

Deploying a Laravel Application to a Production Server.

Hazem Hadi

Testers Co

2023-04-24T08:08:08Z

Abstract

This tutorial describes the steps needed to be completed to deploy a Laravel application on a production server.

Before deploying a Laravel application to a production server, there are a few things that need to be done:

1. Ensure that the server meets the Laravel application's requirements. This includes PHP version, database software, and any necessary extensions.
2. Set up a web server on the production server. Apache or Nginx are the most commonly used web servers for Laravel applications.
3. Configure the web server to serve the Laravel application. This involves creating a virtual host or server block that points to the application's public directory.
4. Set up a database for the Laravel application. This can be MySQL, PostgreSQL, or another supported database software.
5. Configure the Laravel application to use the production database. This involves updating the .env file with the database credentials.
6. Install Composer on the production server. Composer is a dependency manager for PHP, and is used to install the Laravel application's dependencies.
7. Clone the Laravel application's code onto the production server. This can be done using Git, FTP, or another method.
8. Install the Laravel application's dependencies using Composer.
9. Run any necessary database migrations or seeders to set up the production database.

10. Configure the Laravel application to use the production environment. This involves updating the `.env` file with the `APP_ENV` variable set to “production”.
11. Set up any necessary cron jobs or scheduled tasks for the Laravel application.
12. Test the Laravel application to ensure that it is working correctly on the production server.