**NAME:** Pulin Prabhu

**BATCH:** SE Comps, C

**ROLL NO.:**45

***EXPERIMENT 8***

**AIM:** Implementing Server Side Scripting using Laravel.

**THEORY:**

Laravel is a powerful MVC PHP framework, designed for developers who need a simple and elegant toolkit to create full-featured web applications. Laravel was created by Taylor Otwell. This is a brief tutorial that explains the basics of Laravel framework.

Laravel is an open-source PHP framework, which is robust and easy to understand. It follows a model-view-controller design pattern. Laravel reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic.

Laravel offers a rich set of functionalities which incorporates the basic features of PHP frameworks like CodeIgniter, Yii and other programming languages like Ruby on Rails. Laravel has a very rich set of features which will boost the speed of web development.

If you are familiar with Core PHP and Advanced PHP, Laravel will make your task easier. It saves a lot time if you are planning to develop a website from scratch. Moreover, a website built in Laravel is secure and prevents several web attacks.

***Advantages of Laravel***

Laravel offers you the following advantages, when you are designing a web application based on it −

* The web application becomes more scalable, owing to the Laravel framework.
* Considerable time is saved in designing the web application, since Laravel reuses the components from other framework in developing web application.
* It includes namespaces and interfaces, thus helps to organize and manage resources.

**Composer**

Composer is a tool which includes all the dependencies and libraries. It allows a user to create a project with respect to the mentioned framework (for example, those used in Laravel installation). Third party libraries can be installed easily with help of composer.

All the dependencies are noted in **composer.json** file which is placed in the source folder.

**Artisan**

Command line interface used in Laravel is called **Artisan**. It includes a set of commands which assists in building a web application. These commands are incorporated from Symphony framework, resulting in add-on features in Laravel 5.1 (latest version of Laravel).

***Features of Laravel***

Laravel offers the following key features which makes it an ideal choice for designing web applications −

**Modularity**

Laravel provides 20 built in libraries and modules which helps in enhancement of the application. Every module is integrated with Composer dependency manager which eases updates.

**Testability**

Laravel includes features and helpers which helps in testing through various test cases. This feature helps in maintaining the code as per the requirements.

**Routing**

Laravel provides a flexible approach to the user to define routes in the web application. Routing helps to scale the application in a better way and increases its performance.

**Configuration Management**

A web application designed in Laravel will be running on different environments, which means that there will be a constant change in its configuration. Laravel provides a consistent approach to handle the configuration in an efficient way.

**Query Builder and ORM**

Laravel incorporates a query builder which helps in querying databases using various simple chain methods. It provides ORM (Object Relational Mapper) and ActiveRecord implementation called Eloquent.

**Schema Builder**

Schema Builder maintains the database definitions and schema in PHP code. It also maintains a track of changes with respect to database migrations.

**Template Engine**

Laravel uses the Blade Template engine, a lightweight template language used to design hierarchical blocks and layouts with predefined blocks that include dynamic content.

**E-mail**

Laravel includes a mail class which helps in sending mail with rich content and attachments from the web application.

**Authentication**

User authentication is a common feature in web applications. Laravel eases designing authentication as it includes features such as register, forgot password and send password reminders.

**Redis**

Laravel uses Redis to connect to an existing session and general-purpose cache. Redis interacts with session directly.

**Queues**

Laravel includes queue services like emailing large number of users or a specified Cron job. These queues help in completing tasks in an easier manner without waiting for the previous task to be completed.

**Event and Command Bus**

Laravel 5.1 includes Command Bus which helps in executing commands and dispatch events in a simple way. The commands in Laravel act as per the application’s lifecycle.

***Structure of Laravel project***

The application structure in Laravel is basically the structure of folders, sub-folders and files included in a project. Once we create a project in Laravel, we get an overview of the application structure as shown in the image here.

The snapshot shown here refers to the root folder of Laravel namely laravel-project. It includes various sub-folders and files.

**App**

It is the application folder and includes the entire source code of the project. It contains events, exceptions and middleware declaration.

**Console**

Console includes the artisan commands necessary for Laravel. It includes a directory named Commands, where all the commands are declared with the appropriate signature. The file Kernel.php calls the commands declared in Inspire.php.

If we need to call a specific command in Laravel, then we should make appropriate changes in this directory.

**Events**

This folder includes all the events for the project.

Events are used to trigger activities, raise errors or necessary validations and provide greater flexibility. Laravel keeps all the events under one directory. The default file included is event.php where all the basic events are declared.

**Exceptions**

This folder contains all the methods needed to handle exceptions. It also contains the file handle.php that handles all the exceptions.

**Http**

The Http folder has sub-folders for controllers, middleware and application requests. As Laravel follows the MVC design pattern, this folder includes model, controllers and views defined for the specific directories.

The Middleware sub-folder includes middleware mechanism, comprising the filter mechanism and communication between response and request.

The Requests sub-folder includes all the requests of the application.

**Jobs**

The Jobs directory maintains the activities queued for Laravel application. The base class is shared among all the Jobs and provides a central location to place them under one roof.

