

Web Application Programming

Week 11. Advanced Front-End Concepts(Introduction to front-end frameworks (e.g.,
Laravel)

By Elubu Joseph - MSc.IS

Lecturer

Department of Information Technology

Kumi University

[Email: josebulinda@gmail.com](mailto:josebulinda@gmail.com) or jose@kumiuniversity.ac.ug

Agenda

1. Summary of previous lecture
2. Advanced Front-End Concepts(Introduction to front-end frameworks (e.g., Laravel))

Summary of previous lecture

1. Web Application Security(Implementing security mechanisms:
 - i. input validation(Definition, looked at measures, created `validateInput Function ()` to help with validation, `PASSWORD_DEFAULT` dynamically selects the most secure hashing algorithm),
 - ii. Sanitization(used `FILTER_SANITIZE_STRING`),
 - iii. prepared statements-)

Advanced Front-End Concepts – Web App Dev't Frameworks

Advanced Front-End Concepts – Web App Dev't Frameworks

Advanced front-end concepts in web application development often involve the use of frameworks and libraries that enable developers to build complex and interactive user interfaces efficiently.

Here are some advanced front-end concepts related to web application development frameworks:

Advanced Front-End Concepts – Web App Dev't Frameworks

1. Single Page Applications (SPA):

- i. **Concept:** SPAs load a single HTML page and dynamically update content as the user interacts with the application.
- ii. **Frameworks:** React, Angular, Vue.js.

Benefits: Enhanced user experience, faster navigation, and reduced server load.

2. Component-Based Architecture:

- i. **Concept:** UIs are built as a composition of independent, reusable components.
- ii. **Frameworks:** React, Angular, Vue.js.

Benefits: Encapsulation, reusability, and maintainability.

Advanced Front-End Concepts – Web App Dev't Frameworks

3. State Management:

- i. **Concept:** Managing and synchronizing the state of the application across components.
- ii. **Libraries:** Redux (for React), Vuex (for Vue.js), NgRx (for Angular).

Benefits: Predictable state changes, easier debugging, and improved scalability.

4. Server-Side Rendering (SSR) and Static Site Generation (SSG):

- i. **Concept:** Generating HTML on the server side or during the build process to improve initial page load performance.
- ii. **Frameworks:** Next.js (React), Nuxt.js (Vue.js).

Benefits: Faster time to first paint, improved SEO, and better user experience.

Advanced Front-End Concepts – Web App Dev't Frameworks

5. Progressive Web Apps (PWA):

- i. **Concept:** Web applications that provide a native app-like experience, including offline functionality, push notifications, and responsive design.
- ii. **Frameworks:** Workbox, Angular Service Worker, Preact.

Benefits: Improved performance, reliability, and engagement.

Etc.

Advanced Front-End Concepts – Web App Dev't Frameworks

6. Cross-Browser Compatibility and Testing:

- i. **Concept:** Ensuring consistent behavior across different web browsers.
- ii. **Tools:** Selenium, Browser Stack, Cross Browser Testing.

Benefits: Enhanced user experience, broader audience reach.

These advanced front-end concepts and frameworks empower developers to create modern, responsive, and feature-rich web applications. **Choosing the right framework depends** on the project requirements, team expertise, and specific goals for the application.

Advanced Front-End Concepts – Web App Dev't Frameworks

Web Apps Frameworks

Like discussed before, web application frameworks are pre-built libraries, tools, and templates that provide a structured way to develop web applications. They streamline the development process by offering a foundation for developers to build upon, reducing the need to write repetitive code. Here are some popular web application frameworks in various programming languages:

1. JavaScript Frameworks:
2. Ruby Frameworks
3. Java Frameworks
4. PHP Frameworks
5. C#/.NET Frameworks
6. Go Frameworks
7. Node.js Frameworks

Advanced Front-End Concepts – Web App Dev't Frameworks

Web Apps Frameworks +

1. JavaScript Frameworks:

- i. **React:** A JavaScript library for building user interfaces, particularly for single-page applications (SPAs). It's commonly used with other libraries and tools to create complete web applications.
- ii. **Angular:** A comprehensive front-end framework developed by Google. It provides a full set of tools for building dynamic and complex web applications.
- iii. **Vue.js:** A progressive JavaScript framework that is often compared to React and Angular. It is known for its simplicity and flexibility.

Advanced Front-End Concepts – Web App Dev't Frameworks

Web Apps Frameworks ++

2. Python Frameworks:

- i. **Django:** A high-level Python web framework that emphasizes rapid development, clean and pragmatic code, and the "batteries-included" philosophy, offering many built-in features.
- ii. **Flask:** A lightweight and micro web framework for Python. It provides the essentials for web development but allows developers to choose and integrate additional libraries as needed.
- iii. **FastAPI:** A modern Python web framework for building APIs quickly and efficiently. It is known for its speed and automatic generation of documentation.

Advanced Front-End Concepts – Web App Dev't Frameworks

Web Apps Frameworks +++

3. Ruby Frameworks:

- i. **Ruby on Rails (Rails):** A full-stack web framework for Ruby that follows the convention over configuration (CoC) and don't repeat yourself (DRY) principles. It is well-suited for building web applications rapidly.

Advanced Front-End Concepts – Web App Dev't Frameworks

Web Apps Frameworks +++++

4. Java Frameworks:

- i. **Spring Boot:** A Java-based framework that simplifies the development of Java applications, including web applications. It offers a wide range of modules and tools.
- ii. **JavaServer Faces (JSF):** A component-based Java web framework for building user interfaces for web applications. It is part of the Java EE (Enterprise Edition) platform.

Advanced Front-End Concepts – Web App Dev't Frameworks

Web Apps Frameworks++++++

6. C#/.NET Frameworks:

- i. **ASP.NET Core:** A cross-platform, high-performance framework for building modern, cloud-based, and internet-connected applications using C#.

7. Go Frameworks:

- i. **Gin:** A minimalistic and fast web framework for the Go programming language. It is designed for building efficient and scalable APIs.

Advanced Front-End Concepts – Web App Dev't Frameworks

Web Apps Frameworks ++++++

8. Node.js Frameworks:

- i. **Express.js:** A minimalist Node.js web application framework that provides a simple, unopinionated way to build web applications and APIs.
- ii. **NestJS:** A TypeScript-based framework for building scalable and maintainable server-side applications using Node.js. It is often used for building APIs.

In summary, we can say these are few examples of web application frameworks available in different programming languages. The choice of framework often depends on the *specific requirements of the project*, the programming language *expertise* of the development team, and other factors such as *scalability*, *performance*, and *community support*.

PHP Frameworks

Web Application Front-End Dev't – PHP Frameworks

PHP (Hypertext Preprocessor) is a popular server-side scripting language used for web development. It has a wide range of frameworks that simplify and expedite the web development process by providing pre-built libraries, tools, and conventions.

Here are some of the most popular PHP frameworks as of January 2022:

1. **Laravel:** Laravel is one of the most widely used PHP frameworks known for its elegant syntax and a wide range of features, including an expressive ORM (Object-Relational Mapping), routing, authentication, and templating.
2. **Symfony:** Symfony is a highly modular PHP framework with a set of reusable components and a strong emphasis on best practices. It's often used for large and complex applications.
3. **CodeIgniter:** CodeIgniter is known for its simplicity and a small footprint. It's an excellent choice for developers who prefer a lightweight framework without many pre-configured features.

Web Application Front-End Dev't – PHP Frameworks

Here are some of the most popular PHP frameworks as of January 2022:

4. **Yii:** Yii is a high-performance PHP framework that's suitable for developing both small and large applications. It includes many features, such as caching, security, and RESTful APIs.
5. **Zend Framework (Laminas Project):** Zend Framework has been rebranded as the Laminas Project. It offers a collection of PHP packages for building web applications, focusing on flexibility and extensibility.
6. **Phalcon:** Phalcon is unique because it's a PHP extension written in C. This makes it one of the fastest PHP frameworks available. It's particularly suitable for performance-critical applications.

Web Application Front-End Dev't – PHP Frameworks

Here are some of the most popular PHP frameworks as of January 2022:

7. **lim:** Slim is a micro-framework that's minimalistic and ideal for creating small web applications and RESTful APIs. It's known for its simplicity and speed.
8. **CakePHP:** CakePHP is a framework that follows the "convention over configuration" principle, making it easier to get started. It has built-in tools for handling authentication, security, and more.

