Part 2: Laravel Folder Structure

Describe the purpose of each of the following folders in a Laravel project:

1. App: app is core folder where most of the application's functionality is implemented.

It has several subfolders as below.

Http: Contains the controllers, middleware, and requests for handling HTTP requests and responses.

Providers: Contains the service providers that bootstrap various parts of the Laravel framework and register application-specific services.

Console: Contains the console commands that can be run in the command-line interface.

Exceptions: Contains the exception handlers for handling exceptions that occur within the application.

Models: Contains the application's data models that represent the data entities of the application.

Services: Contains the application's services that provide specific functionality to the application.

1. Bootstrap: Laravel initializes itself and sets up some basic configurations and services needed to run the application.

app: Contains the "app.php" file, which loads the application and returns an instance of the application.

cache: Contains the cached files of the application's configuration, routes, and services. This speeds up the application's bootstrapping process by loading the cached files instead of recompiling them every time the application runs.

config: Contains the configuration files of the application.

database: Contains the database-related configuration files and migrations.

framework: Contains the framework-related files and scripts, such as the exception handler and session management.

logs: Contains the application's log files.

1. Config: it contains files that configure different aspects of the application, such as the database connection, cache configuration, application environment, and more.

app.php: Contains the configuration for the application's name, environment, timezone, and more.

database.php: Contains the configuration for the application's database connections.

cache.php: Contains the configuration for the application's caching system.

mail.php: Contains the configuration for the application's email settings.

services.php: Contains the configuration for the application's third-party services

1. database: where database-related files and scripts are stored

migrations: Contains database migration files, which define the changes to the database schema and allow developers to version control the database structure.

seeds: Contains database seed files, which allow developers to insert sample data into the database for testing purposes.

factories: Contains model factory files, which define the default values for the application's data models.

database.php: Contains the configuration for the application's database connections.

sqlite-database.sqlite: A default SQLite database file that can be used for local development and testing.

1. Public: It is where the application's front controller, index.php, resides, and where all incoming HTTP requests are directed to.
2. index.php: The front controller of the application that handles all incoming HTTP requests and loads the application.
3. .htaccess: A hidden file that contains Apache web server configuration settings for the application.
4. assets: Contains static files such as images, stylesheets, and JavaScript files that are publicly accessible.
5. css, js, images: Subfolders within the assets folder for organizing different types of static files.
6. favicon.ico: The icon file that appears in the browser's address bar and tab.

5) Resources:it contains files that are used for building the application's user interface and other assets such as language files, views, and assets.

assets: Contains uncompiled assets such as Sass or Less files, JavaScript files, and images.

lang: Contains language files that define the application's translations and localizations.

views: Contains the application's view files, which define the user interface of the application.

js: Contains JavaScript files that are used in the application's user interface.

sass: Contains Sass files that define the application's stylesheets.

7) routes: it manage the application's URL routing.

web.php: Contains routes for handling web requests and rendering views.

api.php: Contains routes for handling API requests and returning JSON responses.

console.php: Contains routes for handling commands that are executed through the command line.

channels.php: Contains routes for handling Laravel's broadcast system.

webhooks.php: Contains routes for handling incoming webhooks from external services.

8) storage: ontains files that are generated by the application and should not be directly accessible from the public web.

tests

9) Vendor: contains files that are not directly related to the application's logic but are necessary for its proper functioning.