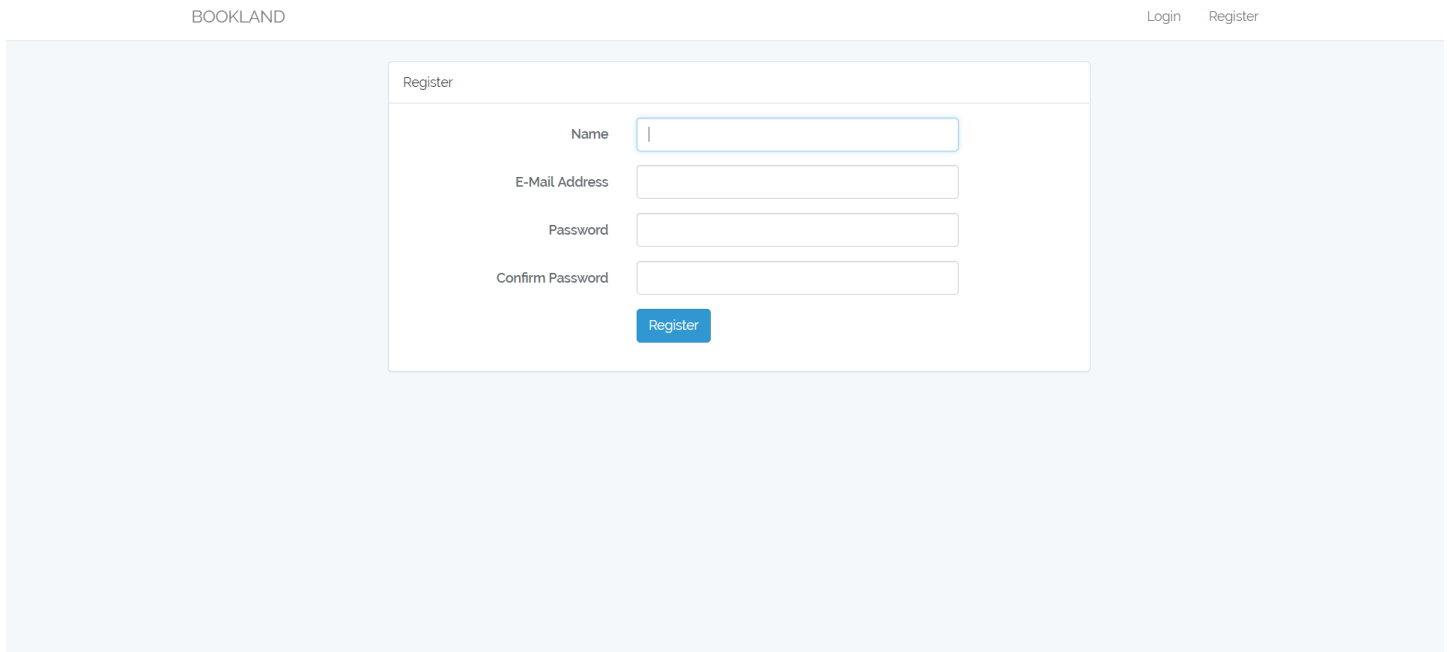
In AppServiceProvider.php line 16:
Class 'App\Providers\Schema' not found

C:\laragon\www\bookland
λ php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table

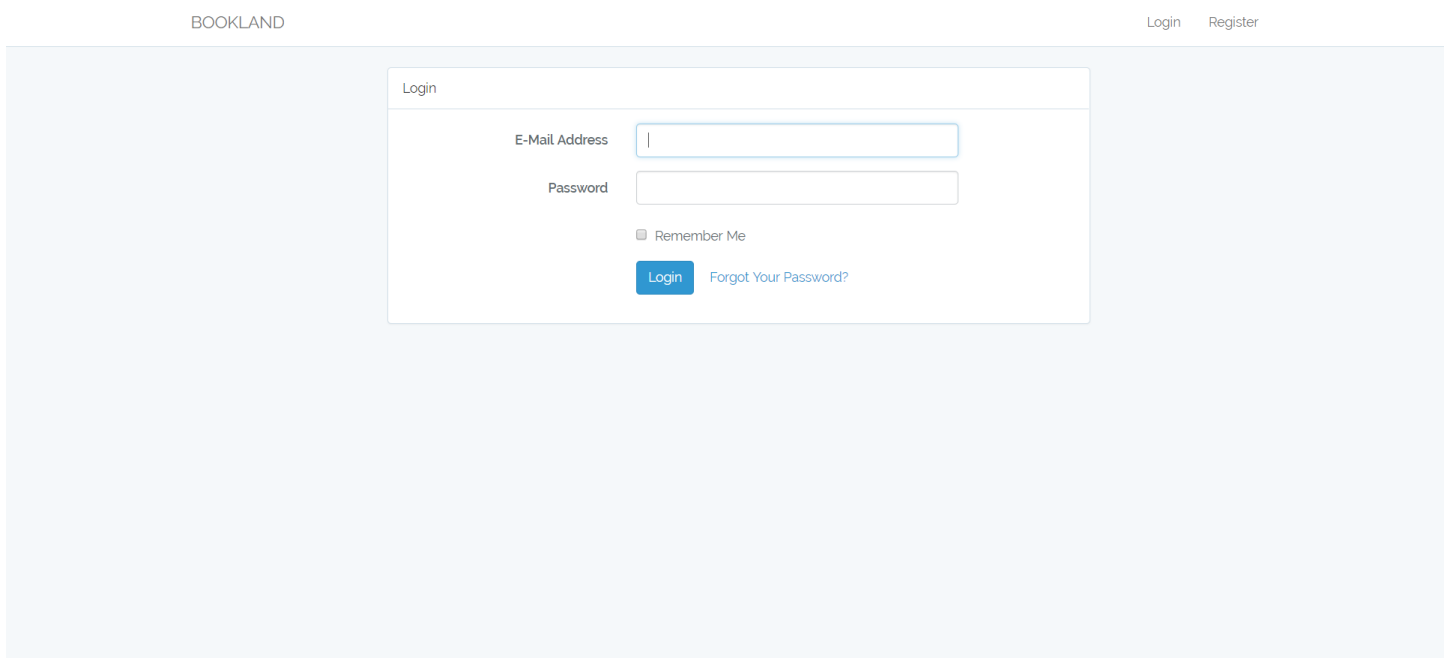
C:\laragon\www\bookland
λ
  
```

Figure 4.11





The screenshot shows the 'Register' form on the Bookland website. The form is titled 'Register' and is located in the center of the page. It contains four input fields: 'Name', 'E-Mail Address', 'Password', and 'Confirm Password'. Below these fields is a blue 'Register' button. The background is a light blue gradient. In the top left corner, the text 'BOOKLAND' is visible, and in the top right corner, the links 'Login' and 'Register' are present.

*Figure 4.12*

The screenshot shows the 'Login' form on the Bookland website. The form is titled 'Login' and is located in the center of the page. It contains two input fields: 'E-Mail Address' and 'Password'. Below these fields is a checkbox labeled 'Remember Me' and a blue 'Login' button. To the right of the 'Login' button is a link that says 'Forgot Your Password?'. The background is a light blue gradient. In the top left corner, the text 'BOOKLAND' is visible, and in the top right corner, the links 'Login' and 'Register' are present.