Web Application Front-End Dev't – PHP Frameworks

Here are some of the most popular PHP frameworks as of January 2022:

9. **Lumen:** Lumen is a micro-framework developed by the creators of Laravel. It's designed for building fast, lightweight microservices and APIs.
10. **Flight:** Flight is another micro-framework that is extremely lightweight and easy to set up. It's suitable for small applications and RESTful services.
11. **Silex:** Silex is a micro-framework built on top of the Symfony components. It's known for its simplicity and is often used for small to medium-sized projects.

Web Application Front-End Dev't – PHP Frameworks-Laravel

Web Application Front-End Dev't – PHP Frameworks-Laravel

Laravel is a web application framework with expressive, elegant syntax. A web framework provides a structure and starting point for creating your application, allowing you to focus on creating something amazing while details are worked out for you.

Laravel strives to provide an amazing developer experience while providing powerful features such as thorough dependency injection, an expressive database abstraction layer, queues and scheduled jobs, unit and integration testing, and more.

Whether you are new to PHP web frameworks or have years of experience, Laravel is a framework that can grow with you. This lecture introduces and help you take your first steps as a web developer or give you a boost as you take your expertise to the next level. I can't wait to see what you build. Laravel. (n.d.).

Web Application Front-End Dev't – PHP Frameworks-Laravel

Why Laravel?

There are a variety of tools and frameworks available to you when building a web application. However, we believe Laravel is the best choice for building modern, full-stack web applications

1. A Progressive Framework

Laravel is a "progressive" framework. That means Laravel grows with you. If you're just taking your first steps into web development, Laravel's vast library of documentation, guides, and video tutorials online will help you learn the ropes without becoming overwhelmed.

Web Application Front-End Dev't –

PHP Frameworks-Laravel

Why Laravel?

2. A Scalable Framework

Laravel is incredibly scalable. Thanks to the scaling-friendly nature of PHP and Laravel's built-in support for fast, distributed cache systems like Redis, horizontal scaling with Laravel is a breeze.

In fact, Laravel applications have been easily scaled to handle hundreds of millions of requests per month.

Web Application Front-End Dev't – PHP Frameworks-Laravel

Why Laravel?

3. A Community Framework

Laravel combines the best packages in the PHP ecosystem to offer the most robust and developer friendly framework available.

In addition, thousands of talented developers from around the world have contributed to the framework. Who knows, maybe you'll even become a Laravel contributor.

Setting up Laravel

PHP Frameworks - Setting up Laravel

Setting up Laravel involves several steps.

Here's a basic guide on how to set up Laravel on your local development environment

1. Install Composer:

- i. If you don't already have Composer installed, download and install it. Composer is a PHP dependency management tool that Laravel relies on.
- ii. You can download Composer from getcomposer.org, and installation instructions can be found on the website as well.

2. Install Laravel via Composer:

Once Composer is installed, open your command-line interface and run the following command to create a new Laravel project:

```
composer global require laravel/installer
```

PHP Frameworks - Setting up Laravel

Setting up Laravel involves several steps.

3. Create and Migrate the Database plus tables:

Laravel uses migrations to create and manage your database schema. Run the following command to execute the pending migrations:

```
php artisan migrate
```

4. Start the Development Server:

To start a development server and access your Laravel application in your web browser, run the following command after changing the directory to project directory:

```
php artisan serve
```

This will start a development server on <http://localhost:8000>. You can access your Laravel application in your web browser by navigating to this address.

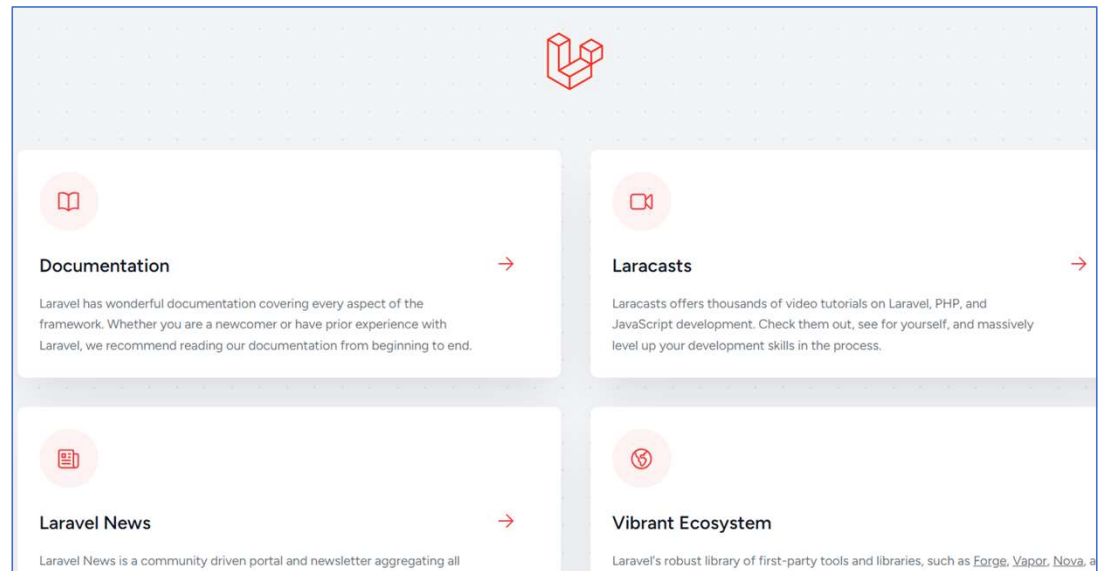
PHP Frameworks - Setting up Laravel

Setting up Laravel involves several steps.

5. Access the Application:

Open your web browser and go to <http://localhost:8000> or the URL you've configured in your web server. You should see the default Laravel welcome page.

Lets now follow the above steps one at a time to create our first web application project based on PHP Framework called Laravel.



Web Application Front-End Dev't – PHP Frameworks-Installing Composer

Step 1. Installation of composer on Windows#

Composer is a tool for dependency management in PHP. It allows you to declare the libraries your project depends on and it will manage (install/update) them for you.

Dependency management#

Composer is not a package manager in the same sense as Yum or Apt are. Yes, it deals with "packages" or libraries, but it manages them on a per-project basis, installing them in a directory (e.g. vendor) inside your project. By default, it does not install anything globally. Thus, it is a dependency manager. It does however support a "global" project for convenience via the global command. Laravel. (n.d.).

This idea is not new and Composer is strongly inspired by node's npm and ruby's bundler.

Web Application Front-End Dev't – PHP Frameworks

Step 1. Installation of composer on Windows#

Suppose:

1. You have a project that depends on a number of libraries.
2. Some of those libraries depend on other libraries.

Composer:

1. Enables you to declare the libraries you depend on.
2. Finds out which versions of which packages can and need to be installed, and installs them (meaning it downloads them into your project).
3. You can update all your dependencies in one command.

Web Application Front-End Dev't – PHP Frameworks

Step 1. Installation of composer on Windows#

Composer in its latest version requires PHP 7.2.5 to run. A long-term-support version (2.2.x) still offers support for PHP 5.3.2+ in case you are stuck with a legacy PHP version. A few sensitive php settings and compile flags are also required, but when using the installer you will be warned about any incompatibilities.

Composer needs several supporting applications to work effectively, making the process of handling package dependencies more efficient. For decompressing files, Composer relies on tools like 7z (or 7zz), gzip, tar, unrar, unzip and xz.

As for version control systems, Composer integrates seamlessly with Fossil, Git, Mercurial, Perforce and Subversion, thereby ensuring the application's smooth operation and management of library repositories. Before using Composer, ensure that these dependencies are correctly installed on your system.(<https://getcomposer.org/doc/00-intro.md>), Laravel. (n.d.).

Web Application Front-End Dev't – PHP Frameworks

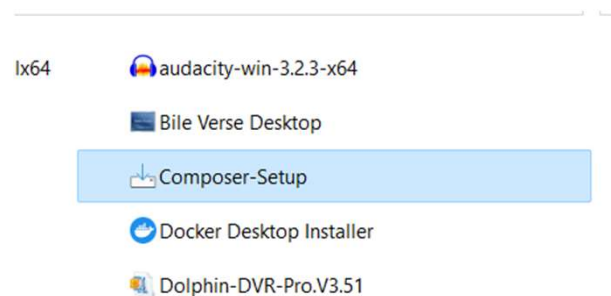
Step 1. Installation of composer on Windows Using the Installer#

This is the easiest way to get Composer set up on your machine.

Download and run [Composer-Setup.exe](#). It will install the latest Composer version and set up your PATH so that you can call composer from any directory in your command line.

Steps.

