

# Getting Started with Laravel (PHP Framework)

---

## GETTING STARTED



## Prerequisites

PHP Fundamentals  
including OOP PHP

HTML/ CSS  
Fundamentals

JavaScript  
Fundamentals

MySQL  
Database

# Laravel ?

PHP Framework

Utilities & Helpers

Great Documentation



Beautiful Code

Awesome Community

Rich Ecosystem

Secure, Lean and Fast

Model-View-Controller (MVC)

## Non-MVC vs MVC

Non-MVC



MVC



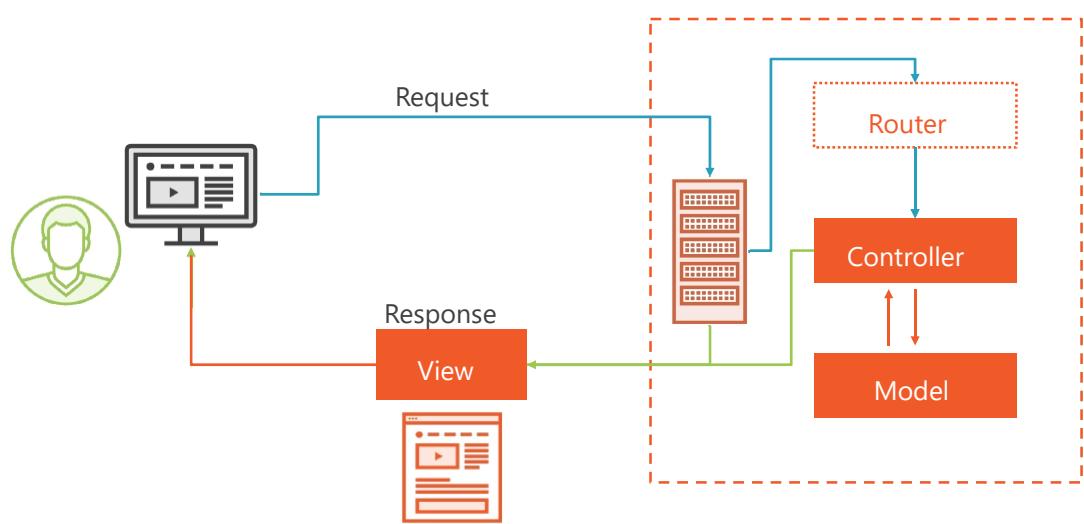
Business Logic

Presentation

# MVC

- *Model*: the Model represents your data structures. Typically your Model classes will contain functions that help you retrieve, insert, and update information in your database.
- *View*: the View is the information that is being presented to a user. A View will normally be a web page, but in CodeIgniter, a view can also be a page fragment like a header or footer.
- *Controller*: the Controller serves as a connector between the Model, the View, and any other resources needed to process the HTTP request and generate a web page.

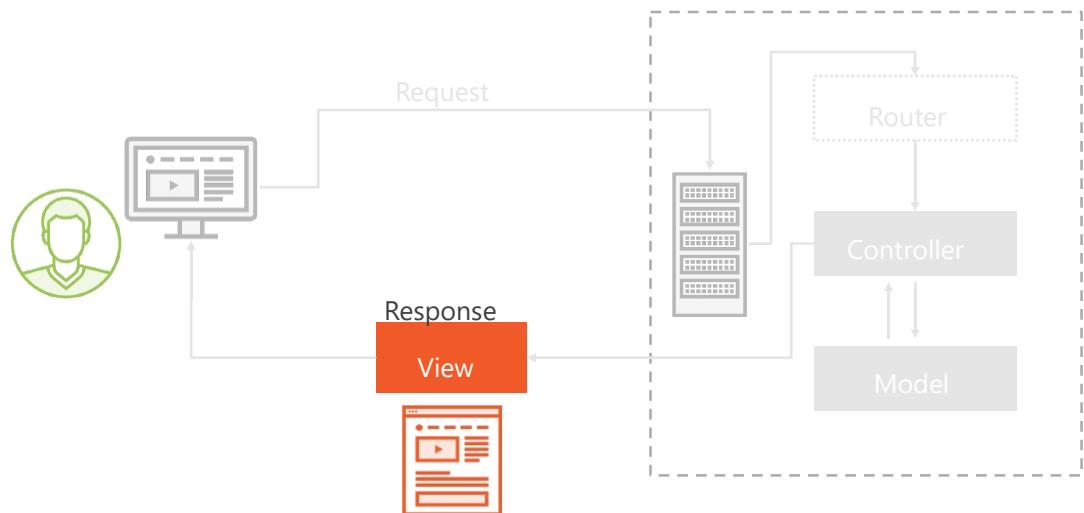
## Model-View-Controller (MVC)