*Figure 4.13*

## Middleware

Middleware acts as a bridge between a request and a response. It is a type of filtering mechanism. This chapter explains you the middleware mechanism in Laravel. Laravel includes a middleware that verifies whether the user of the application is authenticated or not.

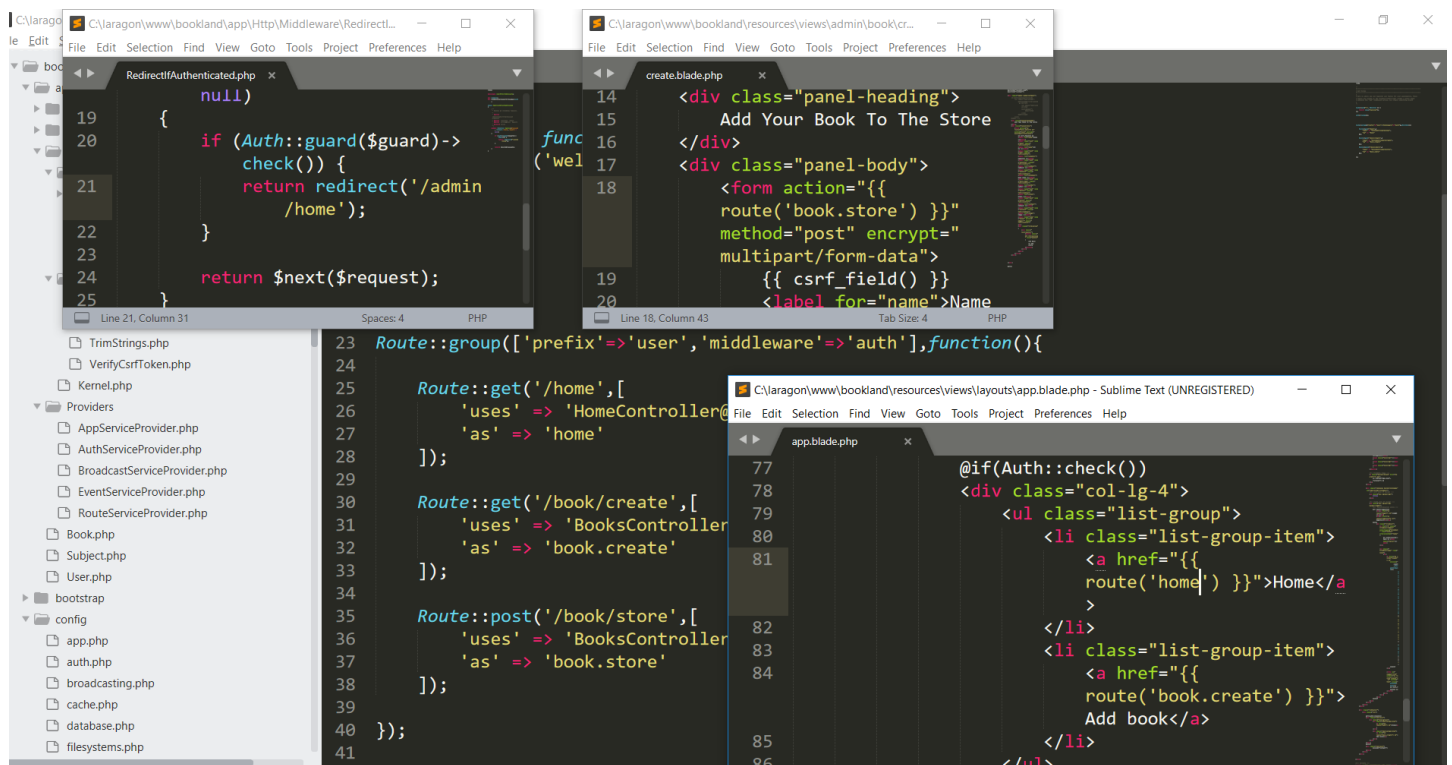


Figure 4.14

### Toastr notifications

Toastr is a Javascript library for Gnome / Growl type non-blocking notifications. jQuery is required. The goal is to create a simple core library that can be customized and extended.



Figure 4.15

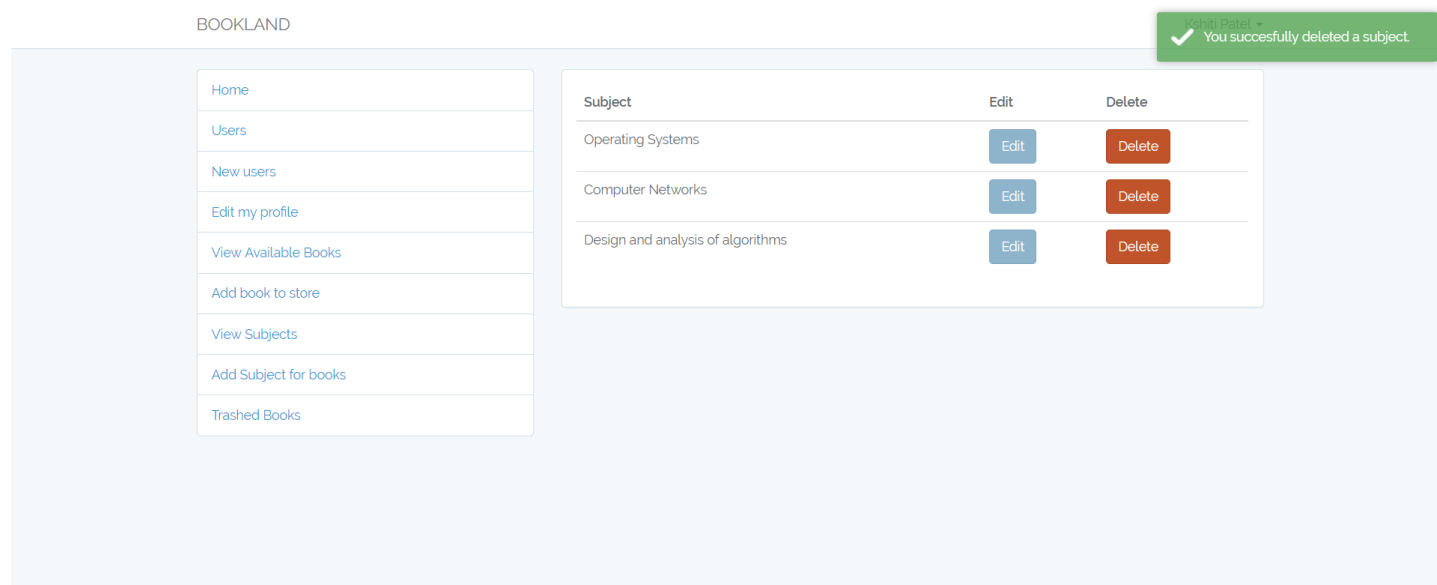


Figure 4.16

### Soft deletes

Laravel is shipped with Eloquent and supported Soft Delete feature. In addition to actually removing records from your database, Eloquent can also “soft delete” models. When models are soft deleted, they are not actually removed from your database. Instead, a `deleted_at` attribute is set on the model and inserted into the database.

