**Laravel Installation and GitHub Deployment Process I followed:**

**Installed Laravel:**

Using Composer, the latest version of Laravel was installed globally on the system.

**Created a new Laravel project:**

A new Laravel project was created using the command laravel new project-name.

**Started the Laravel development server:**

Navigated to the project directory and started the development server using php artisan serve.

**Initialized a Git repository:**

In the project directory, initialized a new Git repository using *git init.*

**Staged the project files:**

Staged all the Laravel project files for commit using *git add ..*

**Committed the project files:**

Committed the project files with an initial commit message using *git commit -m "laravel project installed for ostad php laravel"*.

**Created a new GitHub repository:**

Created a new repository named 'laravel-project' on GitHub.

**Linked the local repository to the GitHub repository:**

Added the GitHub repository as a remote repository using *git remote add origin https://github.com/devhafix/laravel-project.git.*

**Encountered an authentication error:**

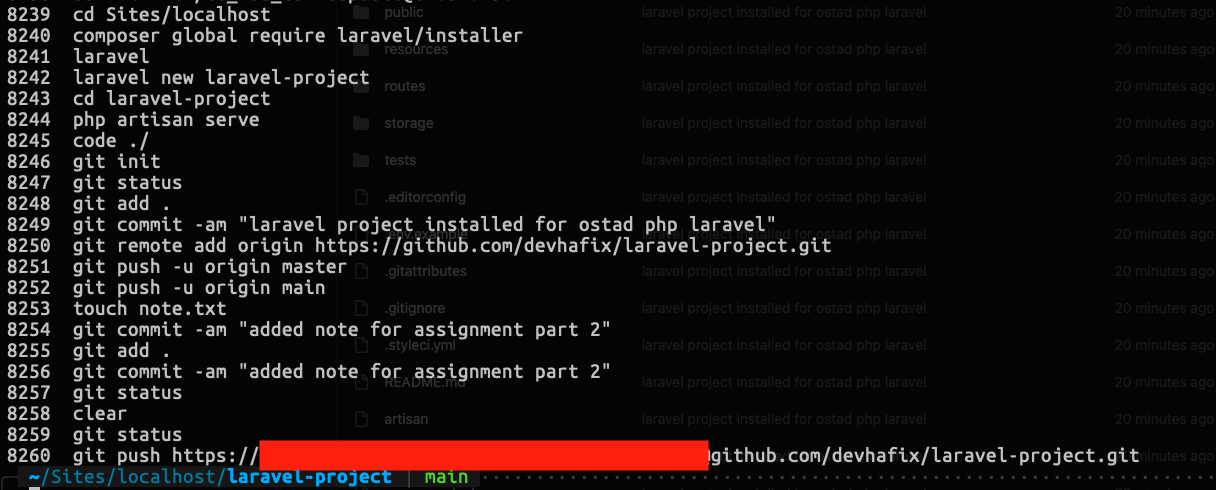
Faced an error when trying to push to GitHub due to GitHub discontinuing support for password authentication.