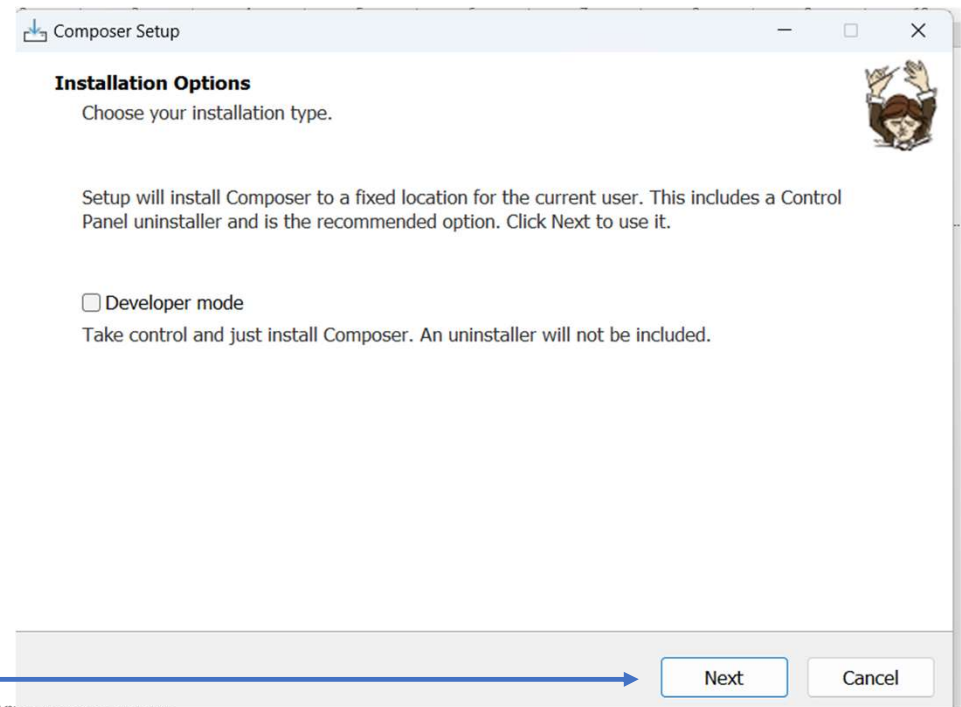
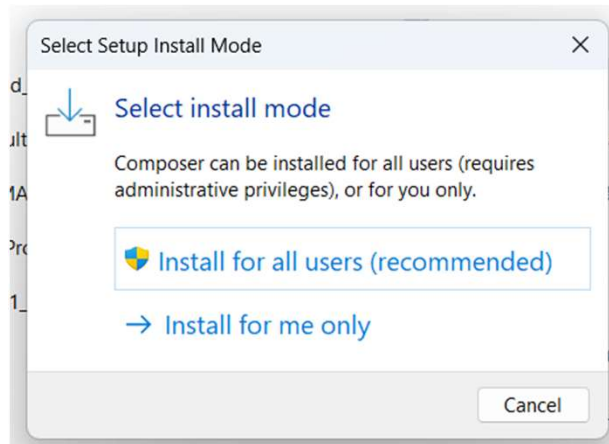
1. Open the downloaded file



Web Application Front-End Dev't – PHP Frameworks

Steps.

2. Choose whether to install for all computer users or current users only

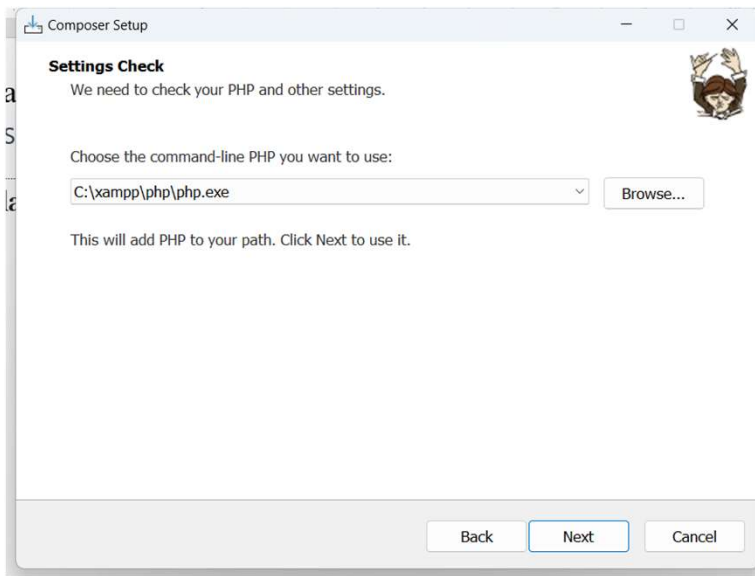


3. Click Next

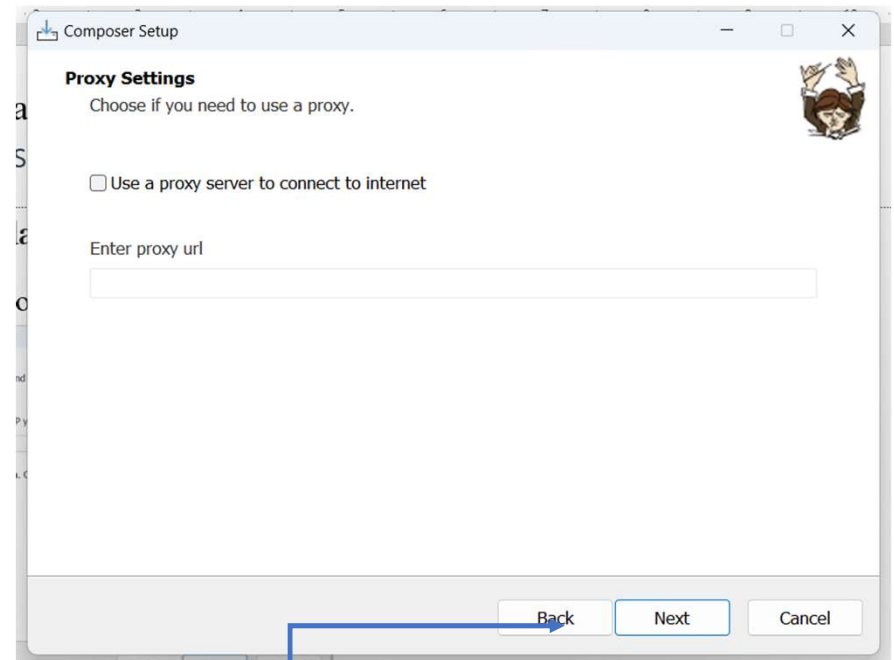
Web Application Front-End Dev't – PHP Frameworks

Step 1. Installation of composer on Windows Using the Installer#

4. Choose the location to install composer then click Next



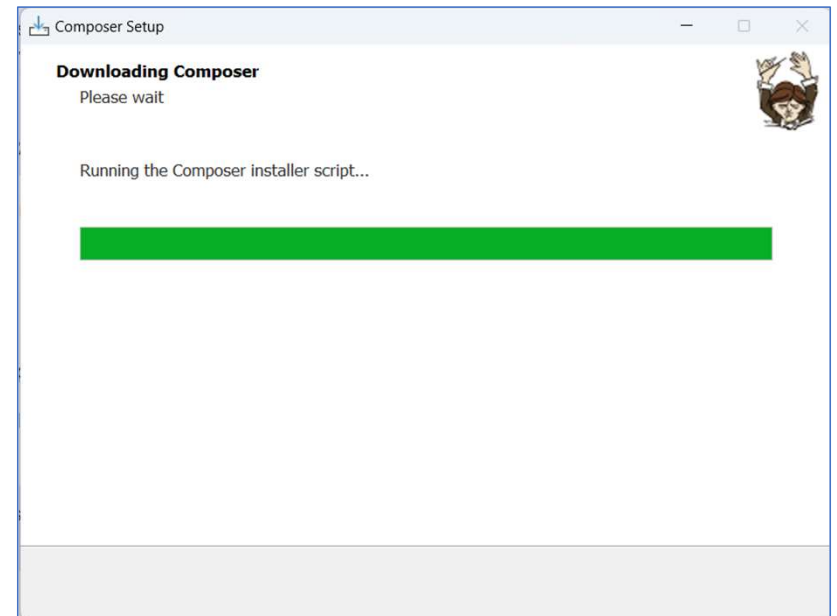
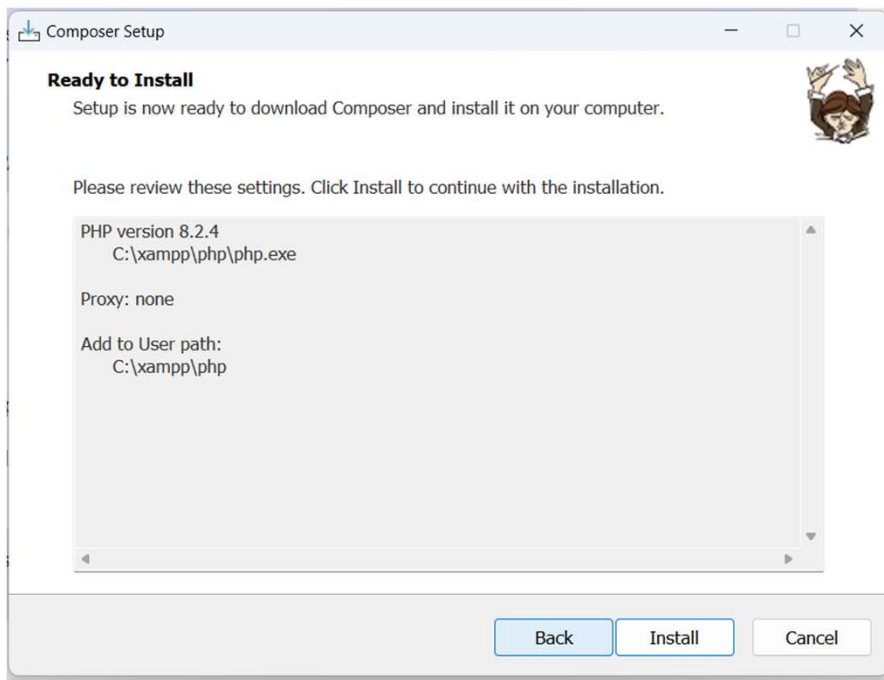
5. Choose if you want use the proxy then click Next