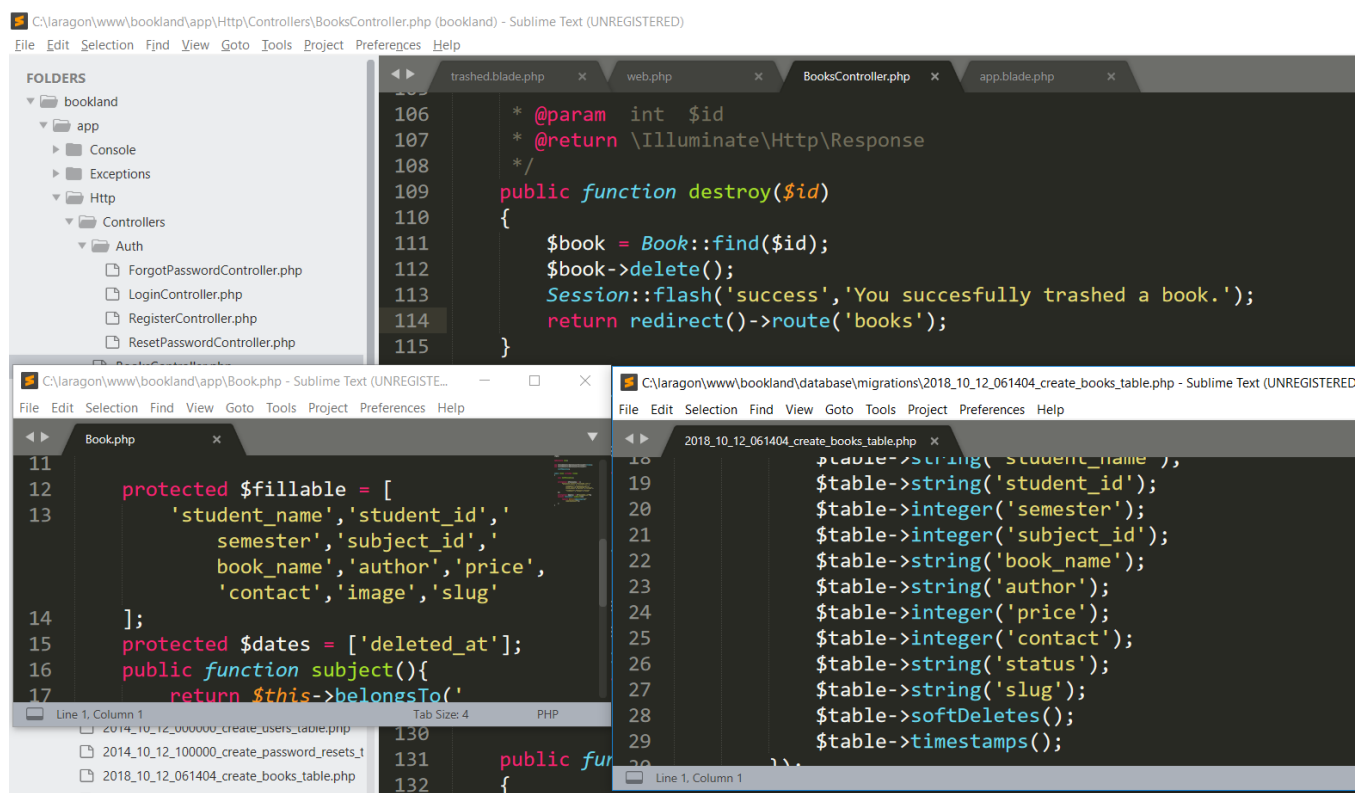


Figure 4.17

## Seeder class

Laravel includes a simple method of seeding your database with test data using seed classes. Seed classes may have any name you wish, but probably should follow some sensible convention, such as UsersTableSeeder, etc. By default, a DatabaseSeeder class is defined for you.

```

C:\laragon\www\bookland
Migrated: 2018_10_12_062054_create_subjects_table
C:\laragon\www\bookland
A php artisan make:seeder UsersTableSeeder
Seeder created successfully.
C:\laragon\www\bookland
A php artisan migrate:refresh
Rolling back: 2018_10_12_062054_create_subjects_table
Rolling back: 2018_10_12_062054_create_subjects_table
Rolling back: 2018_10_12_061404_create_books_table
Rolling back: 2018_10_12_061404_create_books_table
Rolling back: 2014_10_12_100000_create_password_resets_table
Rolling back: 2014_10_12_100000_create_password_resets_table
Rolling back: 2014_10_12_000000_create_users_table
Rolling back: 2014_10_12_000000_create_users_table
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table
Migrating: 2018_10_12_061404_create_books_table
Migrated: 2018_10_12_061404_create_books_table
Migrating: 2018_10_12_062054_create_subjects_table
Migrated: 2018_10_12_062054_create_subjects_table
C:\laragon\www\bookland
A php artisan db:seed
Seeding: UsersTableSeeder
C:\laragon\www\bookland
A php artisan migrate:refresh
Rolling back: 2018_10_12_062054_create_subjects_table
Rolling back: 2018_10_12_062054_create_subjects_table
Rolling back: 2018_10_12_061404_create_books_table
Rolling back: 2018_10_12_061404_create_books_table
Rolling back: 2014_10_12_100000_create_password_resets_table
Rolling back: 2014_10_12_100000_create_password_resets_table
Rolling back: 2014_10_12_000000_create_users_table
Rolling back: 2014_10_12_000000_create_users_table
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table
Migrating: 2018_10_12_061404_create_books_table
Migrated: 2018_10_12_061404_create_books_table
Migrating: 2018_10_12_062054_create_subjects_table
Migrated: 2018_10_12_062054_create_subjects_table

DatabaseSeeder.php
10  * @return void
11  */
12  public function run()
13  {
14      $this->call(UsersTableSeeder::class);
15  }
16  }

UsersTableSeeder.php
5  class UsersTableSeeder extends Seeder
6  {
7      /**
8       * Run the database seeds.
9       *
10      * @return void
11      */
12  public function run()
13  {
14      App\User::create([
15          'name' => 'Kshiti Patel',
16          'email' => 'kshiti Patel1999@gmail.com',
17          'password' => bcrypt('password')
18      ]);
19  }
20  }

```

Figure 4.18

## Slugs

Creating Title Slugs with Laravel. When storing posts in the database a typical pattern for retrieving it is to create a unique “slug” based on the title. This gives users SEO-friendly way to access it through the URL.

## User permissions and Restrictions

There are some functionalities in the project where a particular user can be made admin or restricted from doing so. For doing so the user controller needs to be disabled. Glance of such procedure is provided in the following snippet.

```

C:\laragon\www\bookland\app\Http\Controllers\UserController.php
10 class UsersController extends Controller
11 {
12     public function __construct()
13     {
14         $this->middleware('admin');
15     }
16     /**
17      * Display a listing of the resource.
