Introduction of Laravel and creating a customized framework out of its packages

s.farshad.k@gmail.com

Presents By : Sayed-Farshad Kazemi

For : Software Free Day

Fall 2016

@farshadfelix

@s.farshad.k

@farshad92

Headlines

- **Why should we use frameworks?**
- **Why Laravel?**
- **How to install**
  - A. FIRST THING FIRST: XAMPP
  - B. WHAT IS DEPENDENCY MANAGER?
  - C. INSTALL COMPOSER
  - D. INSTALL LARAVEL
- **HOW TO KEEP WHAT WE NEED?**
  - A. HOW TO USE WHAT WE NEED?
  - B. WHAT MAY WE REALLY NEED?
  - C. LET'S TRY IT OUT!
  - D. DISCUSSION : IS IT BETTER TO WRITE A FRAMEWORK FROM SCRATCH
- **CONCLUSION**

Headlines

- **Why should we use frameworks?**
- Why Laravel?
- How to install
  - A. FIRST THING FIRST: XAMPP
  - B. WHAT IS DEPENDENCY MANAGER?
  - C. INSTALL COMPOSER
  - D. INSTALL LARAVEL
- HOW TO KEEP WHAT WE NEED?
  - A. HOW TO USE WHAT WE NEED?
  - B. WHAT MAY WE REALLY NEED?
  - C. LET'S TRY IT OUT!
  - D. DISCUSSION : IS IT BETTER TO WRITE A FRAMEWORK FROM SCRATCH
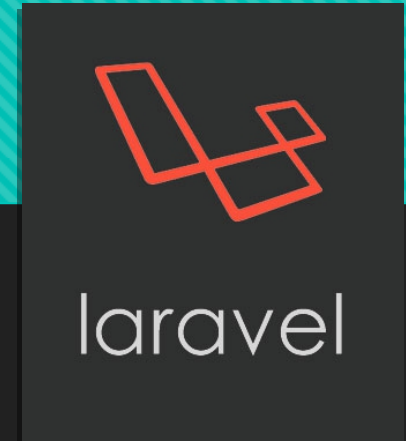- CONCLUSION

Why should we use frameworks?

○ Reinventing the wheel?

 ○ Can get a great wheel, for free!

 ○ No time will be put on inventing wheel

 ○ You get a reliable wheel, in terms of privacy, security, performance and etc.

○ Security matters

○ Time is Gold

○ Performance Talks

○ Standard coding style

4

Headlines

Why Laravel?

- Ease of Use
  - Great community
  - So many tutorials out there in the web
  - Easy to install
- MVC based framework, yet easy to understand
- Uses separable packages to form a framework
  - Can use whatever you wanted, wherever you wanted
- Very good performance
- REST routing
- ...

Headlines

- Why should we use frameworks?
- Why Laravel?
- **How to install**
  - A. **FIRST THING FIRST: XAMPP**
  - B. **WHAT IS DEPENDENCY MANAGER?**
  - C. **INSTALL COMPOSER**
  - D. **INSTALL LARAVEL**
- HOW TO KEEP WHAT WE NEED?
  - A. HOW TO USE WHAT WE NEED?
  - B. WHAT MAY WE REALLY NEED?
  - C. LET'S TRY IT OUT!
  - D. DISCUSSION : IS IT BETTER TO WRITE A FRAMEWORK FROM SCRATCH
- CONCLUSION

First thing first: Xampp

- Couple of things you need to have:
  1. Php : As a server side scripting language
  2. Mysql (or any other preferred DB) : As a database
- Can find both of them wrapped up in a nice package with other in Xampp

