Name of Assignment:

Laravel Installation and Folder Structure Assignment

Part 1: Laravel Installation

Part 2: Laravel Folder Structure

Full Name:  Momtarin Jahan

Module: 13

Subject: Web Development with PHP & Laravel

Date of Submission: 15/05/2023


Part 1: Laravel Installation

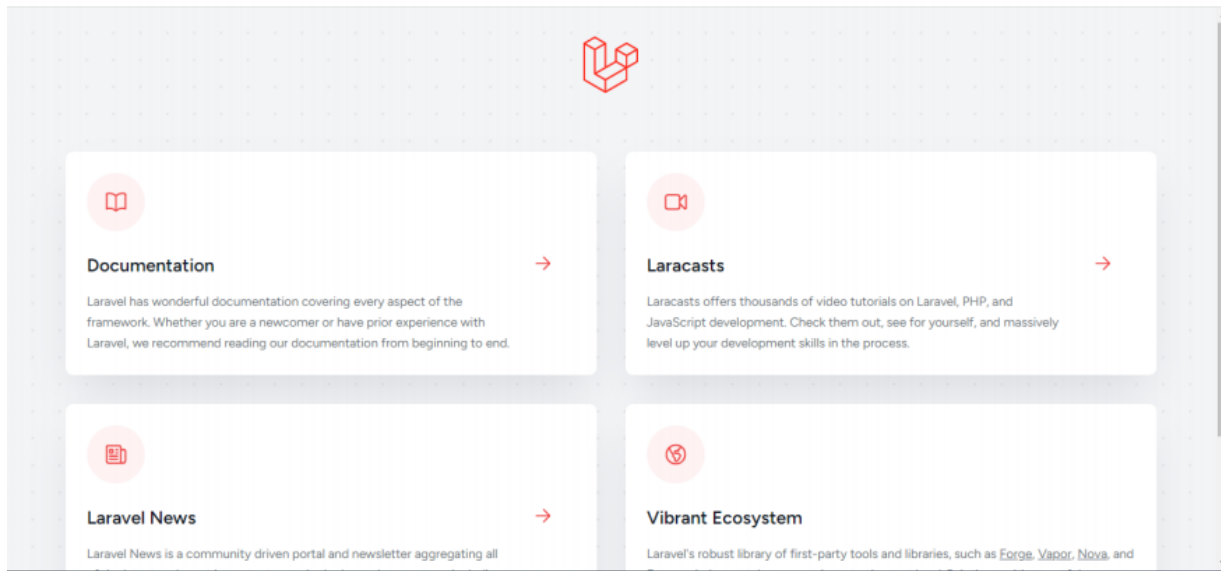To install Laravel, I followed the steps below:

Installed PHP and Composer via Homebrew as recommended in the Laravel

documentation.

Installed Node and NPM.

Created a new Laravel project using the following command in the terminal:

'composer create-project laravel /laravel example-app'



where example-app is the name of my new Laravel project.

After creating the project, I navigated to the project's root directory in the terminal and

ran the following command to start the Laravel development server:

"php artisan serve"

I then visited the URL [http://localhost:8000](http://localhost:8000) in my web browser, and I saw the default

Laravel welcome page. I took a screenshot of the welcome page as proof that Laravel

was installed and working correctly.

Part 2: Laravel Folder Structure

Here is a brief description of each of the folders in a Laravel project:

- app: contains the core code of the application, including controllers, models,

views, and other supporting files.

- bootstrap: contains the files needed to bootstrap the framework and configure

autoloading.

- config: contains configuration files for the application, such as database

connection details and environment variables.

- database: contains database-related files, including migrations and seeders.

- public: contains the front-facing assets of the application, such as CSS,

JavaScript, and images.

- resources: contains the uncompiled resources, such as blade templates and

JavaScript files, that will be compiled by Laravel's asset compilation system.

- routes: contains the route definitions for the application.

- storage: contains files that need to be stored for the application, such as logs,

cache files, and user-generated content.

- tests: contains the automated tests for the application.

• vendor: contains the application's dependencies installed via Composer.

To create a new route in my Laravel project that displays a "Hello, World!" message, I

added the following code to the routes/web.php file:

Route::get('/hello', function () {

return 'Hello, World!';

Hello world

});

I then navigated to the URL http://localhost:8000/hello in my web browser, and I saw

the "Hello, World!" message. I took a screenshot of the message as proof that the new

route was working correctly.