Web Application Front-End Dev't – PHP Frameworks

Step 1. Installation of composer on Windows Using the Installer#

6. Click install button under Ready to install section

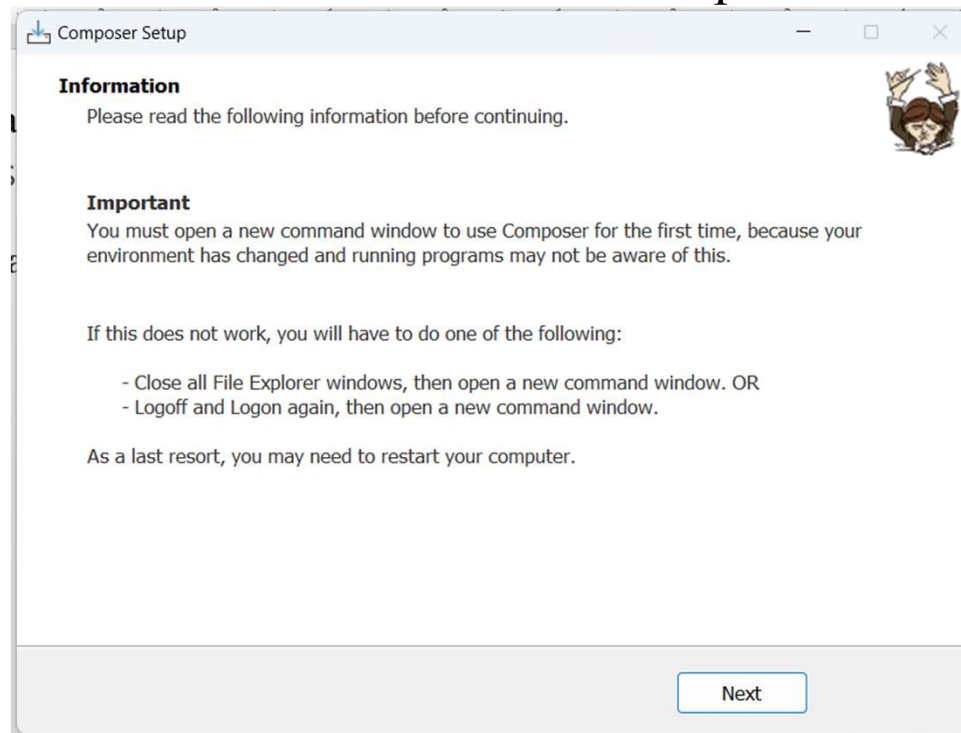


You will see this window for a while

Web Application Front-End Dev't – PHP Frameworks

Step 1. Installation of composer on Windows#

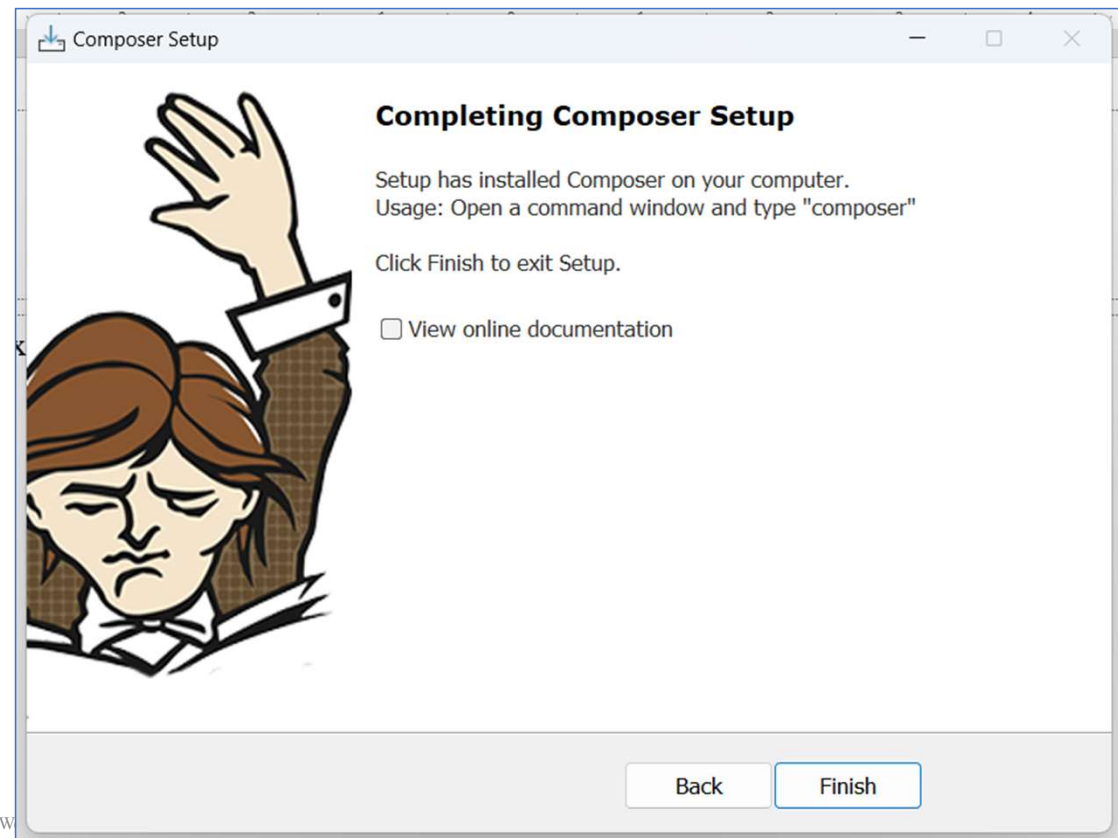
7. Read important information about how to run composer
8. Click Next



Web Application Front-End Dev't – PHP Frameworks

Step 1. Installation of composer on Windows#

9. Now hit Finish button

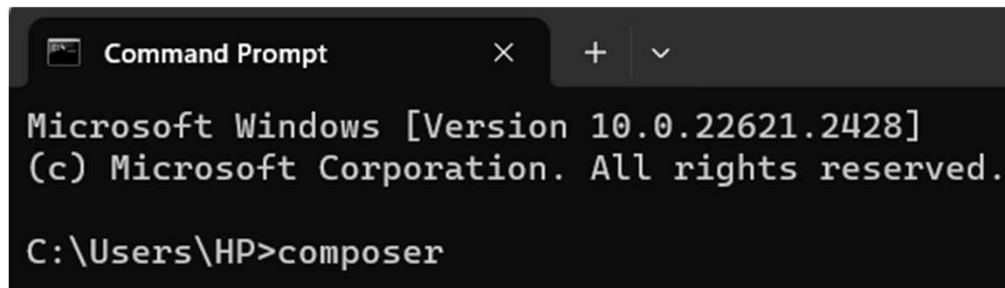


Web Application Front-End Dev't – PHP Frameworks

Using Composer#

Now that we've installed Composer, you are ready to use it! Head on over to the next to install some packages e.g