18      */
19 }

C:\laragon\www\bookland\app\Http\Kernel.php
51 protected $routeMiddleware = [
52     'auth' => \Illuminate\Auth\Middleware\
53         Authenticate::class,
54     'auth.basic' => \Illuminate\Auth\Middleware\
55         AuthenticateWithBasicAuth::class,
56     'bindings' => \Illuminate\Routing\Middleware\
57         SubstituteBindings::class,
58     'can' => \Illuminate\Auth\Middleware\Authorize
59         ::class,
60     'guest' => \App\Http\Middleware\
61         RedirectIfAuthenticated::class,
62     'throttle' => \Illuminate\Routing\Middleware\
63         ThrottleRequests::class,
64     'admin' => \App\Http\Middleware\Admin::class
65 ];

```

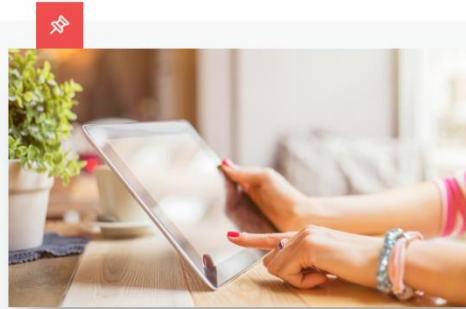
Figure 4.1

## Project snippets

BOOKLAND

[Home](#) [About Us](#)

LOGIN



REGISTER

Figure 4.20 – Homepage

BOOKLAND

[Home](#) [About Us](#)

BOOKLAND

Invest less gain more knowledge

Books play a significant role in shaping a person's mentality and developing ethics. They show us direction and give us an objective to accomplish our dreams and do something great for the society, family, and ultimately ourselves. The