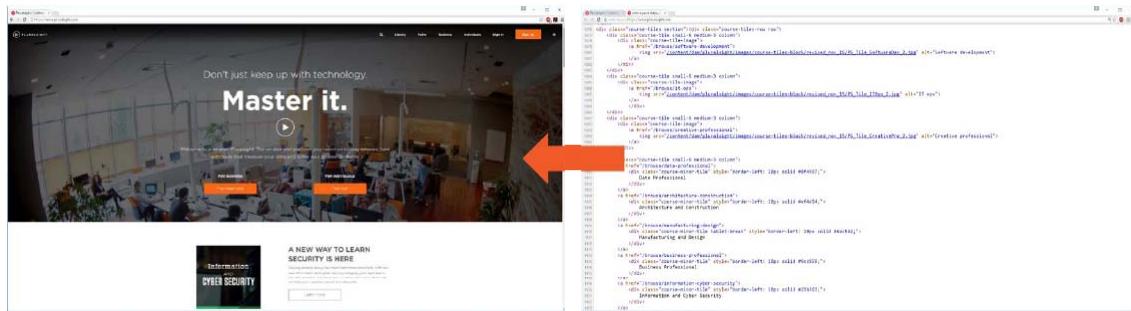
## Model



## View



## What Is a View?

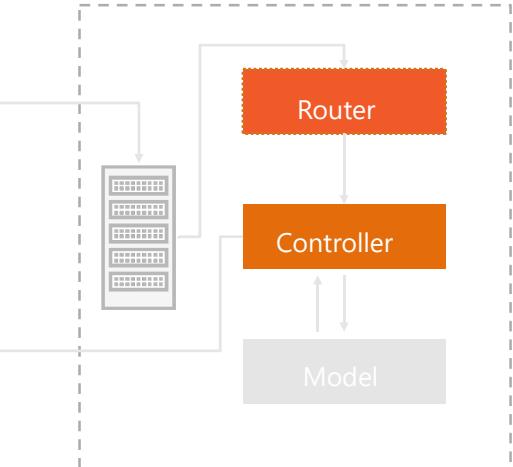


A View is what the User Sees

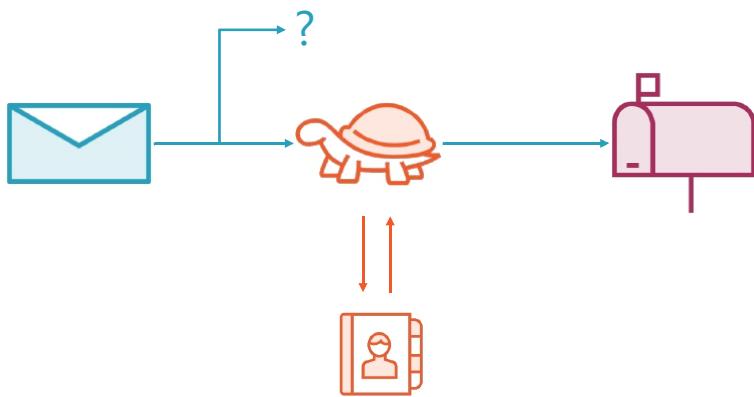
## Controller



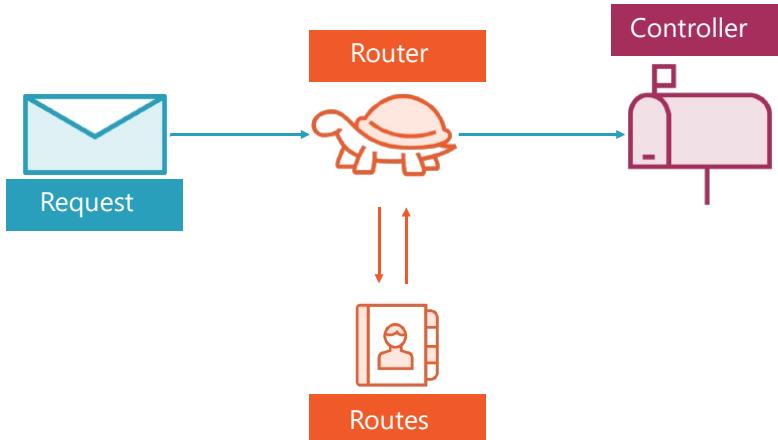
Request  
Response



## Delivering Mail



## The Laravel Router



## Laravel Setup (Using Composer)

PHP Environment Setup

Apache, PHP , MySQL

Download Composer

PHP Package Manager

Install Laravel

```
composer create-project laravel/laravel appname
```

## The Artisan CLI

`php artisan COMMAND`

Quick Creation of Application Components

Alternative to Manual File Creation

Utility Commands

## Artisan CLI Command Examples

```
$ php artisan list
```

```
$ php artisan help migrate
```

```
$ php artisan make:controller PostsController
```

```
$ php artisan make:model Post -m
```

```
$ php artisan make:migration add_posts_to_db --table=posts
```

```
$ php artisan migrate
```

```
$ php artisan tinker
```

# Eloquent ORM

Object Relational Mapper / ActiveRecord Implementation

```
<?php

use App\Flight;

$flights = App\Flight::all();

foreach ($flights as $flight) {
    echo $flight->name;
}

$flights = App\Flight::where('active', 1)
    ->orderBy('name', 'desc')
    ->take(10)
    ->get();
```

# Blade Templating

Simple but powerful template engine

Does not restrict php in views

Compiled into plain php and cached / No overhead

Views can extend layouts