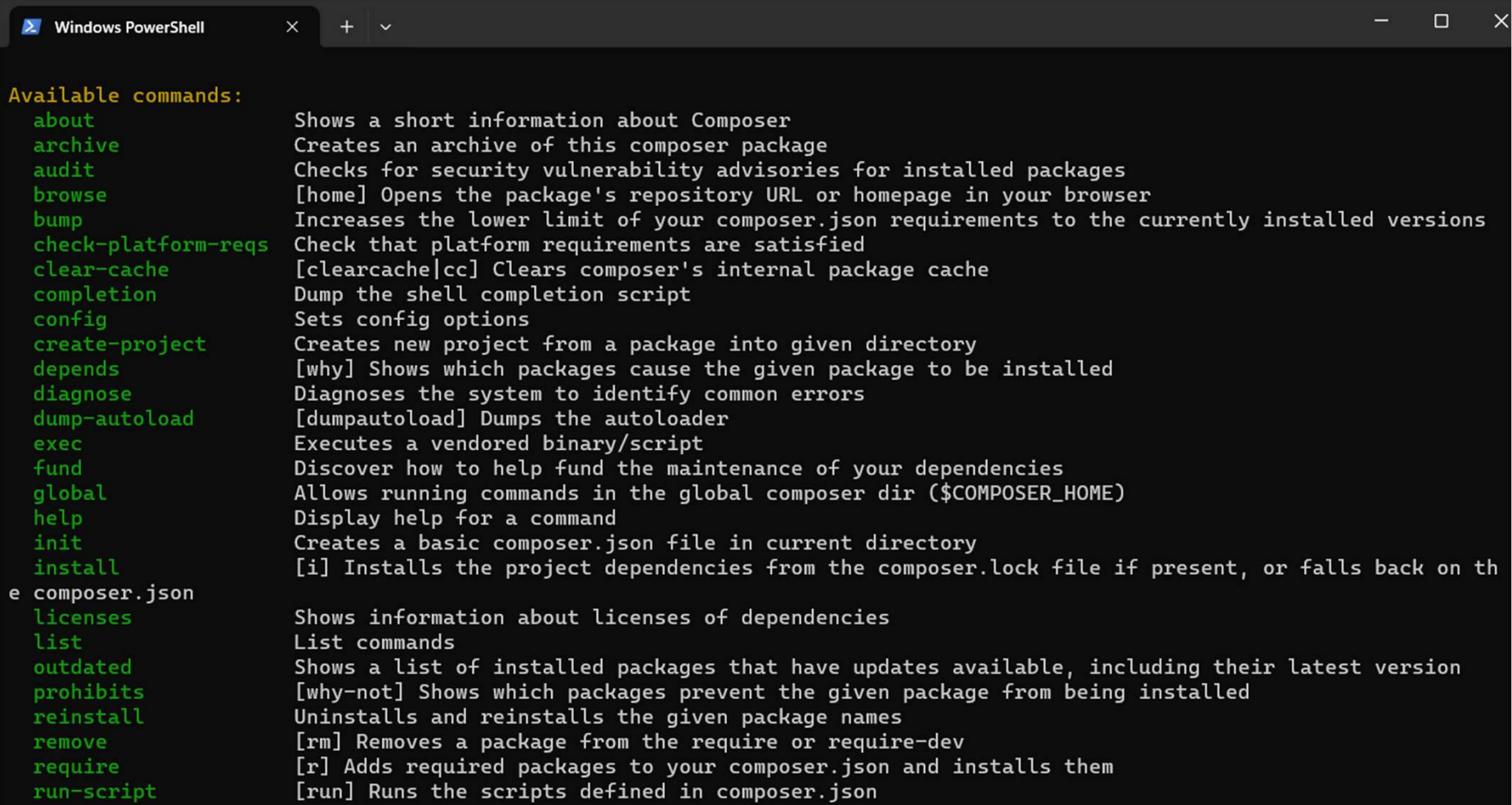
1. First verify that composer is actually installed.
2. Open your terminal/cmd
3. Type: `COMPOSER` then press Enter Key



```
Command Prompt
Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.
C:\Users\HP>composer
```


Web Applications Front-End Dev't –

PHP Frameworks:



```
Windows PowerShell

Available commands:
about Shows a short information about Composer
archive Creates an archive of this composer package
audit Checks for security vulnerability advisories for installed packages
browse [home] Opens the package's repository URL or homepage in your browser
bump Increases the lower limit of your composer.json requirements to the currently installed versions
check-platform-reqs Check that platform requirements are satisfied
clear-cache [clearcache|cc] Clears composer's internal package cache
completion Dump the shell completion script
config Sets config options
create-project Creates new project from a package into given directory
depends [why] Shows which packages cause the given package to be installed
diagnose Diagnoses the system to identify common errors
dump-autoload [dumpautoload] Dumps the autoloader
exec Executes a vendored binary/script
fund Discover how to help fund the maintenance of your dependencies
global Allows running commands in the global composer dir ($COMPOSER_HOME)
help Display help for a command
init Creates a basic composer.json file in current directory
install [i] Installs the project dependencies from the composer.lock file if present, or falls back on th
e composer.json
licenses Shows information about licenses of dependencies
list List commands
outdated Shows a list of installed packages that have updates available, including their latest version
prohibits [why-not] Shows which packages prevent the given package from being installed
reinstall Uninstalls and reinstalls the given package names
remove [rm] Removes a package from the require or require-dev
require [r] Adds required packages to your composer.json and installs them
run-script [run] Runs the scripts defined in composer.json
```

More on Laravel

Installing Laravel via composer

Before creating your first Laravel project, you should ensure that your local machine has PHP and Composer installed.

After you have installed PHP and Composer, you may now install Laravel using the command line code below:

```
composer global require laravel/installer
```

Enter the code above in you Terminal

```
Install the latest PowerShell for new features and improvements! https://aka.ms/PowerShell-installation

Warning: PowerShell detected that you might be using a screen reader and a screen reader is not enabled. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS C:\Users\HP> composer global require laravel/installer
Changed current directory to C:/Users/HP/AppData/Roaming/Composer
```

More on Laravel

Installing Laravel via composer

You may encounter the following error during the installation of composer due to commented zip extension in the php.ini file.

Loading composer repositories with package information Updating dependencies (including require-dev) Failed to download psr/log from dist: The zip extension and unzip command are both missing, skipping. The php.ini used by your command-line PHP is: /etc/php/7.0/cli/php.ini Now trying to download from source

Solution.

For Windows, in case you're using a locale development environment like XAMPP, just locate the php.ini file (C:\xampp\php in my case), open it and remove the semicolon from the following line.

`;extension=zip` → `extension=zip`

More on Laravel

Installing Laravel via composer

```
Windows PowerShell
ses. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS C:\Users\HP> composer global require laravel/installer
Changed current directory to C:/Users/HP/AppData/Roaming/Composer
./composer.json has been created
Running composer update laravel/installer
Loading composer repositories with package information
Updating dependencies
Lock file operations: 27 installs, 0 updates, 0 removals
- Locking doctrine/inferno (2.0.8)
- Locking illuminate/collections (v10.31.0)
- Locking illuminate/conditionable (v10.31.0)
- Locking illuminate/contracts (v10.31.0)
- Locking illuminate/filesystem (v10.31.0)
- Locking illuminate/macroable (v10.31.0)
- Locking illuminate/support (v10.31.0)
- Locking laravel/installer (v5.1.3)
- Locking laravel/prompts (v0.1.13)
- Locking nesbot/carbon (2.71.0)
- Locking psr/clock (1.0.0)
- Locking psr/container (2.0.2)
- Locking psr/simple-cache (3.0.0)
- Locking symfony/console (v6.3.4)
- Locking symfony/deprecation-contracts (v3.3.0)
- Locking symfony/finder (v6.3.5)
- Locking symfony/polyfill-ctype (v1.28.0)
- Locking symfony/polyfill-intl-grapheme (v1.28.0)
- Locking symfony/polyfill-intl-normalizer (v1.28.0)
- Locking symfony/polyfill-mbstring (v1.28.0)
- Locking symfony/polyfill-php80 (v1.28.0)
```

