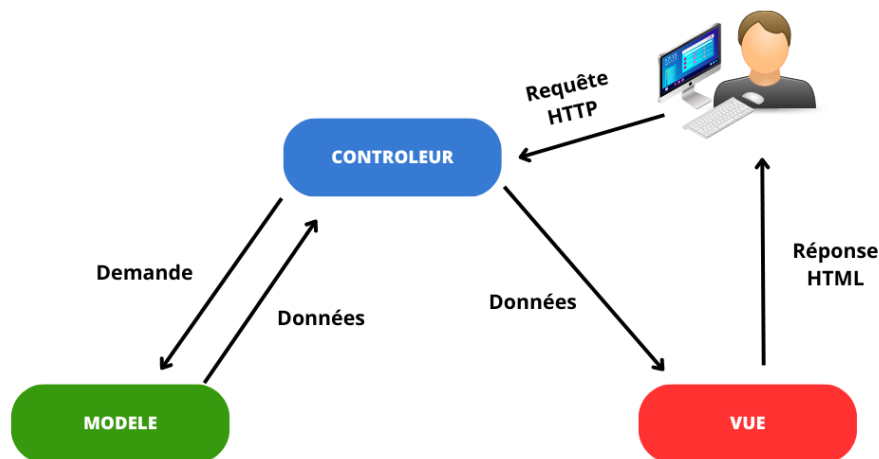


# EVALUATION MODULE MVC - SYMFONY

1. Réaliser un schéma du modèle MVC. Indiquer les dépendances entre composants. Identifier au moins un composant Contrôleur, Modèle et Vue dans l'application bibliothèque.



Pour le contrôleur il y a le fichier `BookController.php` qui gère les actions liées aux livres dans l'application bibliothèque.

Pour le modèle il y a le fichier `Book.php` qui décrit l'entité pour les livres de l'application.

Pour le modèle il y a le fichier `add_book.html.twig` qui permet d'ajouter de nouveaux livres à la base de données.

2. Est-ce que le framework Symfony oblige l'utilisateur à utiliser le pattern MVC ?

Oui car Symfony est un framework PHP qui repose sur le pattern MVC pour structurer les applications web qu'il permet de développer. Ce pattern permet de séparer les différentes tâches liées à l'affichage, à la gestion des données et à la logique de l'application, ce qui facilite la maintenabilité et la réutilisabilité du code.

De plus, Symfony conseil l'utilisation du pattern MVC mais il n'est pas obligatoire de l'utiliser car on peut également utiliser les patterns MVVM (Model-View-ViewModel) ou MVP (Model-View-Presentation). En soit le pattern MVC est recommandé pour bénéficier pleinement des avantages qu'offre Symfony.

### 3. Expliquer en quelques lignes à quoi sert un Objet Relationnal Mapper (ORM) ?

Un ORM est un outil logiciel qui permet de faire le lien entre les données stockées dans une base de données relationnelle et les objets utilisés dans une application logicielle. Il permet de manipuler des données stockées dans une base de données comme s'il s'agissait d'objets en mémoire, sans avoir à écrire des requêtes SQL complexes. De plus, ses avantages sont nombreux (exemple: il peut permettre une meilleure maintenabilité, sécurité, une plus grande flexibilité..). Il facilite donc la manipulation des données stockées dans une base de données relationnelle en les transformant en objets en mémoire. Cela permet de simplifier le développement et la maintenance de l'application, tout en améliorant sa sécurité et sa flexibilité.