**Laravel Installation and Folder Structure Assignment**

**Part 1: Laravel Installation**

1. To install a Laravel project using Composer at first, I have run following command in terminal-

composer create-project laravel/laravel Laravel-Installation-and-Folder-Structure

1. Now, I must test this project to work or not. I can run this development project on my pc using the following command in terminal-

php artisan serve

1. It’s given an URL. By clicking this I’ve seen the following view. It means my installation process is ok and work perfectly.

INFO Server running on [http://127.0.0.1:8000].

A screenshot of a computer

Description automatically generated

**Part 2: Laravel Folder Structure**

Here the description of the purpose of each of the provided folders in a Laravel project:

1. **app:** The 'app' folder contains the core of Laravel application. It houses the application's models, controllers, middleware and other essential classes. It also contains kernel.php file.
2. **bootstrap:** The 'bootstrap' folder in a Laravel project is where the necessary files for initializing the Laravel framework are stored. It houses the 'app.php' file, which plays a crucial role in initializing the application. This file is responsible for setting up essential components such as the services, packages and other vital aspects needed for the proper functioning of the Laravel framework.
3. **config:** The 'config' folder contains configuration files for Laravel application. It includes files such as 'app.php' (general application configuration), 'database.php' (database connection settings), 'mail.php' (email configuration) and more. We can modify these files to customize the behavior of the application.
4. **database:** The 'database' folder holds database-related files and migrations. It contains the 'migrations' subfolder, where we define and manage database schema changes using migration files. We can also find a 'seeders' subfolder where we can define seeders to populate the database with sample data.
5. **public:** The 'public' folder is the web root of Laravel application. It contains the 'index.php' file, which serves as the entry point for all incoming HTTP requests. This folder also houses publicly accessible assets like CSS, JavaScript, images and any other files that need to be accessible from the web.
6. **resources:** The 'resources' folder contains various assets and resource files for application. It includes the 'views' subfolder, where we store Blade templates (PHP-based templates) that define the structure and presentation of application’s user interface. Additionally, we can find subfolders for language files, CSS and JavaScript assets and other resources.
7. **routes:** The 'routes' folder contains the routing configuration for Laravel application. It includes the 'web.php' file, where we define routes that handle web requests and the 'api.php' file, which defines routes for application’s API endpoints. We can create additional route files or group routes based on the application’s needs.
8. **storage:** The 'storage' folder contains various files generated and used by the application. It includes subfolders for storing application logs, cached views, session files, uploaded files and other temporary or persistent data. This folder should have written permissions to allow the application to store and retrieve files.
9. **tests:** The 'tests' folder holds the automated tests for Laravel application. It includes test files that we can write using Laravel's testing framework. We can create unit tests, feature tests and more to ensure the application behaves as expected.
10. **vendor:** The 'vendor' folder contains all the third-party libraries and dependencies installed via Composer. Laravel itself resides in this folder along with other packages and libraries application requires. We don't typically modify files in this folder directly, as Composer manages it for us.

**Route Creation:**

1. To create a new route in Laravel project first go to **routes** folder. Then open web.php file and write the required php code for routing.
2. Now displaying a simple "Hello, World!" message following screenshot.

A screenshot of a computer

Description automatically generated