First thing first: Xampp

- Proper download link can be found:
  - https://www.apachefriends.org/download.html
- Hit "Next" several times, till it says finished◄◄ you are done installing
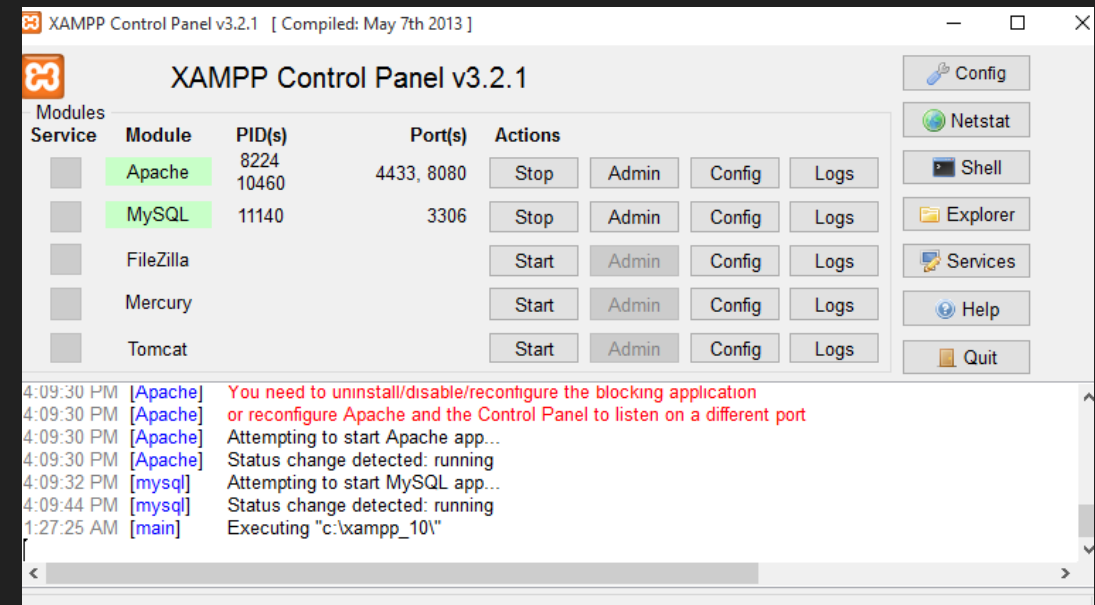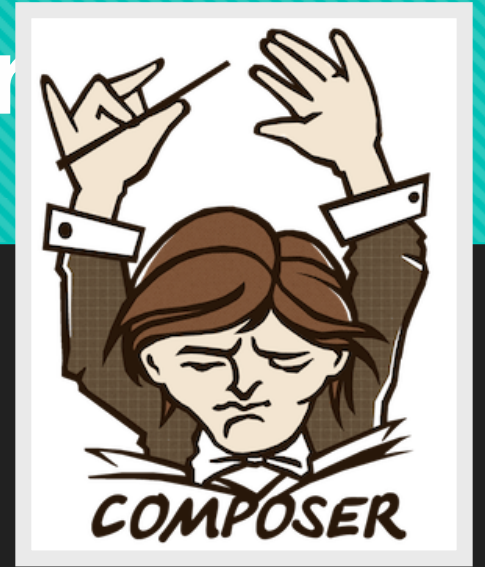- Open control panel, will end up seeing something like this:

Figure 1 – xampp, up and running

What is dependency manager

- As a easy-to-understand explanation is that it refers to a tool gathers proper packages per project.
- In PHP world, it is composer who does that!
- How to use:
  - Initial composer.json
  - Insert dependencies (namespace and version)
  - Run composer update/install
  - Have a coffee till it gets ready (personally prefer chocolate milk)
  - Include vendor/autoload.php and that it!