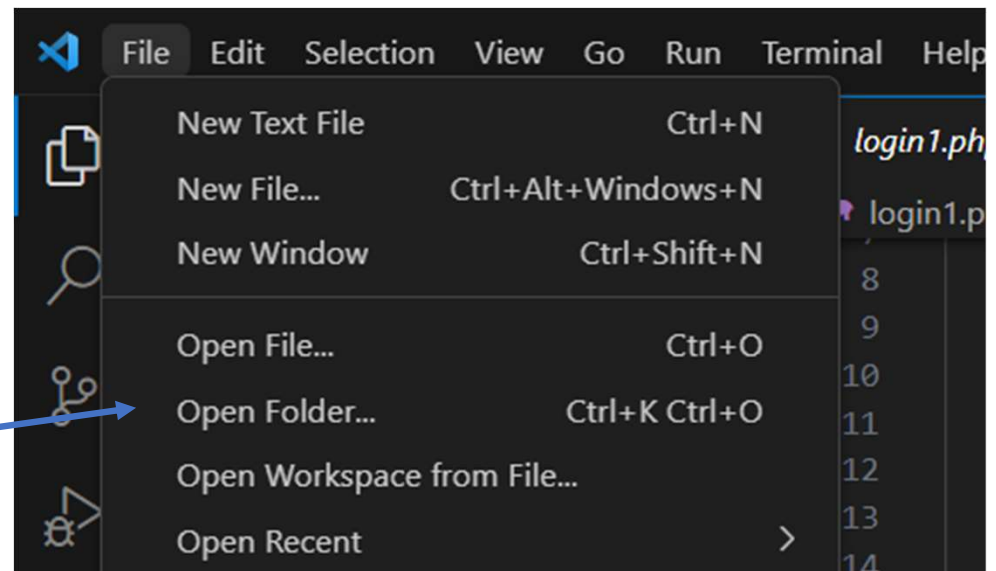
```
- Installing symfony/polyfill-intl-grapheme (v1.28.0): Cloning 875e90aeea from cache
- Installing symfony/polyfill-ctype (v1.28.0): Cloning ea208ce43c from cache
- Installing symfony/string (v6.3.5): Cloning 13d76d0fb0 from cache
- Installing symfony/service-contracts (v3.3.0): Cloning 40da9cc13e from cache
- Installing symfony/deprecation-contracts (v3.3.0): Cloning 7c3aff79d1 from cache
- Installing symfony/console (v6.3.4): Cloning eca495f2ee from cache
- Installing illuminate/collections (v10.31.0): Cloning bb8784ce91 from cache
- Installing laravel/prompts (v0.1.13): Cloning e1379d8ead from cache
- Installing voku/portable-ascii (2.0.1): Cloning b56450eed2 from cache
- Installing symfony/translation-contracts (v3.3.0): Cloning 02c24deb35 from cache
- Installing symfony/translation (v6.3.7): Cloning 30212e7c87 from cache
- Installing symfony/polyfill-php80 (v1.28.0): Cloning 6caa57379c from cache
- Installing psr/clock (1.0.0): Cloning e41a24703d from cache
- Installing nesbot/carbon (2.71.0): Cloning 9827623318 from cache
- Installing illuminate/support (v10.31.0): Cloning 7c28b263a1 from cache
- Installing symfony/finder (v6.3.5): Cloning alb31d88c0 from cache
- Installing illuminate/filesystem (v10.31.0): Cloning 4dfff993a0d from cache
- Installing laravel/installer (v5.1.3): Cloning 7547b807bb from cache
18 package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating autoload files
16 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
```

Installing Laravel into your preferred directory

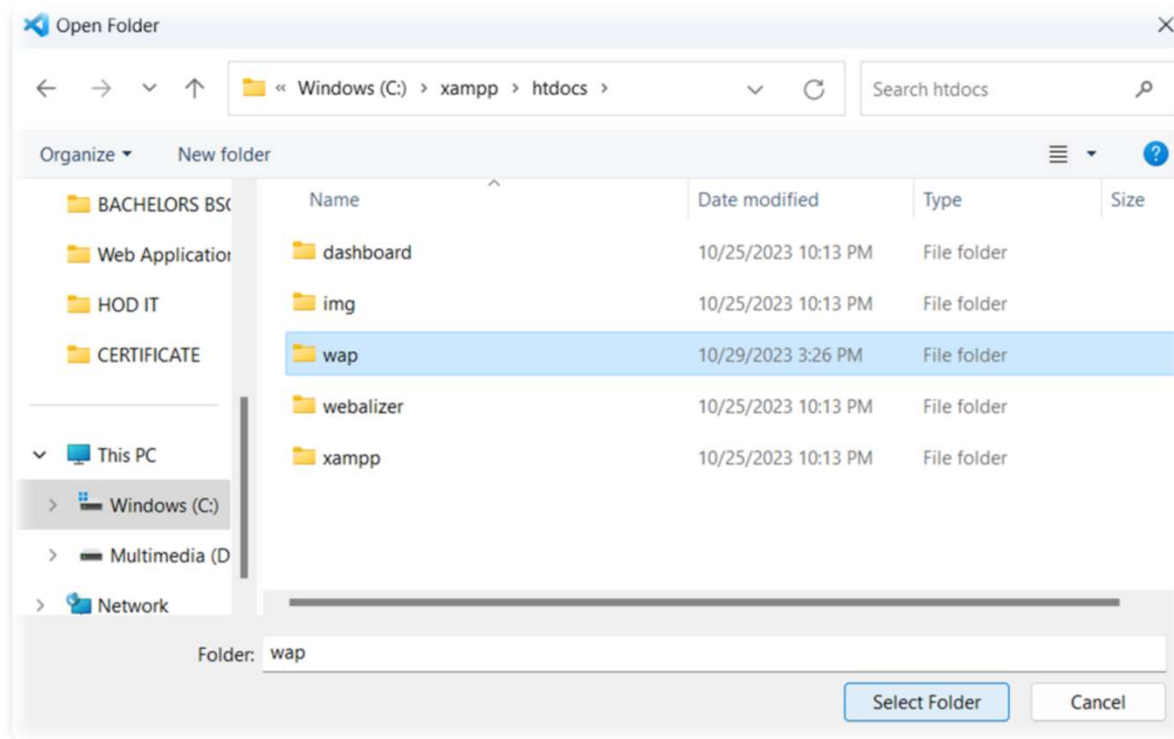
Check if Laravel can be accessed through your previous system development directory e.g wap.

Steps

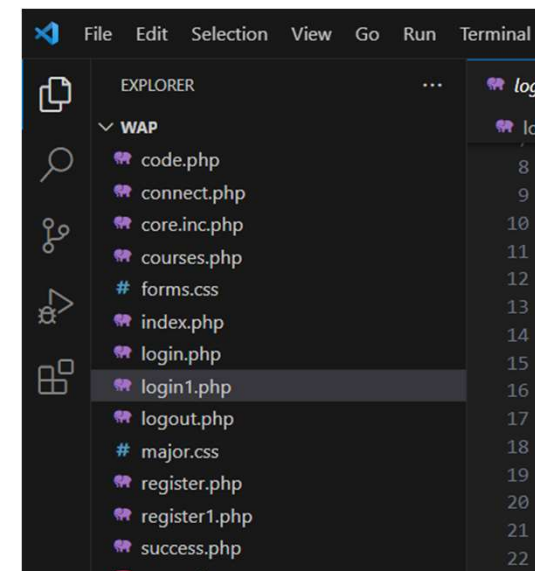
1. Open Visual code
2. Click File Menu
3. Select Open folder-> Navigate to your directory e.g. C:\xampp\htdocs\wap
4. Click select Folder



