Step 1: Download XAMPP

Go to the official Apache Friends website.

Click on the XAMPP for Windows button to start the download.

Step 2: Install XAMPP

Once the download is complete, navigate to your Downloads folder and double-click on the XAMPP installer file.

A setup wizard will open. Click Next to proceed.

You will be asked to select components to install. Make sure to select **Apache, MySQL, and PHP** as they are required for running a server. Click Next.

Choose the installation directory. It's recommended to use the default directory.

Continue following the instructions on the setup wizard.

Click Next to begin the installation. The installation process may take a few minutes.

Once the installation is complete, click Finish.

Step 3: Launch XAMPP

Open the XAMPP Control Panel. You can do this by searching for XAMPP in your start menu or navigating to the folder where you installed XAMPP and clicking on the xampp-control icon.

In the control panel, **start the Apache and MySQL services**.

Now, XAMPP is installed and running on your machine. You can test it by opening a web browser and navigating to **http://localhost**. You should see the XAMPP dashboard.

Step 4: Configure PHP for Laravel

Open the php.ini file located in the php folder of your XAMPP installation directory.

Make sure the following extensions are uncommented (remove the semicolon at the start):

extension=curl

extension=fileinfo

extension=mbstring

extension=openssl

extension=pdo_mysql

Save and close the php.ini file.

Step 5: Download Composer

Go to the official Composer website.

Click on the Composer-Setup.exe link to download the installer.

Step 6: Install Composer

Once the download is complete, navigate to your Downloads folder and double-click on the Composer installer file.

A setup wizard will open. Follow the instructions to install Composer.

During the installation, you'll be asked to specify the location of the PHP executable. If you've installed XAMPP, this will be in the php folder inside your XAMPP installation directory.

Once the installation is complete, you can close the setup wizard.

Step 7: Verify the Installation

Open a new command prompt.

Type composer and press Enter.

If Composer is installed correctly, you should see a list of Composer commands. If not, you'll see an error message.

Step 8: Copy Laravel Project

Navigate to the htdocs folder in your XAMPP installation directory.

Copy your Laravel project folder into the htdocs folder.

Step 9: Install Project Dependencies

Open a command prompt in your Laravel project.

Run composer install to install all the project dependencies.

Step 10: Set Up Environment File // Ignore this if you are importing the recipe_app.sql

In your Laravel project folder, find the. env.example file.

Rename the .env.example file to .env.

Open the .env file and update the database credentials (DB_HOST, DB_PORT, DB_DATABASE, DB_USERNAME, DB_PASSWORD) to match your local environment.

Step 11: Generate Application Key

In the command prompt, run php artisan key:generate to generate a new application key. This will update the APP_KEY in your .env file.

Step 12: Set up the database:

Make sure MySql is running - Open a web browser and navigate to http://localhost/phpmyadmin

In phpMyAdmin, click on the Databases tab.

Enter recipe_app in the Create database field.

Click the Create button.

Click on the recipe_app database on the left panel to select it.

Click on the Import tab at the top of the page.

Click the Choose File button and select your recipe_app.sql file (you can find this is in the below path: C:\xampp\htdocs\RecipeManagementApp\app\DB_Import).

Click the Go button at the bottom of the page to start the import process.

Step 13: Run Database Migrations

If your Laravel project includes database migrations, run **php artisan migrate** in the command prompt to run these migrations.

Step 14: Install node and npm.

Step 15:Run npm run dev

Step 15: Run php artisan serve – This will give an url. Hit the URL in browser, The application will be open.

//The database table structures will be automatically stored in the database migration files. Executing php artisan serve will automatically create table.