**Generated a GitHub Personal Access Token (PAT):**

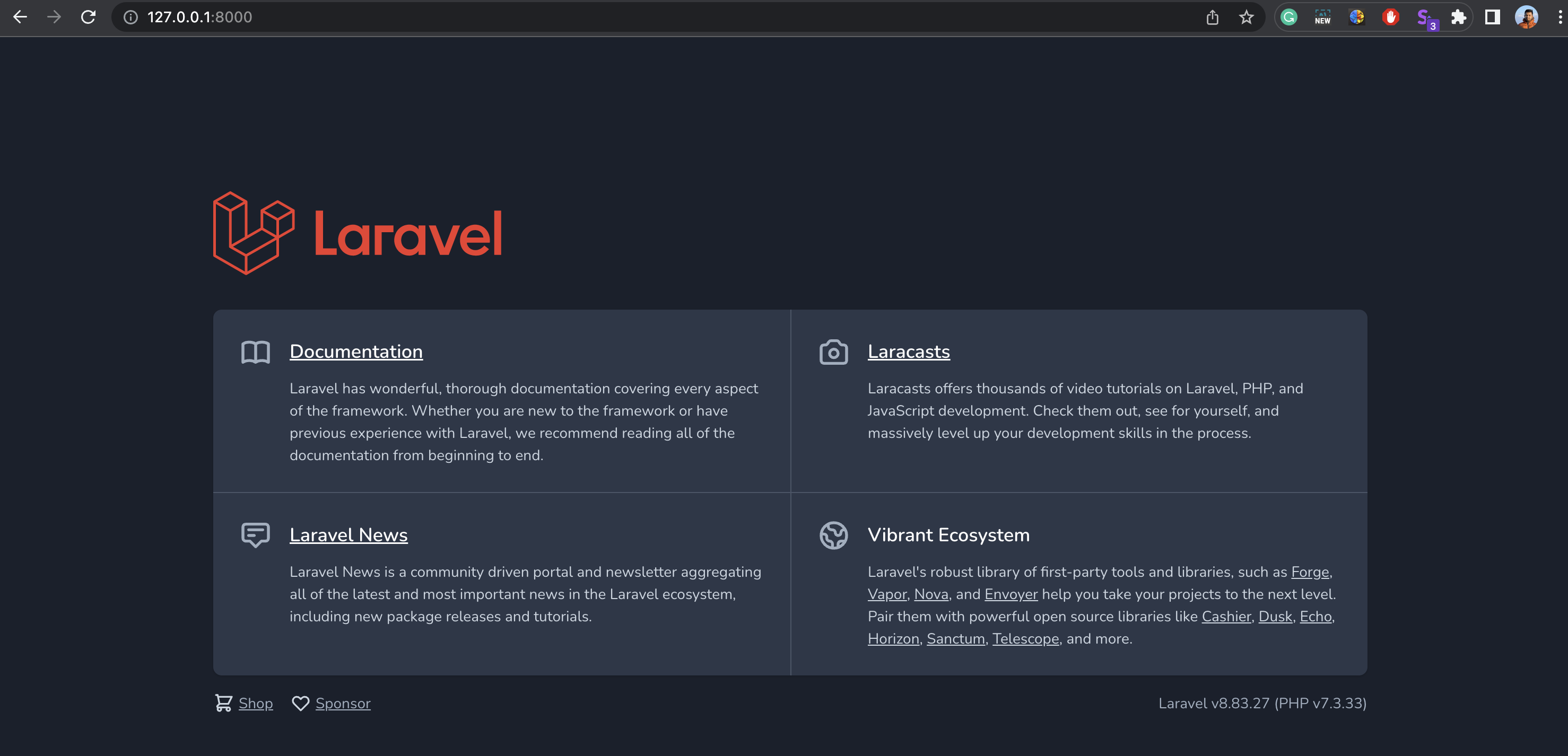
To solve the authentication issue, created a personal access token on GitHub as a replacement for password authentication.

**Pushed the project to GitHub:**

Pushed the Laravel project to the GitHub repository using the personal access token for authentication, with the command *git push https://<token>@github.com/devhafix/laravel-project.git.*



Running project using *php artisan serve*



Hello world route

<http://127.0.0.1:8000/hello>



Part 2: Laravel Folder Structure

app: This directory is considered the core of any Laravel application. It houses essential components such as models, controllers, and providers, all of which contribute to the overall application logic.

bootstrap: This directory contains scripts that bootstrap, or initiate, the Laravel framework. This includes the autoloader and the application start file.

config: All configuration files for the Laravel application are located in this directory. These files are essential for managing various settings throughout the application.

database: This directory houses database-related files, including migrations and seeds. Migrations provide a form of version control for the database schema, while seeds allow for the insertion of initial data into the database.

public: The public directory serves as the application's document root. It initiates the Laravel application and handles autoloading.

resources: This directory contains views, raw assets such as LESS, SASS, CoffeeScript, and localization files. These elements are essential for the front-end aspect of the application.

routes: The routes directory contains all route definitions for the application. These routes determine how an application responds to client requests.

storage: This directory is used for storing compiled Blade templates, file-based sessions, file caches, and other files generated by the framework during runtime.

tests: All automated tests for the application are stored in this directory.

vendor: This directory holds all Composer dependencies. Composer is a tool used for dependency management in PHP, and its dependencies are stored here.

Thank You