Installing Laravel into your preferred directory

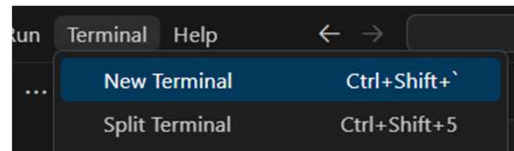


After clicking Select Folder, you will see a list of your previous files. E.g

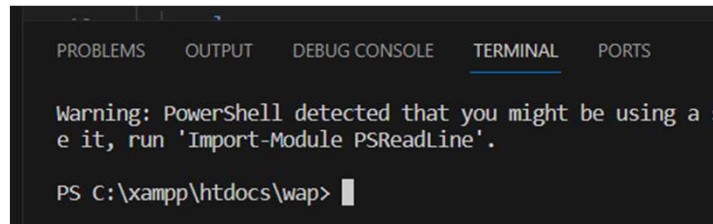


Installing Laravel into your preferred directory

5. Now open new Terminal in Visual Code



6. You should see a new Terminal window as below

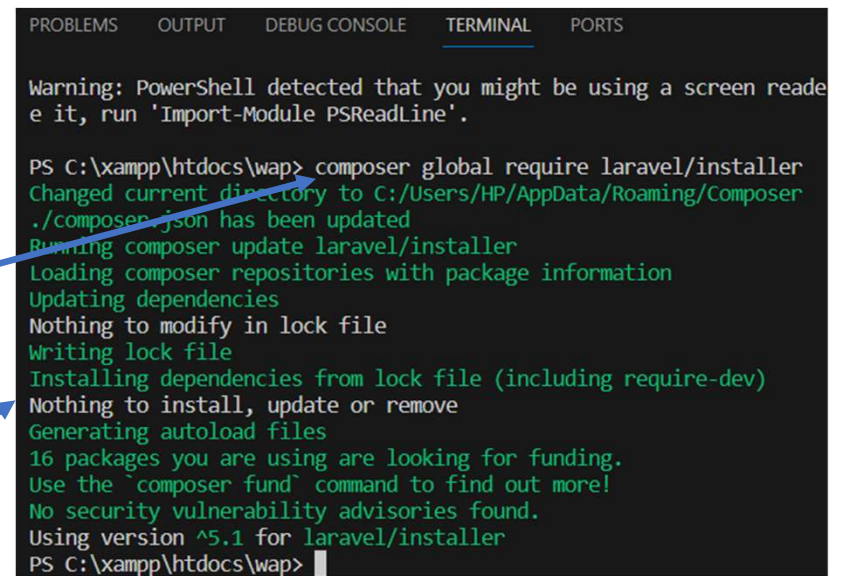


7. Now enter Laravel installation code

```
composer global require laravel/installer
```

Since Laravel was already installed, you should see statements like Nothing to install, update or remove

Web Application Programming



More on Laravel

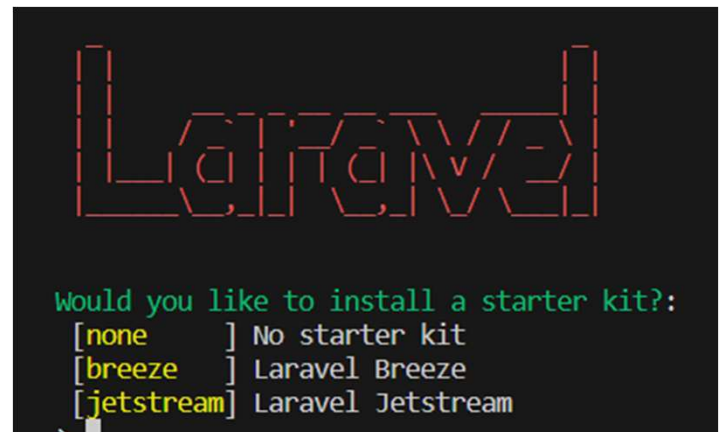
Create First Laravel Project

Now that you have confirmed Laravel is installed, try to create new project.

Steps

1. Enter `Laravel new wap` into your terminal
2. For the first time, you will see the following list of startup kits(Laravel Breeze or Laravel Jetstream). Choose to install Laravel Jetstream.

```
PS C:\xampp\htdocs\wap> laravel new wap
```



```
Laravel

Would you like to install a starter kit?:
[none] ] No starter kit
[breeze] ] Laravel Breeze
[jetstream] ] Laravel Jetstream
```

More on Laravel

Create First Laravel Project

Installing Jetstream startup kit and setting it up.

```
Would you like to install a starter kit?:  
[none    ] No starter kit  
[breeze  ] Laravel Breeze  
[jetstream] Laravel Jetstream  
> Laravel Jetstream
```

Why should one select either Jetstream or Breeze startup kit?

Lets look at the difference between the two.

More on Laravel

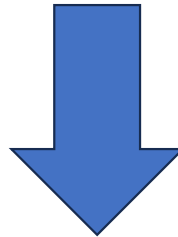
Create First Laravel Project

Difference between Laravel Jetstream and Laravel Breeze

Laravel Jetstream and Laravel Breeze are both **starter kits** provided by the Laravel framework to help one kickstart web application development.

They provide a foundation with user authentication, registration, and other common features, allowing you to focus on building your application. However, there are some

key differences between the two:



More on Laravel

Create First Laravel Project

Difference between Laravel Jetstream and Laravel Breeze

Laravel Jetstream:

- 1. More Features:** Jetstream is a more feature-rich starter kit. It includes features for team management, profile photos, API support, and more out of the box.
- 2. Tailwind CSS:** Jetstream uses the Tailwind CSS framework for styling, which is highly customizable and allows for more flexibility in terms of design.
- 3. Inertia.js and Livewire:** Jetstream offers both Inertia.js and Livewire as front-end stack options. Inertia.js is a modern approach to building server-driven web applications, while Livewire provides real-time interactions with the server.

More on Laravel

Create First Laravel Project

Difference between Laravel Jetstream and Laravel Breeze

Laravel Jetstream:

4. **Two Frontend Stacks:** You can choose between two different frontend stacks when setting up Jetstream: Livewire (with Blade views) or Inertia.js (with Vue.js and React). This provides flexibility in how you build your application's user interface.
5. **Team Management:** Jetstream includes team management features out of the box, which can be useful for applications that require collaboration and multiple user roles.
6. **Fortify Integration:** Jetstream builds on top of Laravel Fortify, a robust authentication and registration package.

More on Laravel

Create First Laravel Project

Difference between Laravel Jetstream and Laravel Breeze

Laravel Breeze

- 1.**Simplicity:** Breeze is a simpler, lightweight starter kit focused primarily on providing the essentials for user authentication and registration.
- 2.**Blade Views:** Breeze uses traditional Blade views for the user interface.
- 3.**Minimal Configuration:** It is designed to be minimal and easier to set up, making it a good choice if you want a basic authentication system without extra features.
- 4.**Frontend Stack:** Breeze uses Blade views for the user interface and can be easily extended with JavaScript for interactive features.

More on Laravel

Create First Laravel Project

Difference between Laravel Jetstream and Laravel Breeze

Laravel Breeze

4. **Laravel Sanctum:** Breeze includes Laravel Sanctum for API authentication. While it's a capable option, Jetstream offers more choices for API support.

In summary, the main difference between Jetstream and Breeze is their complexity and the range of features they provide. Jetstream is more feature-rich, flexible, and suitable for larger applications, especially when you need team management and are open to using Tailwind CSS with Inertia.js or Livewire. Breeze, on the other hand, is simpler and more lightweight, making it a great choice for smaller projects that require basic user authentication and registration without additional features.

Your choice between Jetstream and Breeze will depend on the specific requirements of your project and your familiarity with the tools and technologies they use.

More on Laravel

Create First Laravel Project

Installing Jetstream startup kit and setting it up. Having know why we selected Jetstream in this case, we now need to install its stack. It may be livewire or inertia.

```
Which Jetstream stack would you like to install?:  
[livewire] Livewire  
[inertia ] Vue with Inertia  
> Livewire
```

```
Would you like any optional features? [None]:  
[none      ] None  
[api       ] API support  
[dark      ] Dark mode  
[verification] Email verification  
[teams     ] Team support  
> api
```

```
Which testing framework do you prefer?:  
[0] PHPUnit  
[1] Pest  
> 0
```

```
Would you like to initialize a Git repository? (yes/no) [no]:  
> yes  
  
Creating a "laravel/laravel" project at "./wap"  
Installing laravel/laravel (v10.2.8)  
Failed to download laravel/laravel from dist: The zip extension and unzip  
The php.ini used by your command-line PHP is: C:\xampp\php\php.ini  
Now trying to download from source  
- Syncing laravel/laravel (v10.2.8) into cache  
  - Installing laravel/laravel (v10.2.8): Cloning 024c86a24b from cache  
Created project in C:\xampp\htdocs\wap\wap  
> @php -r "file_exists('.env') || copy('.env.example', '.env');"  
Loading composer repositories with package information  
Updating dependencies  
Lock file operations: 110 installs, 0 updates, 0 removals  
- Locking brick/math (0.11.0)  
- Locking dflydev/dot-access-data (v3.0.2)  
- Locking doctrine/inflector (2.0.8)  
- Locking doctrine/lexer (3.0.0)  
- Locking dragonmantank/cron-expression (v3.3.3)  
- Locking egulias/email-validator (4.0.2)  
- Locking fakerphp/faker (v1.23.0)  
- Locking filp/whoops (2.15.4)
```

More on Laravel

Create First Laravel Project

Select the database of you project

```
Which database will your application use? [MySQL]:  
[mysql ] MySQL  
[pgsql ] PostgreSQL  
[sqlite] SQLite  
[sqlsrv] SQL Server  
> mysql
```