interest for books depends largely on its price. There can be a very fascinating book that we had longed to read, but when we see its price tag at the book store, all our dreams are shattered. This happens to each one of us at-least once in a lifetime. Expensive books squeeze our interest and pull us apart from the basic ideology of surging our level of knowledge. This is when the role of second hand textbooks come into picture. Majority of times, they are available at less than half the price of a new text book. There are countless book-stores that offer used textbooks to students who can or cannot afford to buy a new one. It is not only the monetary benefits

Figure 4.21- About

BOOKLAND Login Register

Login

E-Mail Address

Password

☐ Remember Me

Login

[Forgot Your Password?](#)

Figure 4.22- Login

BOOKLAND Login Register

Register

Name

E-Mail Address

Password

Confirm Password

Register

Figure 4.23- Register

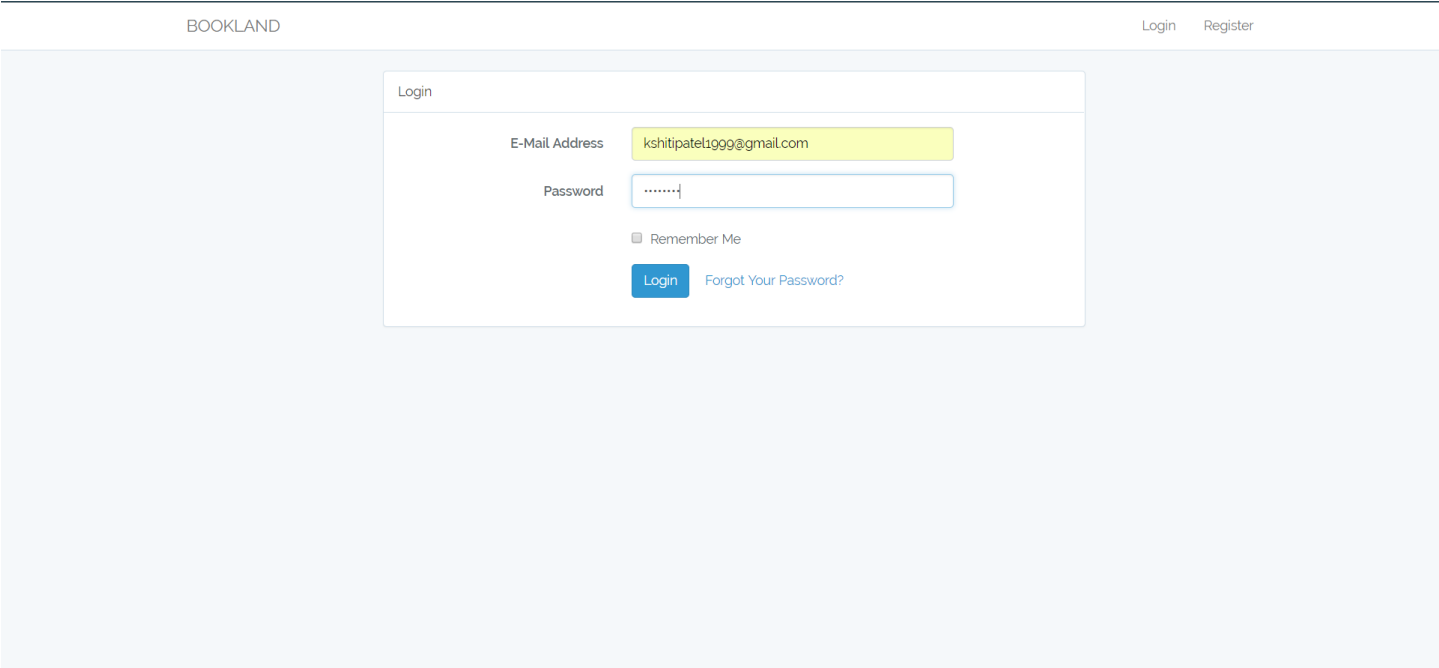


Figure 4.24- Admin Login

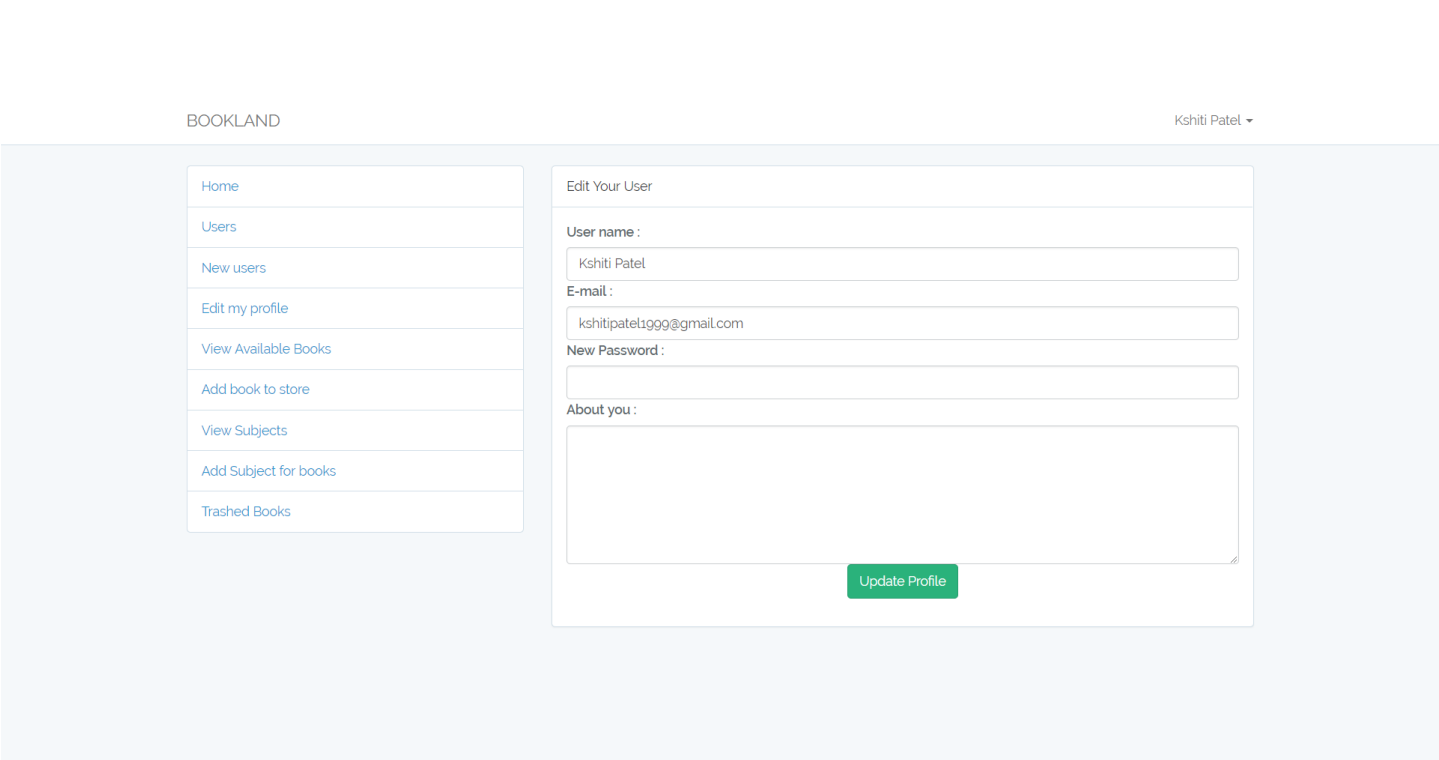


Figure 4.25- Edit Profile

BOOKLAND

Kshiti Patel ▾

Home

Users

New users