**Listeners**

Listeners are event-dependent and they include methods which are used to handle events and exceptions. For example, the login event declared includes a LoginListener event.

**Policies**

Policies are the PHP classes which includes the authorization logic. Laravel includes a feature to create all authorization logic within policy classes inside this sub folder.

**Providers**

This folder includes all the service providers required to register events for core servers and to configure a Laravel application.

**Bootstrap**

This folder encloses all the application bootstrap scripts. It contains a sub-folder namely cache, which includes all the files associated for caching a web application. You can also find the file app.php, which initializes the scripts necessary for bootstrap.

**Config**

The config folder includes various configurations and associated parameters required for the smooth functioning of a Laravel application. Various files included within the config folder are as shown in the image here. The filenames work as per the functionality associated with them.

**Database**

As the name suggests, this directory includes various parameters for database functionalities. It includes three sub-directories-

Seeds − This contains the classes used for unit testing database.

Migrations − This folder helps in queries for migrating the database used in the web application.

Factories − This folder is used to generate large number of data records.

**Public**

It is the root folder which helps in initializing the Laravel application. It includes the following files and folders −

.htaccess − This file gives the server configuration.

javascript and css − These files are considered as assets.

index.php − This file is required for the initialization of a web application.

**Resources**

Resources directory contains the files which enhances your web application. The sub-folders included in this directory and their purpose is explained below −

assets − The assets folder include files such as LESS and SCSS, that are required for styling the web application.

lang − This folder includes configuration for localization or internalization.

views − Views are the HTML files or templates which interact with end users and play a primary role in MVC architecture.

Observe that the resources directory will be flattened instead of having an assets folder.

**Storage**

This is the folder that stores all the logs and necessary files which are needed frequently when a Laravel project is running. The sub-folders included in this directory and their purpose is given below −

app − This folder contains the files that are called in succession.

framework − It contains sessions, cache and views which are called frequently.

Logs − All exceptions and error logs are tracked in this sub folder.

**Tests**

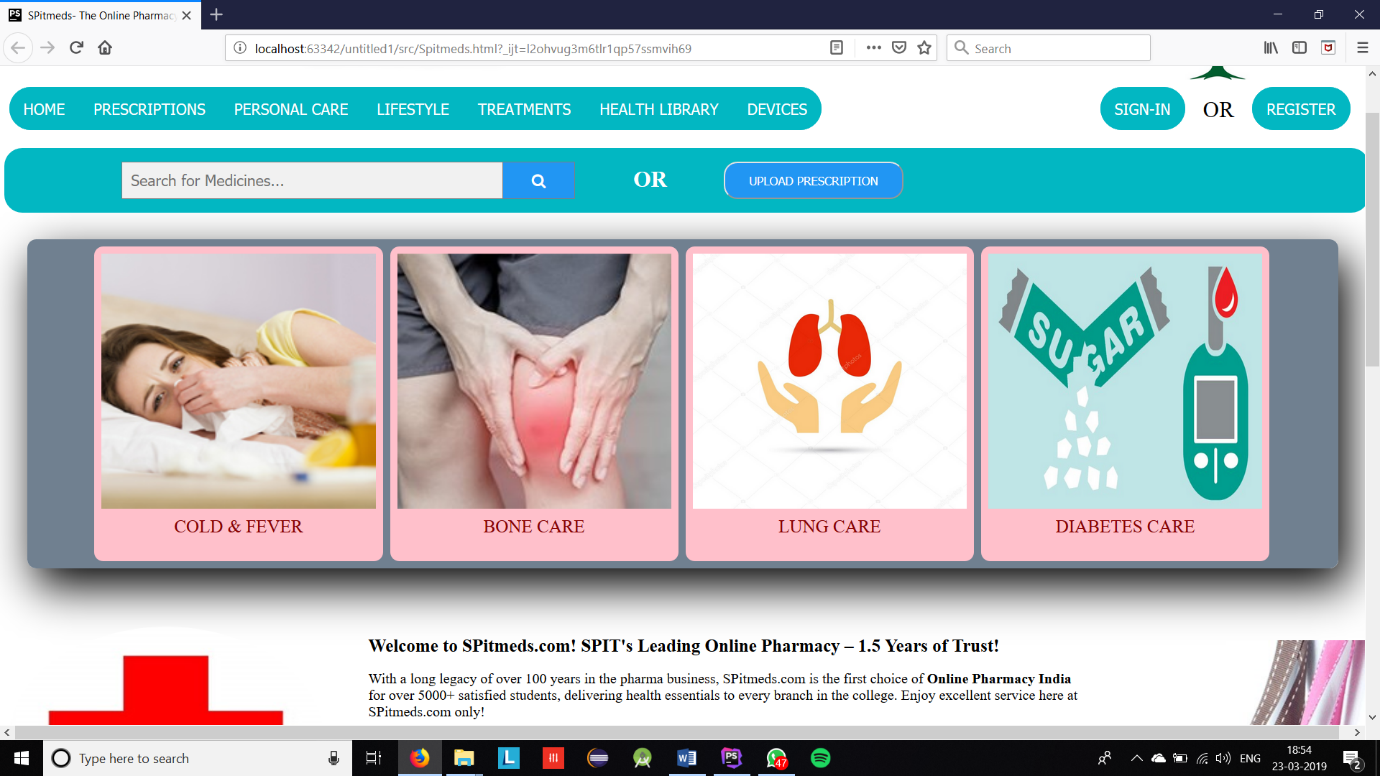
All the unit test cases are included in this directory. The naming convention for naming test case classes is camel\_case and follows the convention as per the functionality of the class.