Now wait until project creating is complete

More on Laravel

Create First Laravel Project

Run the first app

Make sure you are in the project folder

If not change your directory to that folder e.g PS

```
C:\xampp\htdocs\wap> cd wap
```

```
PS C:\xampp\htdocs\wap> cd wap
```

Then type : php artisan serve

e.g. PS C:\xampp\htdocs\wap\wap> php artisan serve

```
PS C:\xampp\htdocs\wap\wap> php artisan serve
```

You will see:

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

Copy the link above and run on your browser.

```
PS C:\xampp\htdocs\wap\wap1> Follow link \(ctrl + click\)  
INFO Server running on [http://127.0.0.1:8000].  
Press Ctrl+C to stop the server
```

More on Laravel

Create First Laravel Project

You may get the following error messages because we have not yet created the database called wap

```
Illuminate \ Database \ QueryException  
SQLSTATE[HY000] [1049] Unknown database 'wap'  
SELECT * FROM `sessions` WHERE `id` = BnMNB4G1pRxVb8NP3vV
```

To solve the problem

You have to clear the cache like that (because your old configuration is in you cache file) :

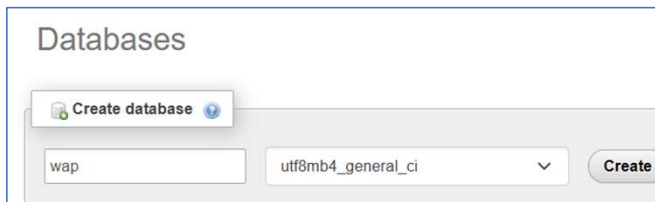
```
PS C:\xampp\htdocs\wap> cd wap  
PS C:\xampp\htdocs\wap\wap> php artisan cache:clear
```

More on Laravel

Create First Laravel Project

After clearing the cache, now create your database called wap or any other name through phpMyAdmin

2. Create your related Database.



3. Now run `php artisan migrate:install`

```
PS C:\xampp\htdocs\wap\wap> php artisan migrate:install
```

4. Finally run `php artisan migrate`

```
PS C:\xampp\htdocs\wap\wap> php artisan migrate
```

More on Laravel

Create First Laravel Project

After creating the database, you will need to migrate the default table called sessions to your newly created database if not you may get the error below.

```
Illuminate \ Database \ QueryException PHP 8.2.4 10.31.0  
  
SQLSTATE[42S02]: Base table or view not found: 1146 Table 'kumu.sessions'  
doesn't exist  
  
SELECT * FROM `sessions` WHERE `id` = pIjtVjy7m6fbpe1dABdKftmuv4ENTgbBHajIHGh4 limit 1
```

Therefore use the following code migrate table by: `php artisan session:table`

```
PS C:\xampp\htdocs\wap\wap> php artisan session:table
```

More on Laravel

Create First Laravel Project

Run the first app

Having successfully taking the steps above, you can try running the project again.

Then type: `php artisan serve`

e.g. `PS C:\xampp\htdocs\wap\wap> php artisan serve`

```
PS C:\xampp\htdocs\wap> cd wap
```

You will see:

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

```
PS C:\xampp\htdocs\wap\wap> php artisan serve
```

Copy the link above and run on your browser.

```
PS C:\xampp\htdocs\wap\wap1> Follow link \(ctrl + click\)
```

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

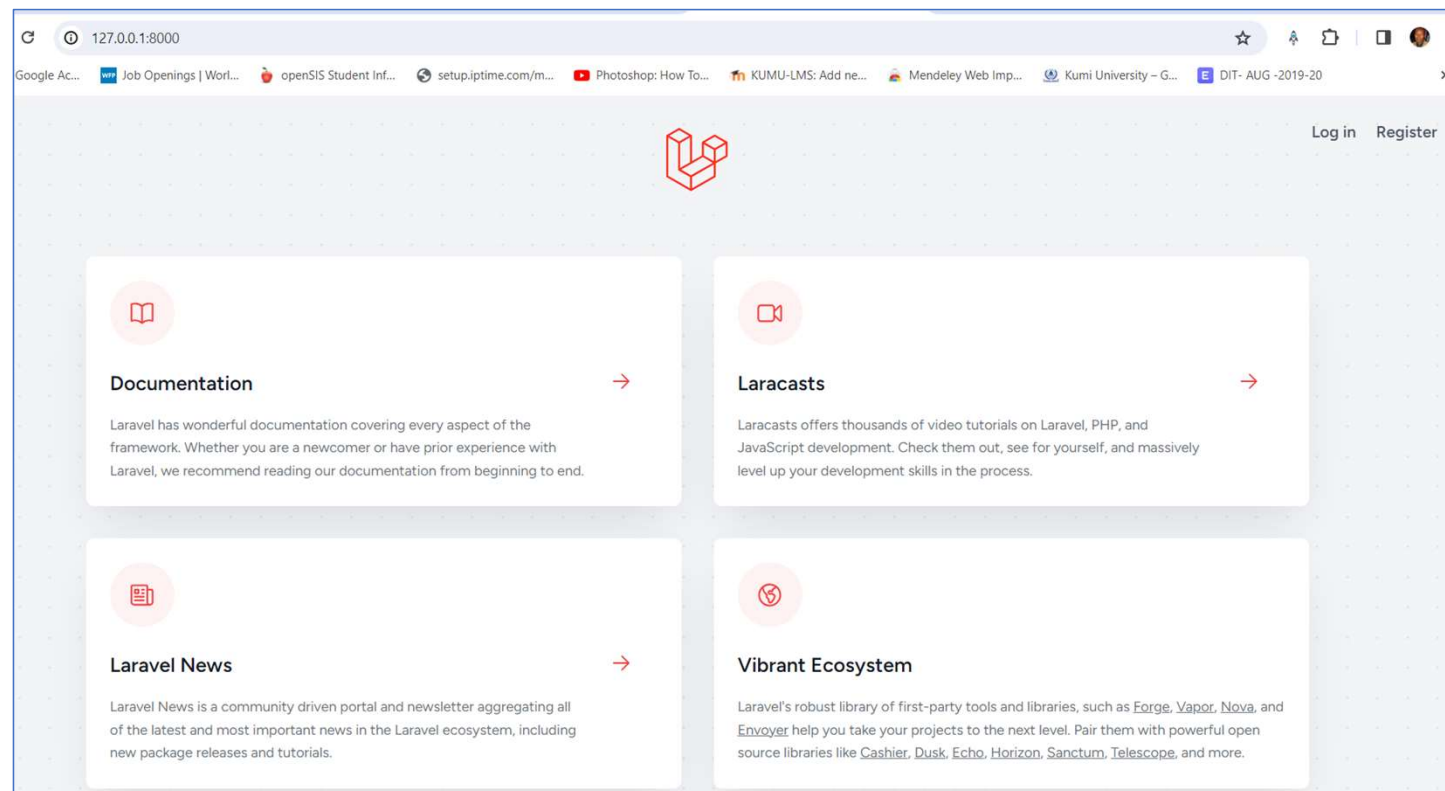
```
Press Ctrl+C to stop the server
```

More on Laravel

Create First Laravel Project

Run the first app

On successful run, you should be able to see the display below on your browser.

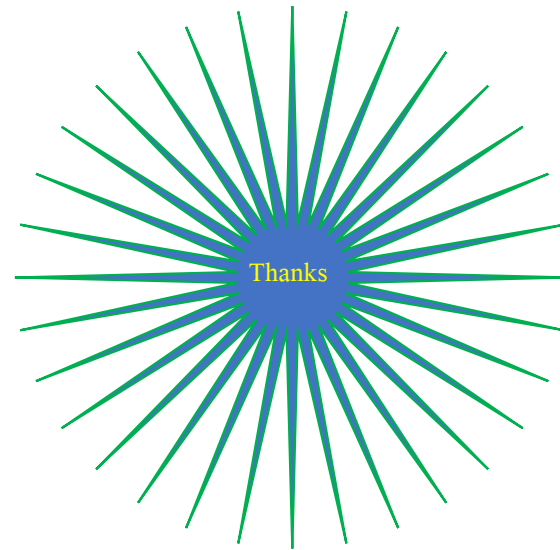


Summary

Summary

1. Advanced Front-End Concepts(
 - i. Intro, recap on web application dev't frameworks,
 - ii. Narrowed down to Laravel,
 - iii. looked Laravel installation steps,
 - iv. did the installation right from composer, Laravel, to creating Laravel Application project etc.)

Thank you for
Listening



References

The PHP framework for web artisans. Laravel. (n.d.). <https://laravel.com/docs/10.x#getting-started-on-windows>