Edit my profile

View Available Books

Add book to store

View Subjects

Add Subject for books

Trashed Books

Add Subject

Subject :

Add Subject

Figure 4.26- Add Subject

BOOKLAND

Kshiti Patel ▾

Home

Users

New users

Edit my profile

View Available Books

Add book to store

View Subjects

Add Subject for books

Trashed Books

Add Your Book To The Store

Name :

ID no :

Semester :

Subject :

Operating Systems ▾

Book Name :

Author :

Price :

Contact :

Figure 4.27- Add Book to store



BOOKLAND Kshiti Patel ▾

Home	Contact Me
Users	Seller's Name : Kshiti
New users	Student ID : 16it085
Edit my profile	Semester : 2
View Available Books	Contact no. : 7227888244
Add book to store	<a href="#">Book my Book</a>
View Subjects	
Add Subject for books	
Trashed Books	

*Figure 4.28- Admin contact to seller*

BOOKLAND Kshiti Patel ▾

Home	Add User
Users	User name :
New users	<input type="text"/>
Edit my profile	E-mail :
View Available Books	<input type="text"/>
Add book to store	<a href="#">Add</a>
View Subjects	
Add Subject for books	
Trashed Books	

*Figure 4.29- Admin add user*

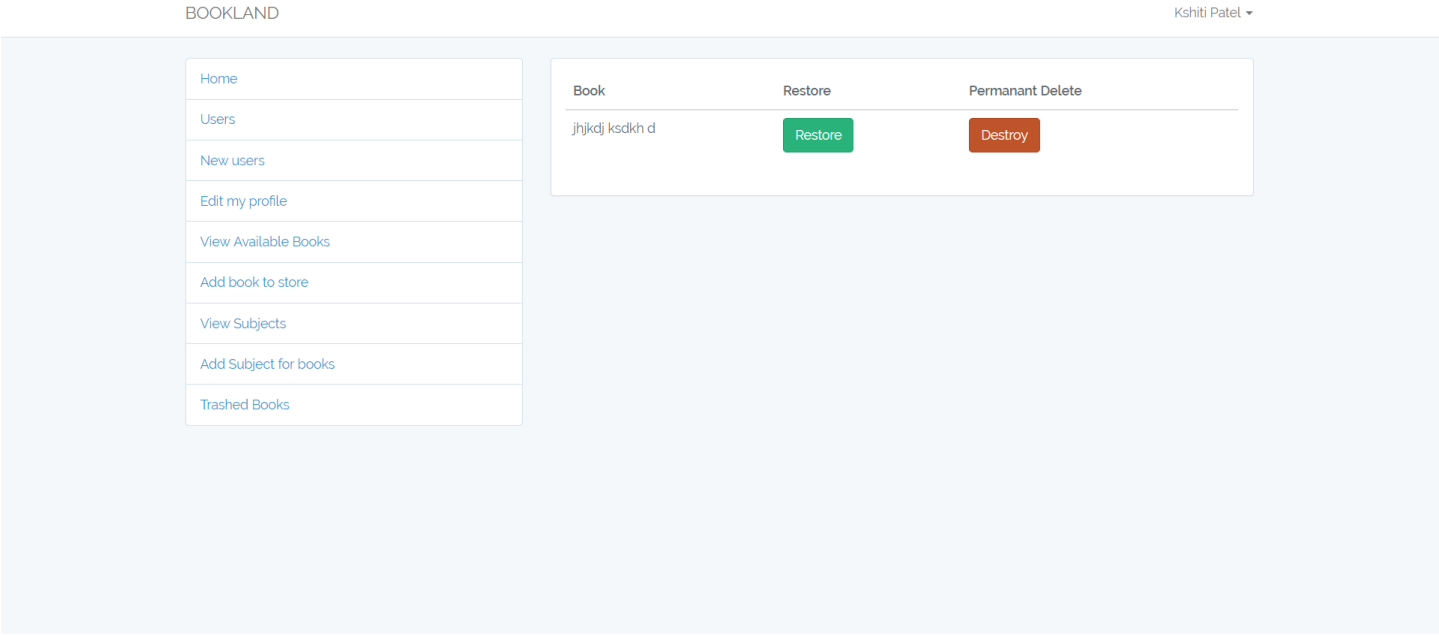


