# Rappel sur l’architecture MVC

## Introduction

Le modèle MVC, Model View Controller est la référence de toutes les applications Web. Il connaît énormément de variantes de et configurations différentes mais tous les Frameworks et librairies applicatives se définissent par rapport à lui.

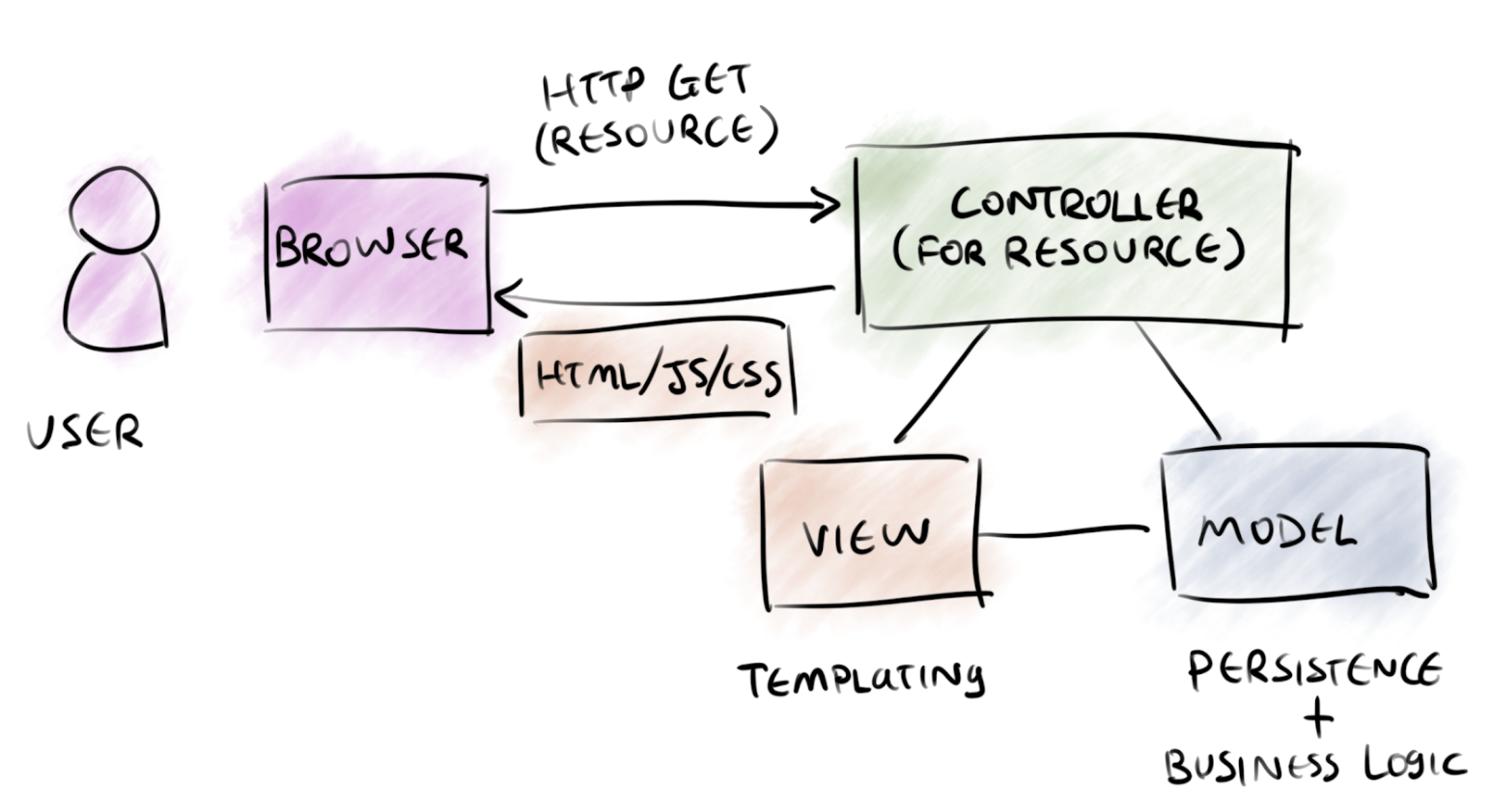
**Source** : <https://hackernoon.com/from-mvc-to-modern-web-frameworks-8067ec9dee65>

