## 5) Architecture de l’application et technologies utilisées

### Contraintes techniques et solutions

La première question qui s'est posée pour la réalisation de la partie technique du projet a été de choisir une architecture pour l'application.

En effet, le site requiert une structure qui puisse stocker des informations, les utiliser et les traiter. L’application doit pouvoir gérer une base de données de façon optimale et nécessitera d’avoir des pages fluides et réactives.

Les types de périphériques qui accèderont à l’application seront variés, ce qui imposera un développement de l’application de manière « **responsive** » pour s’adapter à tous ces médias.



**La Méthode Modèle-Vue-Contrôleur (MVC)**

Le MVC est une architecture et une méthode de conception qui organise l'Interface Homme-Machine d'une application logicielle (dans notre cas un site Web) en trois parties :

* Les données (Modèle).
* L’interface homme-machine (Vue).
* La logique de contrôle (Contrôleur).

Grossièrement, cela permet une séparation entre les traitements de données et la présentation.

**Le modèle**

Le modèle représente les structures de données. Typiquement, les classes modèles contiennent des fonctions qui aident à récupérer, insérer et mettre à jour des informations de la base de données.

**La vue**

La vue correspond à l'interface avec laquelle l'utilisateur interagit. Elle se présente sous la forme d'un Template représentant l'interface, mais sans les données.

**Le contrôleur**

Il gère l'interface entre le modèle et le client. Il va interpréter la requête de ce dernier pour lui envoyer la vue correspondante. Il effectue la synchronisation entre le modèle et les vues.



### Justification des choix technologiques

Le développement de l'application nécessite une phase d'étude et de conception préalable à la mise en œuvre technique proprement dite. Cette phase requiert des compétences pointues et une connaissance approfondie des technologies web afin d'être en mesure de faire les choix pertinents desquels dépendront la pérennité et les performances de l'application.

Dans ce cadre, nous avons recensé les spécifications fonctionnelles de notre projet, ce qui nous amène à la rédaction du dossier d’analyse et des spécifications détaillées, pour entamer par la suite la conception d’une architecture fiable, couvrant les exigences fonctionnelles et techniques du site.

Afin d’ en assurer la mise en œuvre, ce projet sera réalisé à travers plusieurs technologies récentes, telles que le langage de programmation JavaScript (JQuery, Ajax), le Framework CSS Bootstrap



**Pourquoi Laravel ?**

On rencontre souvent dans les forums la question de savoir quel est le meilleur Framework. La réponse la plus intelligente à donner est que le meilleur est celui que l’on connaît bien et qui répond à la fois à nos besoins et notre compétence. Il y a toutefois des critères objectifs dans le choix d’un Framework :

**la documentation :** un Framework mal documenté fait perdre plus de temps qu’il n’en fait gagner, Laravel est très bien documenté

**la communauté :** lorsqu’on rencontre une difficulté il est bon de recevoir une aide rapide pour ne pas rester bloqué longtemps sur un problème de détail, la communauté de Laravel est réactive.

**la durée d’apprentissage :** un temps d’apprentissage trop long fait perdre l’intérêt d’un Framework même si on factorise notre connaissance en multipliant les cas d’application, Laravel est rapide à prendre en mains tout en ne faisant aucune concession sur les fonctionnalités,

**la pérennité :** apprendre un Framework pour apprendre quelques mois plus tard qu’il est abandonné est plutôt frustrant. Laravel est soutenu par une importante communauté de développeurs et d’utilisateurs, ce qui lui garantit une durée de vie raisonnable,

**la richesse des outils :** certains Framework sont intéressants mais nécessitent de nombreuses greffes pour devenir vraiment opérationnels. Laravel est à la base suffisamment complet pour ne nécessiter que des extensions pour des besoins particuliers,

**la simplicité :** utiliser un Framework et se retrouver à écrire un code plus complexe que celui qu’on aurait utilisé avec simplement du PHP n’est pas une opération judicieuse, le code qu’on écrit avec Laravel est toujours simple, lisible, concis,

**la version de PHP utilisée :** Laravel est fondé sur PHP 5.3, ce qui permet d’utiliser par exemple les fonctions anonymes (lambda), les fermetures, les espaces de noms…

**les mises à jour :** on est jamais à l’abri d’une faille de sécurité, le fait d’utiliser un Framework est un peu à double facettes : le code a été normalement bien écrit, validé, mais il est lisible par tout le monde et devient par le fait vulnérable, d’où la nécessité de procéder à des mises à jour en cas de découverte d’une faille, l’équipe de Laravel est assez réactive pour assumer cette tâche essentielle,

**les standards de programmation :** le code d’un « bon » Framework est écrit en suivant les « bonnes pratiques » et il nous invite à faire de même, Laravel a souvent été salué pour la pureté de son code.

**Les possibilités d’extension :** on doit pouvoir facilement étendre les fonctionnalités d’un Framework sans détruire son organisation, Laravel permet de le faire avec élégance en suivant toujours les mêmes principes avec ses bundles.



**Pourquoi Bootstrap ?**   
  
Tout d’abord, Bootstrap est un Framework HTML, CSS & JavaScript fonctionnant sur n'importe quelle technologie serveur ou environnement serveur avec une dizaine de composants, et plugin JavaScript. Son concept est la création et maintenance rapide d'un site internet avec rendu correct, et interface complète.

Les avantages de Bootstrap sont sa facilité, accessibilité, sa structure, sa méthodologie, sa vitesse de développement accrue, porté vers le futur, grille fixe ou fluide, personnalisable et modulable.

**Pourquoi JQuery** ?

Il existe beaucoup de bibliothèque JavaScript alors pourquoi utiliser jQuery ?

Car jQuery est le plus complet et pour être le plus complet jQuery se base sur certaines stratégies.

**Exploiter le CSS**

jQuery base son mécanisme de localisation des éléments de la page sur les sélecteurs CSS, ce qui permet au développeur de pouvoir sélectionner très facilement ce qu'il souhaite.

**Accepter les extensions**

jQuery accepte très bien et très facilement les extensions.

La création de plugin jQuery est simple et très bien documentée ce qui permet à jQuery de posséder une multitude d'extensions.

**Autoriser plusieurs actions sur une ligne**

Le résultat de la plupart des opérations sur un objet est l'objet lui-même, ce qui permet de ne pas dupliquer l'objet mais de travailler tout le temps sur le même.



**Pourquoi MySql ?**

MySQL est la base de données open source la plus répandue au monde, elle est facile d’utilisation, le coût d’exploitation est faible, elle offre de bonnes performances et possède une véritable communauté de développeurs pouvant répondre et aider en cas de soucis quelconque.