Figure 4.30- Admin trashed book

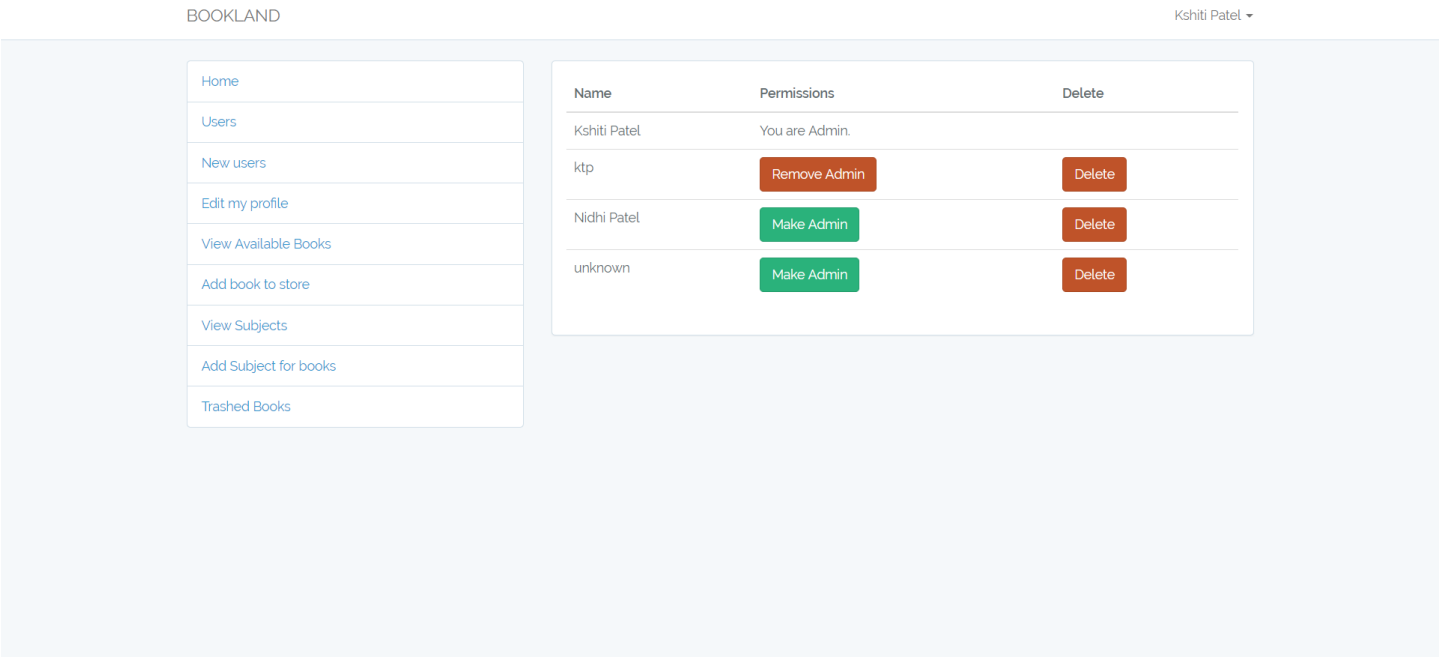


Figure 4.31- Admin user panel

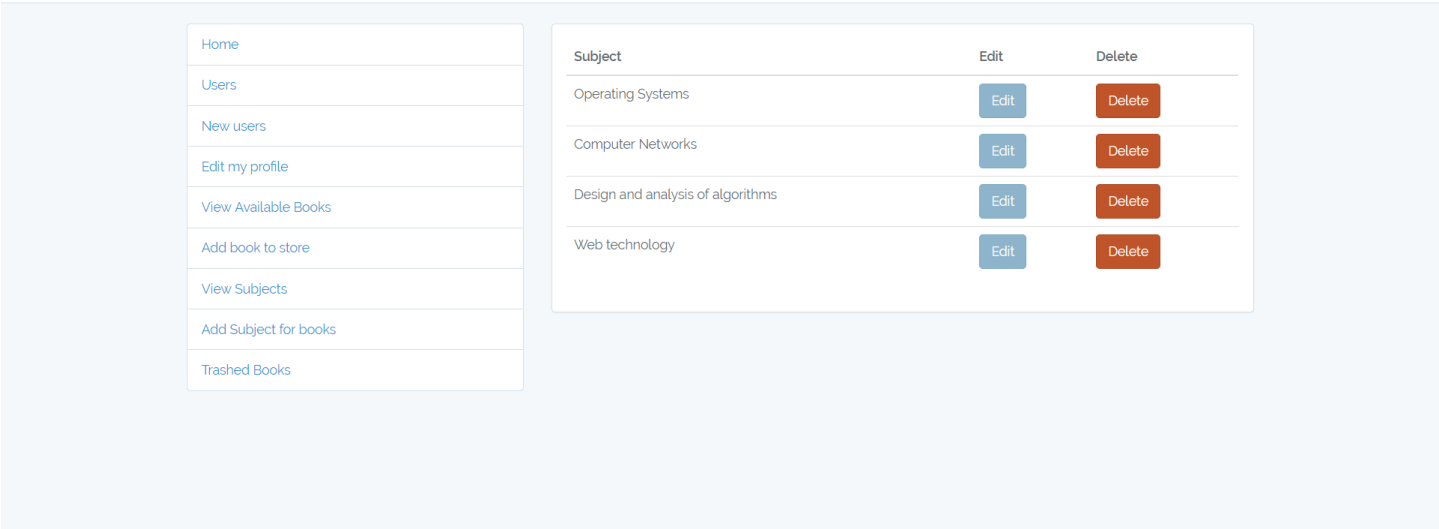


Figure 4.32- Admin view subjects

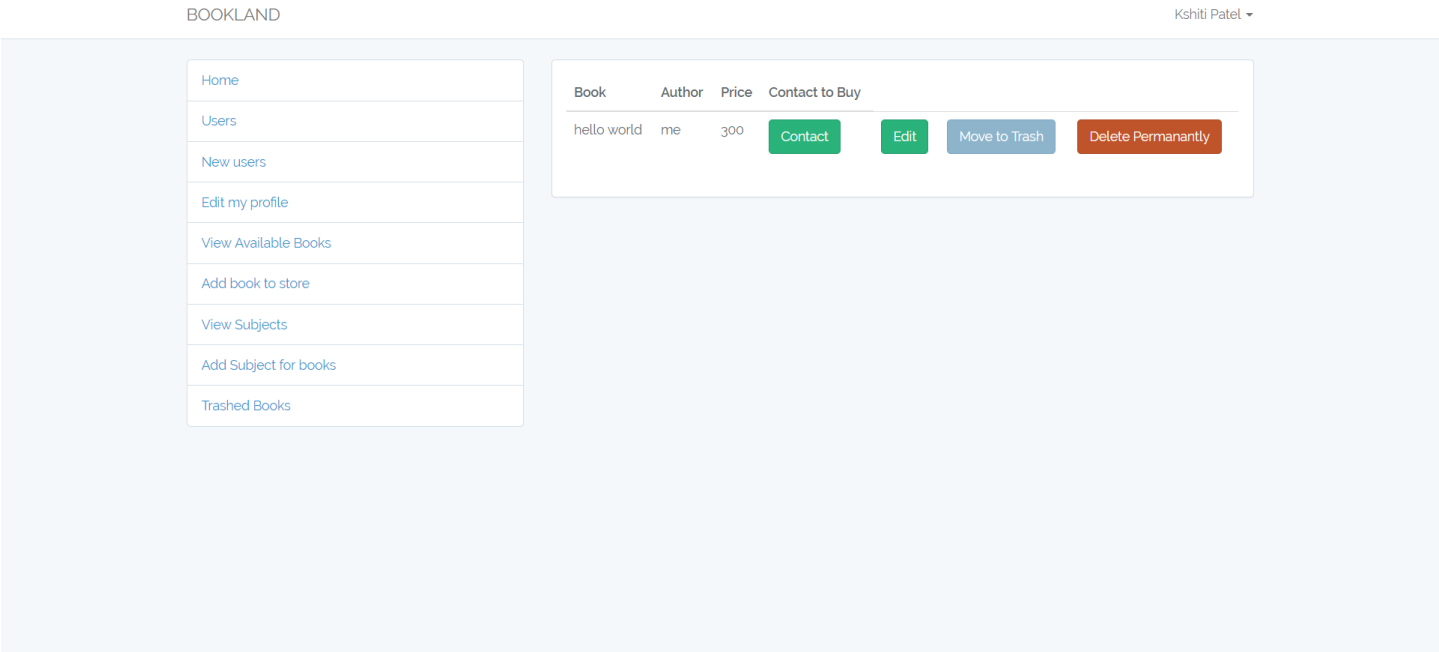


Figure 4.33- Admin view available books

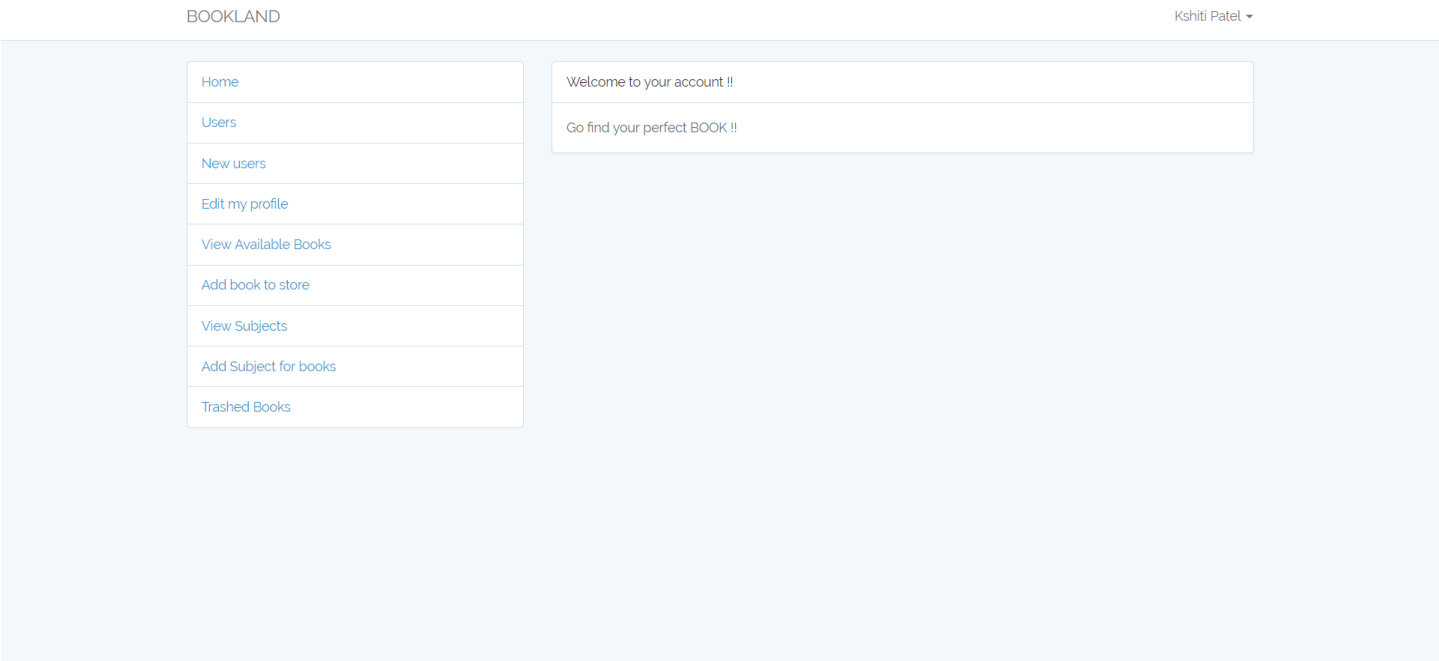


Figure 4.34- Admin home

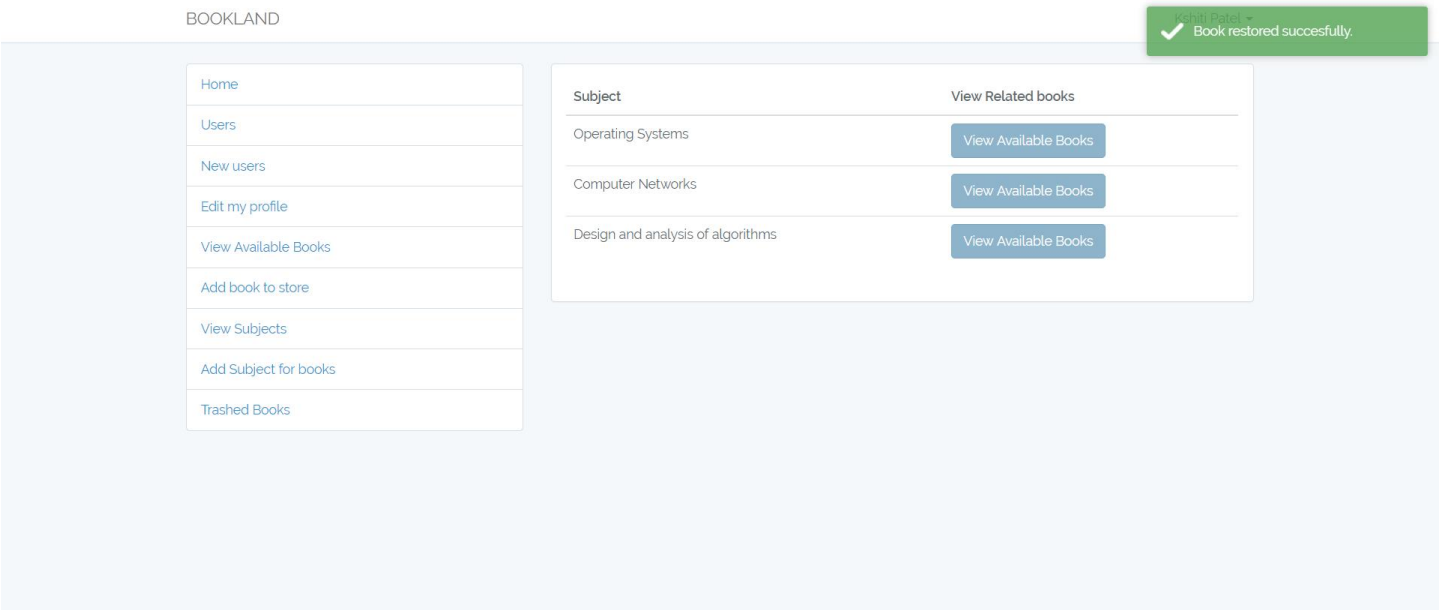


Figure 4.35- Admin not restore book

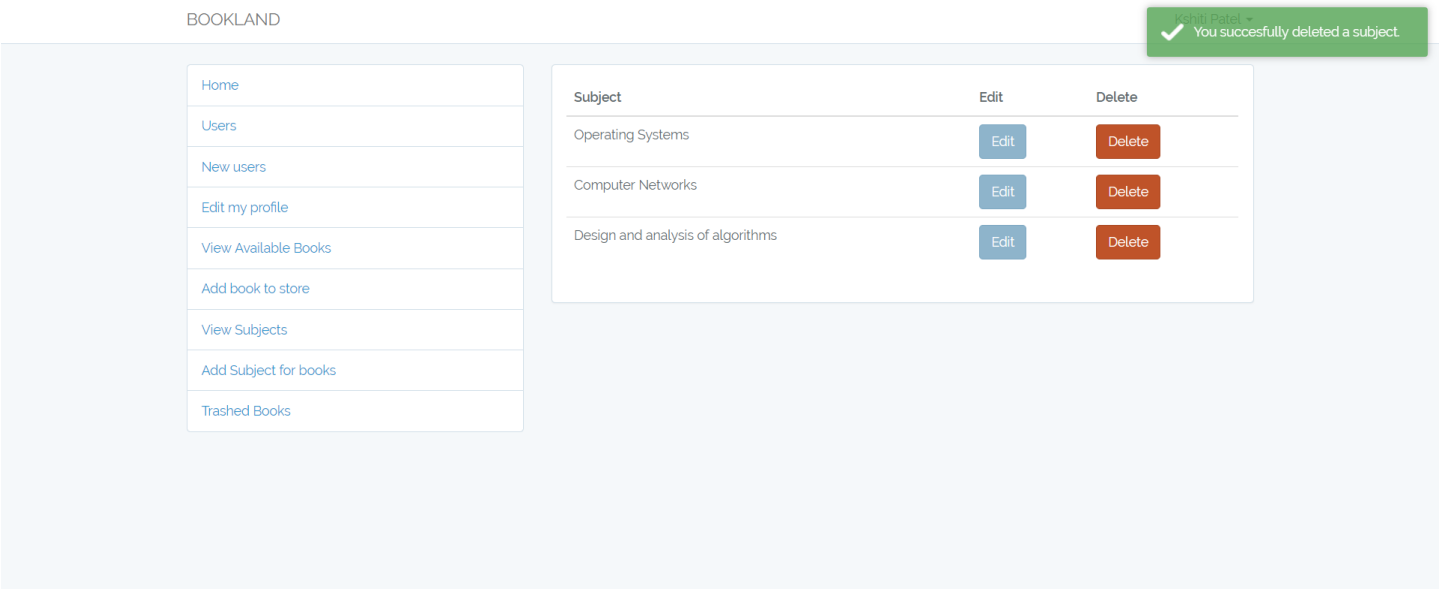


Figure 4.36- Toastr notification

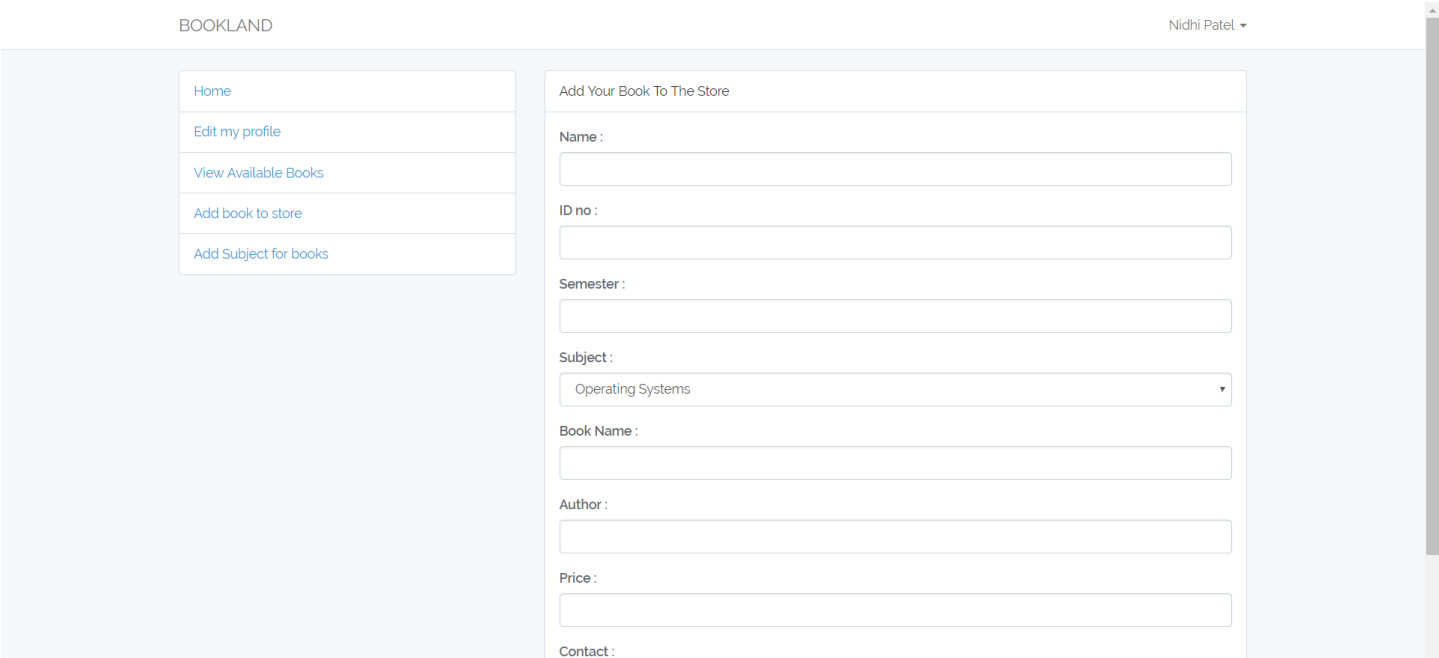


Figure 4.37- User add book

BOOKLAND

Nidhi Patel ▾

Home

Edit my profile

View Available Books

Add book to store

Add Subject for books

Add Subject

Subject :

Add Subject

Figure 4.38- User add subject if not selected

BOOKLAND

Nidhi Patel ▾

Home

Edit my profile

View Available Books

Add book to store

Add Subject for books

Book	Author	Price	Contact to Buy
hello world	me	300	<div>Contact</div>

Figure 4.40- User view book

## Chapter 5

### Constraints and Future enhancements

#### Constraints

- The seller cannot see their past activities.
- Spam records cannot be separated from real records, so cannot protect buyers from fake.
- Buyers cannot view the appearance and condition of the book since we didn't provide the upload picture functionality.

#### Future Enhancements

- Facility of spam record filtration will be added.
- Appearance of books will be added.
- Log of users activity will be added.

## **Chapter 6**

### **Conclusion**

This application ease the process of buying and selling books. This application software is user friendly and has required options which can be utilized by the user to perform the desired options. The software is developed using HTML, CSS and JavaScript as front end and MySQL as back end in windows environment.

The goals that are achieved by software are:

1. Optimum utilization of resources.
2. Efficient management of records.
3. Simplification of operations.
4. Less processing time and getting required information.
5. User friendly.
6. Portable and flexible for further enhancement.



## References

- **Laravel basics**  
<https://laravel.com/>
- **Udemy course for bookland application**  
<https://www.udemy.com/laravel-for-beginners-make-blog-in-laravel-5-2/>