## Schéma traditionnel de l’architecture MVC

# 

## Schéma Web MVC :

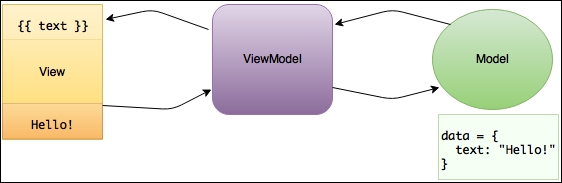
Le contrôleur a un rôle supplémentaire : prendre en charge la requête http initiale émise par le navigateur.



## Schéma MVVM correspondant à Vue.js

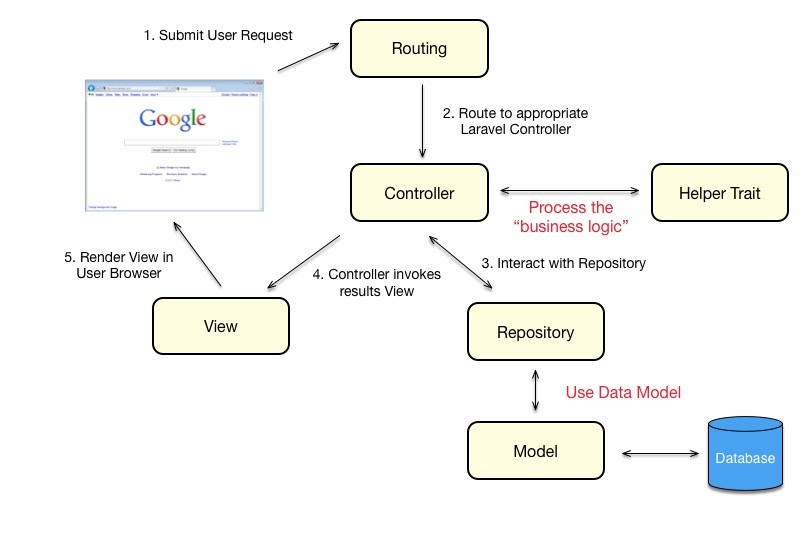
MVVM : Modèle – Vue - VueModele





## LARAVEL MVC Architecture

Source : <http://grigorov.website/blog/how-to-organize-your-project-with-php-and-laravel-to-get-the-best-structure-in-mvc-pattern>



# Composition du projet Laravel

# Introduction

Lorsqu’ on construit un projet Laravel, plusieurs questions se posent, permettant d’orienter le développeur sur les technologies et outils à configurer et utiliser.

## La sécurité

## La partie Viewer de l’application

Remarques : d’autres sujets importants lors du développement applicatif sont « transparent », à savoir n’ont pas vraiment d’impact lourd sur les développements :

* Le choix de la base de données : le paramétrage de la BDD est simple dans Laravel et une fois celui-ci fait, il n’y a pas d’impact sur le développement
* Le choix de l’ORM : l’ORM par défaut pour Laravel est Eloquent

# Annexes

## Difference d’approches entre ORM – Eloquent versus Doctrine

Laravel utilise principalement Eloquent tandis que Symfony fonctionne avec Doctrine. Cependant Laravel est construit à partir de Symfony et peut également fonctionner avec Doctrine.

Quelles sont les différences entre ces 2 ORM ?

La principale différence est que **Eloquent** utilise le pattern ***Active Record*** tandis que **Doctrine** utilise le pattern ***Data Mapper***.

En une ligne, Data Mapper est plus adapté aux projets d’une certaine taille , il faut limiter l’utilisation d’Active Record, donc d’Eloquent, aux petits projets avec de simples opérations CRUD.

Les discussions sur le sujet sont légion, par exemple :

<https://stackoverflow.com/questions/3828265/is-data-mapper-a-more-modern-trend-than-active-record>

<https://www.infinitypp.com/software-patterns/activerecord-vs-datamapper-pattern-php-laravel/>