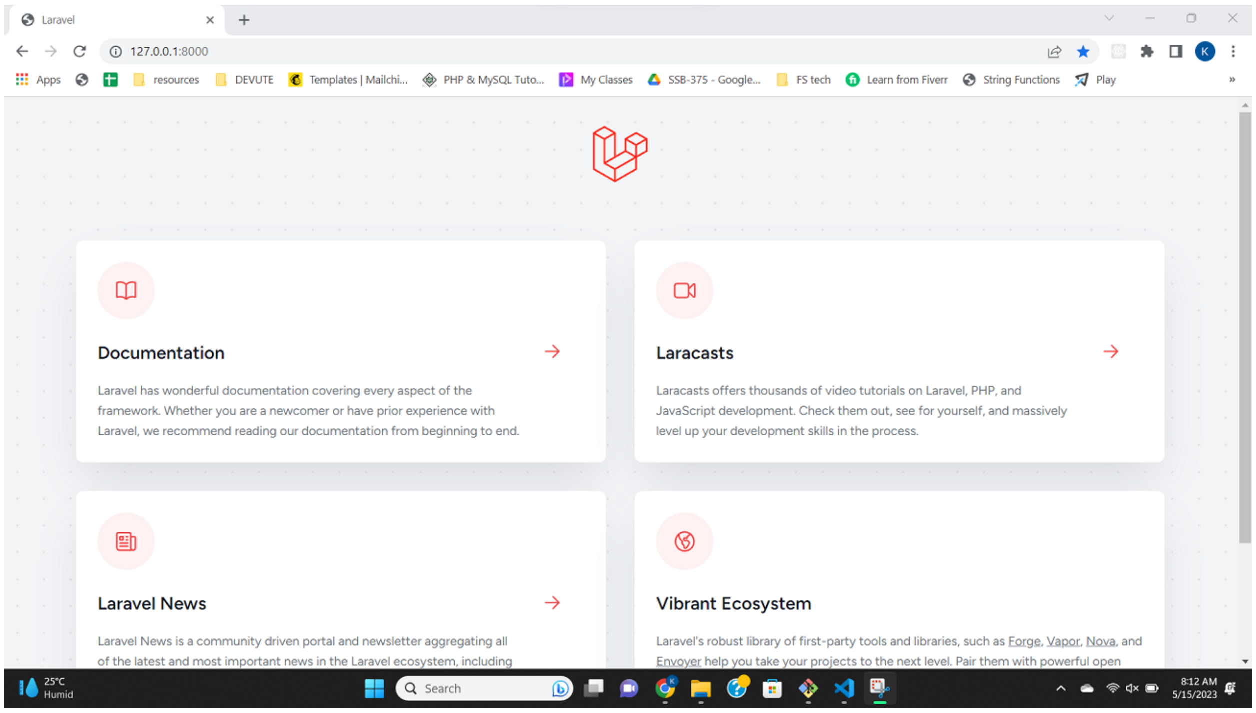
Laravel Installation and Folder Structure Assignment

Part 1: Laravel Installation

Here are the steps to install Laravel using the Laravel installer:

1. To start the installation process, I check if Composer is installed on my system. If not, I go to the official website for Composer (https://getcomposer.org/) and download and install it on my computer.
2. With Composer installed, I use it to install the Laravel installer globally on my system by opening my terminal and running the command "**composer global require laravel/installer".**
3. Once the installation is complete, I create a new Laravel project by running the command "**laravel new project-name**" in my terminal, replacing "project-name" with the name of my project.
4. To ensure that Laravel is installed and working correctly, I navigate to my project directory and run the development server by running the command "**php artisan serve**".
5. This will start the development server at **http://localhost:8000**. I then open my web browser and navigate to **http://localhost:8000** to see the Laravel welcome screen. If the welcome screen is displayed, I can be sure that Laravel is installed and working correctly.



Documentation

Laravel has wonderful documentation covering every aspect of the framework. Whether you are a newcomer or have prior experience with Laravel, we recommend reading our documentation from beginning to end.

Laracasts

Laracasts offers thousands of video tutorials on Laravel, PHP, and JavaScript development. Check them out, see for yourself, and massively level up your development skills in the process.

