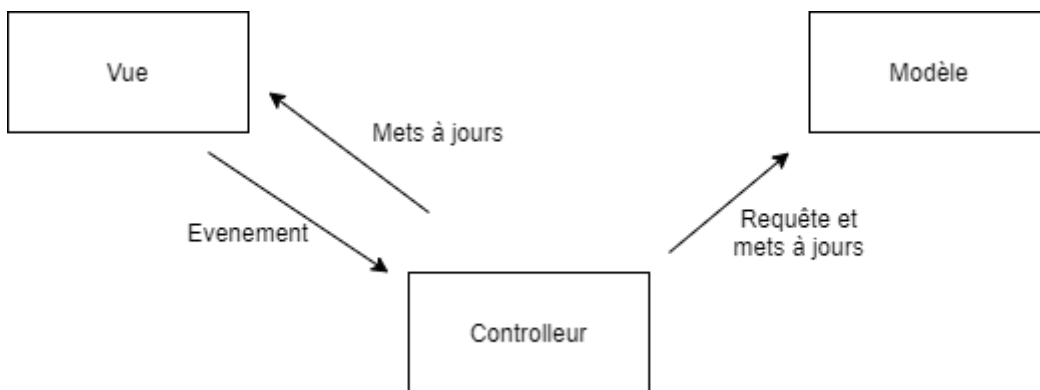


# Mettre en place un architecture MVC avec PHP

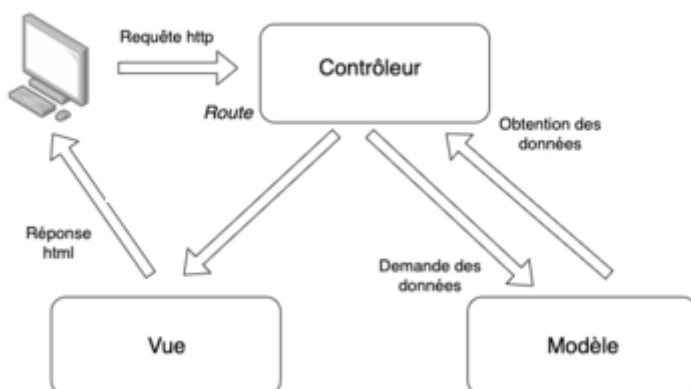
## Le Modèle MVC

Le modèle MVC pour : Model - View - Controller est un Design Pattern visant à séparer une application en trois modules :

- Modèle : les données
- Controlleur : traite les données et mets à jours la vue
- Vue : Définit l'interface utilisateur



Si on applique au modèle du Web cela donne le schéma suivant :



## MCV en PHP Vanilla

Commeçons par créer une arborescence de dossiers pour ranger nos scripts en fonction des différents modules de MVC :

- Un dossier `model` : contient des scripts qui contiennent des fonctions de requêtes SQL ainsi qu'un script de connexion à la base de donnée.
- Un Dossier `view` : va contenir des fichier `.tpl` qui à partir desquels seront générées des pages HTML.
- Un dossier `controller` : contient des scripts qui vont faire appel aux fonctions du `model` afin de récupérer des données, les traiter, puis faire appel à la vue correspondante.

et un fichier `index.php` pour le routage.

## Le routage

Pour accéder aux service des controleurs on va passer dans chaque requêtes une variable `controller` et une variable `action` en paramètre d'URL. Exemple :

`http://localhost/index.html&controller=user&action=login`. Si bien qu'on fera toujours appel à `index.html`, qui lui s'occupera d'appeler la bonne action du bon controleur. Le code qui s'en occupe est le suivant :

```
if (isset ( $_GET ['controller'] ) & isset ( $_GET ['action'] )) {  
    [$controle = $_GET ['controller'];  
    [$action = $_GET ['action'];  
} else { // absence de paramètres : prévoir des valeurs par défaut  
    [$controle = "user";  
    [$action = "login";  
}  
  
// Inclure le fichier php de contrôle  
// Et lancer la fonction-action issue de ce fichier.  
  
require ('./controller/' . $controle . '.php');  
$action ();
```

## Connexion à la base de donnée

On va créer un script de connexion à la base de donnée `connectDb` dans le dossier `model` et on y écrit la fonction suivante :

```
function connect(){  
    $hostname = "localhost";  
    $base = "nom de la base de donnée";  
    $loginBD = "root";  
    $passBD = "mot de passe de la base de donnée";
```

```
try {  
    $bdd = new PDO ( "mysql:server=$hostname; dbname=$base", "$loginBD", "$passBD", array  
(PDO::MYSQL_ATTR_INIT_COMMAND => 'SET NAMES utf8'));  
    return $bdd;  
}  
catch ( PDOException $e ) {  
    echo "Echec de connexion à la base de donnée";  
    die ( "Echec de connexion : " . $e->getMessage () . "\n" );  
}  
}  
``
```

---

Revision #5

Created 27 September 2020 14:02:08 by Arsène Lapostolet

Updated 12 September 2021 13:45:53 by Arsène Lapostolet