

**RAPPORT DE PROJET WEB**

GODARD Thibault IG3

**SOMMAIRE :**

**I] CADRE FONCTIONNEL :**

**1) Exigences fonctionelles :**

**-** Cahier des charges

- Présentation et choix des technologies utilisées.

**2) Exigences non-fonctionelles :**

- Idée personelle de projet.

**II] ARCHITECTURE LOGICIEL :**

**1) Spécification :**

- Diagramme de classes entre les différents acteurs.

- Justification des multiplicités.

**2) Spécification des cas d’utilisation :**

- Diagramme des cas d’utilisation.

**III] ARCHITECTURE DE DEPLOIEMENT :**

**IV] ANALYSE DES RESULTATS :**

**1) Bilan**

**2) Autocritique**

**I] CADRE FONCTIONNEL :**

**1) Exigences fonctionelles :**

- Cahier des charges :

Tout d’abord, il s’agit de réaliser une application Web dynamique utilisant une base de donnée relationnelle sur un sujet au choix.

Le cahier des charges comporte néanmoins certaines contraintes techniques :

- doit être compatible avec tout type de navigateur internet : IE11, Safari, Chrome, Firefox (aussi dans leurs dernières versions bien entendu)

- doit comporter une architecture MVC.

- le choix des langage est libre. (mais à justifier)

- le choix d’une base de donnée libre. (mais relationnel

- Contraintes : 5 tables, 2 triggers.

- Présentation et choix des technologies utilisées.

**-> Pourquoi un site dynamique :**

(INTRODUCTION A PHP)

PHP fait partie d’un des langages permettant de créer un site dynamique : personnalisé à l’utilisateur (afficher le nom de la personne, sa photo, messagerie privée … En gros, personnalise l’expérience du visiteur. C’est différent de statique !

On peut résumer son fonctionnement comme une relation :

Client / serveur entre notre machine / la machine possédant

(stockant)

le site web et l’envoyant à notre machine (le serveur)

**Statique**  -> Client demande au serveur la page -> le serveur affiche la page statique en envoyant le code HTML ET CSS

**Dynamique (cahier des charges)** -> le serveur génère d’abord la page (personnalisée) et l’envoie ensuite au client.

Etant donné qu’il faudra personnaliser chaque page web de notre site pour chacun des clients, cela justifie l’utilisation d’un langage permettant un site web dynamique.

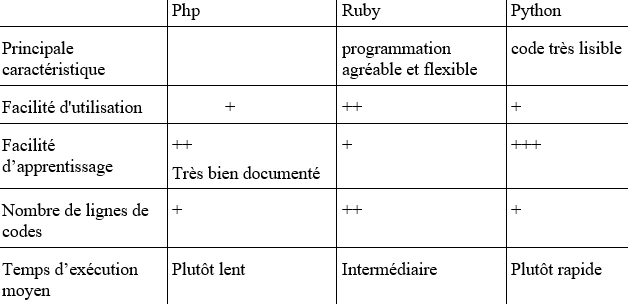
Il existe de nombreuses autres technologies (langages) permettant de créer un site dynamique, je vais donc tenter de justifier le choix des technologies que j’ai utilisées.

-> **HTML et CSS** (langages de description) sont des « standards » dont on ne peut se passer dans le cadre de la création d’un site web, qui sera d’abord statique avant d’être dynamique.

-> **PHP** (Rôle de serveur/langage de « programmation ») :

C'est un langage que seuls les serveurs comprennent et qui permet de rendre notre site dynamique (comme indiqué précédemment). C'est PHP qui va « générer » la page web.

On peut aussi utiliser Java, Python, Ruby et pleins d’autres (moins connus et moins performants.)



PHP ne semble pas être le choix le plus judicieux cependant il est très bien documenté (par une importante communauté) et c’est pour cela que j’ai préféré choisir ce dernier.

-> **MySQL** : Il existe un t’as d’autres outils :

* **Oracle** : c'est le SGBD le plus célèbre, le plus complet et le plus puissant. Il est malheureusement payant (et cher même s’il existe des versions gratuites limitées).
* **Microsoft SQL Server** : (édité par Microsoft), on l'utilise souvent en combinaison avec ASP .NET, bien qu'on puisse l'utiliser avec n'importe quel autre langage. Il est payant (même s’il existe des versions gratuites limitées).
* **PostgreSQL** : Ce SGBD est quant à lui libre et gratuit comme MySQL, qui propose des fonctionnalités plus avancées.

Il dispose d'une communauté un peu moins importante que MySQL

A l’image de mon choix pour PHP, j’ai préféré choisir MySQL car ils interagissent facilement ensemble, de par le package WAMP (pour ma part sous Windows) créé à cet effet.