Laravel News

Laravel News is a community driven portal and newsletter aggregating all of the latest and most important news in the Laravel ecosystem, including

Vibrant Ecosystem

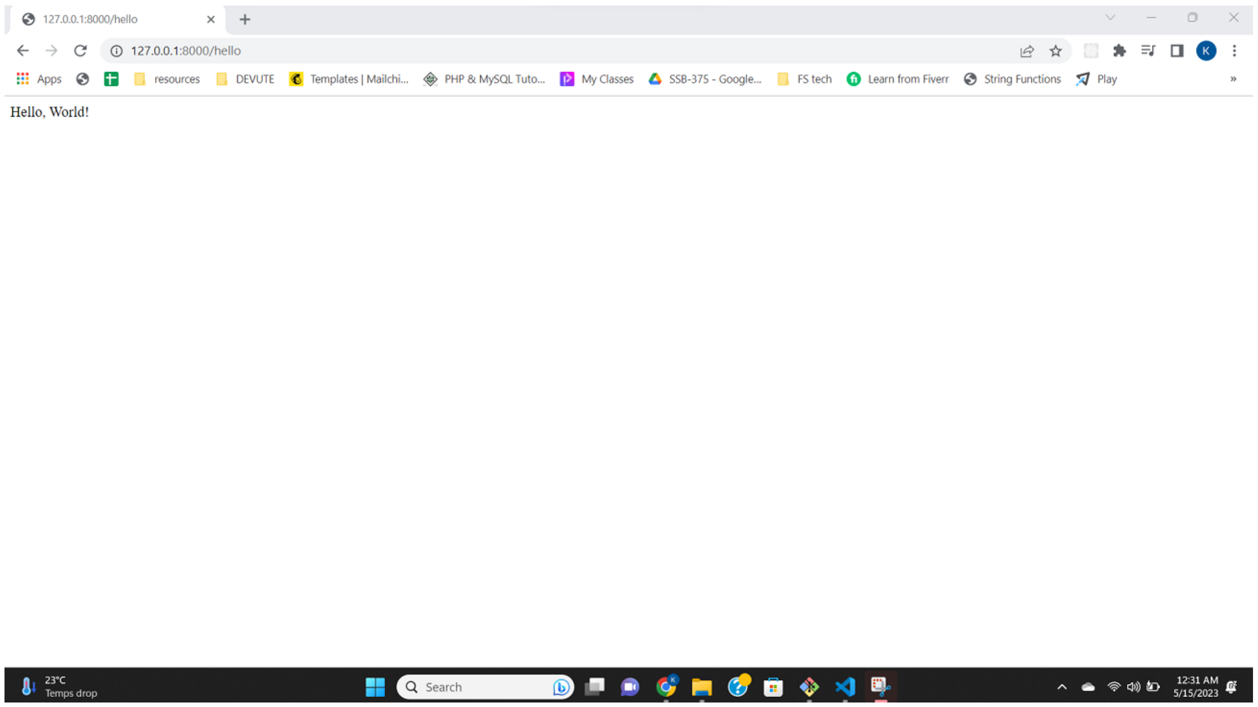
Laravel's robust library of first-party tools and libraries, such as Forge, Vapor, Nova, and Envoyer help you take your projects to the next level. Pair them with powerful open

Figure : Part 1

Part 2: Laravel Folder Structure

Here's a description of each folder in a Laravel project:

1. **app:** This directory contains the core code of the application, including the models, controllers, and other application logic. Developers can create subdirectories within the app directory to organize their code, but it's not strictly necessary.
2. **bootstrap:** This directory contains the files necessary to bootstrap the Laravel framework. This includes the app.php file, which initializes the framework, and the autoload.php file, which registers the Composer autoloader.
3. **config:** This directory contains all of the application's configuration files, including database configuration, caching configuration, and other settings.
4. **database:** This directory contains the application's database migrations, as well as any seed data that developers may want to include with the application.
5. **public:** This directory contains the application's public-facing files, including the index.php file, which serves as the entry point for the application, and any static assets (such as images, CSS files, and JavaScript files) that the application uses.
6. **resources:** This directory contains the application's views (which are used to generate the HTML that the application sends to the client), as well as any other resources that the application needs (such as language files, JavaScript files, and SASS or LESS files).
7. **routes:** This directory contains all of the application's route definitions. Routes define the URL patterns that the application should respond to, and specify which controller method should handle each request.
8. **storage:** This directory contains the application's temporary and cache files, as well as any uploaded files that the application receives.
9. **tests:** This directory contains the application's automated tests. Laravel includes a powerful testing framework that makes it easy to write tests for the application's functionality.
10. **vendor:** This directory contains the application's Composer dependencies. When developers run "**composer install**" , Composer downloads all of the application's dependencies (including Laravel itself) and installs them in the vendor directory.



```
Hello, World!
```

Figure: Part 2