Install Composer

○ To install this magical tools we have couple of ways, you can use binary version for windows or use command in figure 2 in PHP folder and install it using command line:

Figure 2 – install composer using command line

Install Laravel

- Laravel is easy to install and it can be installed using several methods
- Preferred method is using composer dependency manager
- Installation may take 1 up to 20 minutes depending on your internet speed!
- Install Laravel with composer :
  - `composer create-project --prefer-dist laravel/laravel folderName`
- Notes:
  - If you get any error regarding turning on SSL in any of these step, go to php.ini file where you installed xampp and omit the ";" at the beginning of "extension=php_openssl.dll"
  - Creating a Laravel project must be accomplished in htdocs folder, if you want to see the result in you localhost

12

Headlines

- Why should we use frameworks?
- Why Laravel?
- HOW TO INSTALL
  - A. FIRST THING FIRST: XAMPP
  - B. WHAT IS DEPENDENCY MANAGER?
  - C. INSTALL COMPOSER
  - D. INSTALL LARAVEL
- **HOW TO KEEP WHAT WE NEED?**
  - **A. HOW TO USE WHAT WE NEED?**
  - **B. WHAT MAY WE REALLY NEED?**
  - **C. LET'S TRY IT OUT!**
  - **D. DISCUSSION : IS IT BETTER TO WRITE A FRAMEWORK FROM SCRATCH**
- CONCLUSION

13

HOW TO KEEP WHAT WE NEED?

- Remember when we said, Laravel is built using ready? It help us now!
- Fetch necessary packages and put them together to create your own custom made framework
- All packages can be found in https://packagist.org/ under illuminate/*

14

WHAT MAY WE REALLY NEED?

○ Recommended:

1. **illuminate/support** :   This package offer us many great features such as helper functions, ClassLoader, Collections and mo

Figure 3 –reference for more information about support package

2. **illuminate/container**   :   Contain the application class, which is used to connect components

WHAT MAY WE REALLY NEED?

Figure 4 –reference for more information about database package

3. illuminate/database : The Illuminate Database component is a full database toolkit for PHP, providing an expressive query builder, ActiveRecord style ORM, and schema builder. It currently supports MySQL, Postgres, SQL Server, and SQLite.

4. illuminate/routing : This package helps us to have a REST routing system without having so much trouble. It needs a proper .htaccess file (copy it from Laravel).

WHAT MAY WE REALLY NEED?

Figure 5 – reference for more information about Laravel-blade package

5. **illuminate/events**   :   We add this package because Laravel route needs this package but does not state it directly as a dependency.

6. **philo/laravel-blade** :   Instead of view package I prefer this one because it is easier to deal with as a standalone package. However, if you do not like it, feel free to use any other package you want for template engine.

Let's try it out!

It is time to get our feet wet! Let check what we talked about ◀◀

Discussion : is it better to write a framework from scratch

o Some may say create your own framework for your projects. They claim:

1. You do not have to worry about the updates and you and your team members update it every once in a while when you need

2. They say you know what's going on inside your framework so you have absolute control over your application

Discussion : is it better to write a framework from scratch

o Some may say use existing frameworks such as Laravel or symphony for big projects and Lumen and codeIgniter for smaller projects. They claim:

  o Creating a framework is like reinventing the wheel, it is so probable that you do not end up with a good wheel!

  o Popular frameworks has showed their reliability in action. They are more secure and so can be more trusted with more sensitive works

Headlines

- Why should we use frameworks?
- Why Laravel?
- HOW TO INSTALL
  - A. FIRST THING FIRST: XAMPP
  - B. WHAT IS DEPENDENCY MANAGER?
  - C. INSTALL COMPOSER
  - D. INSTALL LARAVEL
- HOW TO KEEP WHAT WE NEED?
  - A. HOW TO USE WHAT WE NEED?
  - B. WHAT MAY WE REALLY NEED?
  - C. LET'S TRY IT OUT!
  - D. DISCUSSION : IS IT BETTER TO WRITE A FRAMEWORK FROM SCRATCH
- **CONCLUSION**

CONCLUSION

○ Laravel shows a great flexibility because of packages it uses. You can use them with the default binding of Laravel or you can choose you favorable packages and create your own personal framework in less than an hour!

TO READ MORE:

1. http://www.richardbagshaw.co.uk/using-illuminate-components/
2. http://www.gufran.me/tag/Laravel/

QUESTIONS?

Thanks for being patience :)