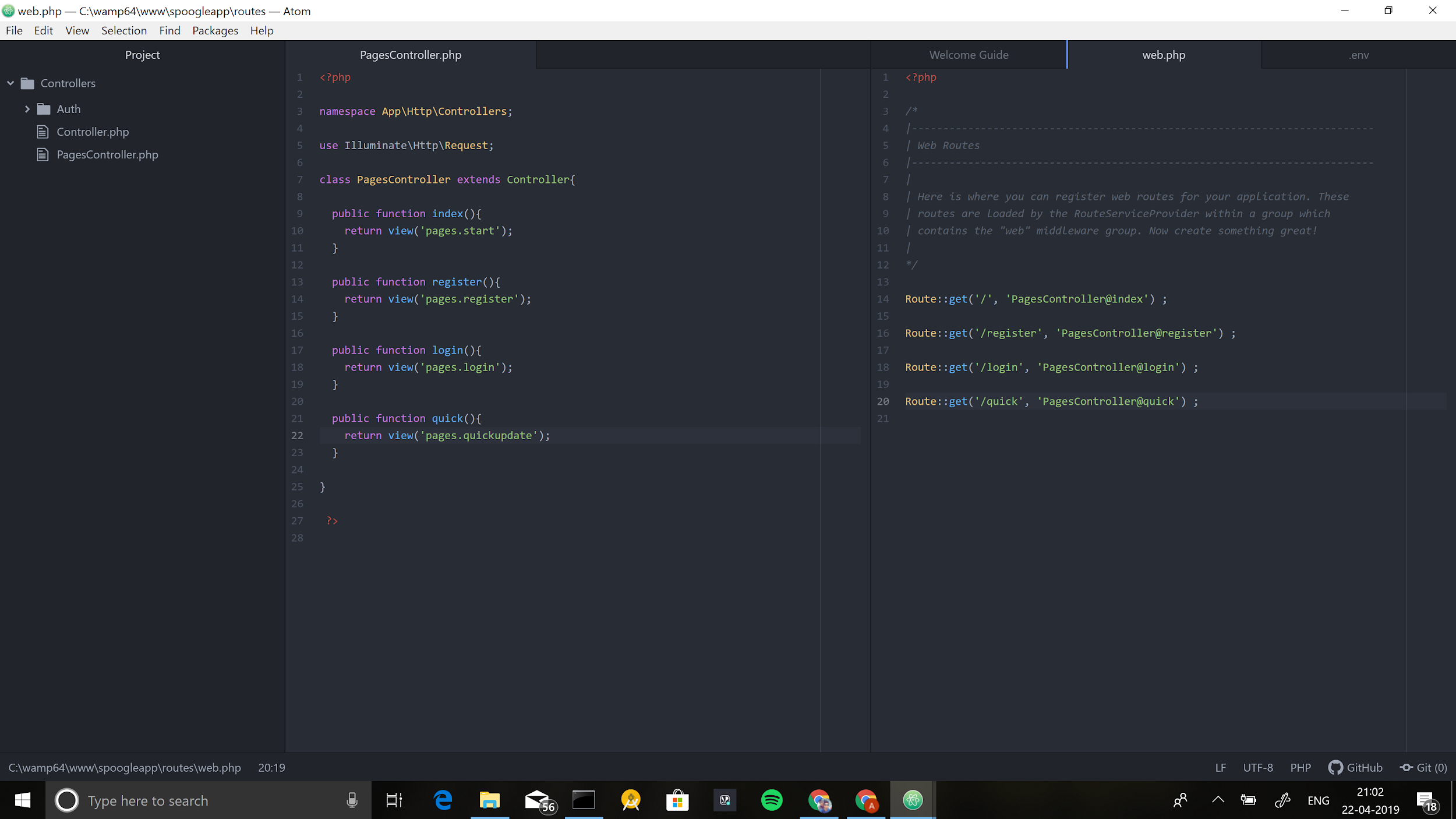
**Vendor**

Laravel is completely based on Composer dependencies, for example to install Laravel setup or to include third party libraries, etc. The Vendor folder includes all the composer dependencies.

In addition to the above mentioned files, Laravel also includes some other files which play a primary role in various functionalities such as GitHub configuration, packages and third party libraries.

**SCREENSHOTS:**



****

**CONCLUSION:** Laravel is php framework which makes web application becomes more scalable and considerable time is saved in designing the web application, since Laravel reuses the components from other framework in developing web application.