# Laravel definition

Laravel is a free, open-source PHP web framework, created by Taylor Otwell and intended for the development of web applications following the model–view–controller (MVC) architectural pattern. Some of the features of Laravel are a modular packaging system with a dedicated dependency manager, different ways for accessing relational databases, utilities that aid in application deployment and maintenance

# Proper way of starting a project by Jeffrey:

Decide the first feature/page I want to write.  
2. Create a route, and direct the endpoint to a controller.  
3. Create the controller, add the method, load an appropriate view.  
4. Create the view, add some dummy text.  
5. Load page in the browser to ensure it's all working.  
6. Flesh out HTML.  
7. Write CSS in app.scss file, while running gulp watch  
8. Import "public/css/app.css" in the head tag of my view master page.  
9. Rinse and repeat.

# MVC

* Model – View – Controller
* Model – contains data and obtain it from DB, files, etc.
* View – displays the information
* Controller – access data from the model and display it in the view

# Composer:

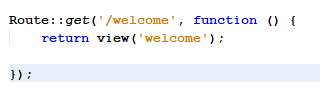
* Dependency manager

**Naming –** as to clearly define what we are doing otherwise we might have problems trying to find things

**Compact –** is an array of variables

# Laravel structure:

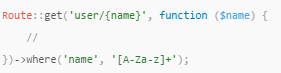
* **Routes – web.php**
  + where you define the path (url) to load the view
  + / -> home page
  + /about -> about page
  + Defining a route /welcome



* Defining route with required parameters



* Defining route with regex



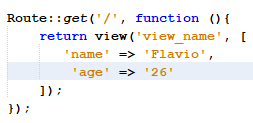
* Defining route referencing a controller (TimeController) with a method (Time)



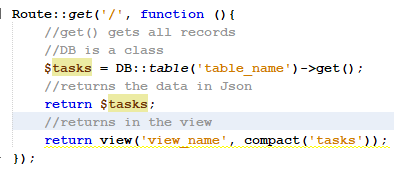
* If you would like a route parameter to always be constrained by a given regular expression, you may use the pattern method**.**



* Another example with parameters

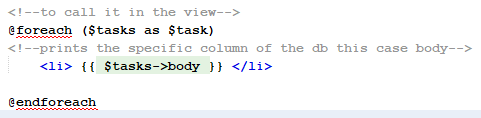


* Creating route taking data out of the db

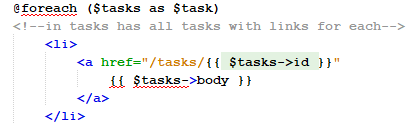


Also possible to run some SQL commands this case we specify the *where* could be possible to check things like the latest or newest





* Creating route to look for a specific task(previous example) instead of printing a list of tasks with links



* **Blade**
  + Laravel templates engine to compile in to vanilla php
  + Cleaner php code
  + With vanilla php instead of using <?php ?> we can simply use @ per line also to use variables {{ $myVariable }}
* **Vendor - .env file**
  + Single secure place to store everything from db connection, passwords, keys, api keys, etc.
  + Also can be used to change for production server with the production keys
* **Resources – Views**
  + Where all the HTML and CSS should be or where all the displayed things are
  + Create view as a layout file to have parts of the webpage that will appear in every page for instance footer and header, that way our html will be also cleaner
* **Database – Migrations**
  + To find the db table
* **Controllers –** makes the connection between the models and views
  + Path -> App/http/controllers

# Laravel most used commands:

* composer create-project laravel/laravel your-project-name – create new project
* php artisan list – List of useful commands, Laravel command line utility
* php artisan serve – Start server
* php artisan make:\_\_\_\_ - generates files
* php artisan help nameOfCommand – to see all the arguments and options of each commans
* in tinker DB::listen(function($query) {var\_dump($query->sql); }); returns the sql queries

# Databases in Laravel:

* starting SQL session with root by typing mysql -uroot -p where -uroot is the user root and -p the password
* to create a db – create database nameofdb;
* to select it – use nameofdb;
* to see all tables – show tables;
* the db connection file is located in **vendor - .env**
* **Migrations** are like version control for your db, after applying changes is necessary to migrate(php artisan migrate) or ‘commit’
* Create migration
* 
* **Schema facade** provides database agnostic support for creating and manipulating tables across all supported db systems
* When we call migrate that’s our way of saying turn the php definition in to how the tables should look like, and create them
* Laravel already includes a users table and passwords
* $card->fresh()->**notes**->first() – selects all the records, call a relationship as a property
* $card->fresh()->**notes**()->first() –only selects 1 record, call a relationship as a method

# Vagrant and Homestead

[Vagrant](https://www.vagrantup.com/) provides a simple, elegant way to manage and provision Virtual Machines.

Homestead is an official, pre-packaged Vagrant box that provides a development environment without requiring you to install PHP, a web server, and any other server software on your local machine.

# Eloquent

Easiest way of finding db records with tinker

The Eloquent ORM included with Laravel provides a beautiful, simple ActiveRecord implementation for working with your database. Each database table has a corresponding "Model" which is used to interact with that table.

Models allow you to query for data in your tables, as well as insert new records into the table.

**Create model**

Model could be a representation of something in your system



Or creates model and generates migration at the same time, this case automatically will create a model user and the specific migration for it, this case users table, because eloquent knows the model user will need to store multiple users in his migration

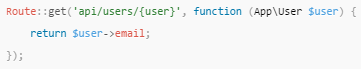


Eloquent assumes that each table as a field called id which correspond to the **primary key**

To find **timestamps** we can use the command *created\_at* or *updated\_at*

# Route model binding

Instead of injecting a user's ID, you can inject the entire User model instance that matches the given ID. This case user and $user variable



# Tinker

Laravel´s shell, nice way of interacting with the application

Run it – *php artisan tinker*

# Elixir

Can be used to make the files run smoother by shrinking them (delete spaces)

https://laravel.com/docs/5.0/elixir

Laravel Elixir provides a clean, fluent API for defining basic [Gulp](http://gulpjs.com/) tasks for your Laravel application. Elixir supports several common CSS and JavaScript pre-processors, and even testing tools.

MassAssignmentException -> is laravel trying to protect the code to be changed, otherwise everybody could do it