



# Projet Evalive

*Projet informatique individuel*

*Livrable Final*

*Renault Loïc*

*27 Avril 2017*



## Table des matières

I.	Introduction.....	3
II.	Sujet du projet.....	3
III.	Présentation de l'application : Evalive .....	3
IV.	Exigences .....	7
1.	Exigences générales: .....	7
2.	Exigences spécifiques : .....	8
a.	Concernant les enseignants : .....	8
b.	Concernant les apprenants : .....	8
V.	Ressources.....	8
VI.	Choix technologiques .....	8
VII.	Choix d'architecture .....	9
VIII.	Base de données.....	9
IX.	Gestion de projet.....	10
1.	Communication avec le client .....	10
2.	Planning.....	10
X.	Mise en œuvre technique .....	10
1.	Règles de code.....	10
2.	Problèmes rencontrés .....	11
3.	Pistes d'amélioration.....	11
XI.	Conclusion .....	12

## I. Introduction

Ce document présente le travail réalisé dans le cadre du projet informatique individuel de 2<sup>ème</sup> année à l'ENSC, sur une durée de trois mois et demi. Il s'agissait dans mon cas de la réalisation d'une application web ayant pour objectif d'aider les enseignants à suivre l'avancement des progrès de leurs étudiants.

Vous trouverez dans ce document la présentation du travail réalisé jusqu'à aujourd'hui mais aussi un bilan fonctionnel qui prendra en compte les exigences fixées en début de projet et un bilan sur la gestion du projet.

## II. Sujet du projet

Le projet est né de la nécessité, pour les enseignants, de pouvoir suivre la progression de leurs étudiants sur une activité parfois longue et décomposée en plusieurs étapes. En effet, pouvoir identifier le plus tôt possible les étudiants en difficultés est d'un intérêt majeur pour adapter la vitesse d'enseignement et proposer une aide adaptée.

Ce projet avait donc pour but de proposer une interface simple d'utilisation et facile d'accès pour permettre un suivi de l'avancement des étudiants sur une activité spécifique. Des fonctionnalités de départ ont été définies en accord avec le client dès le début du projet.

## III. Présentation de l'application : Evalive

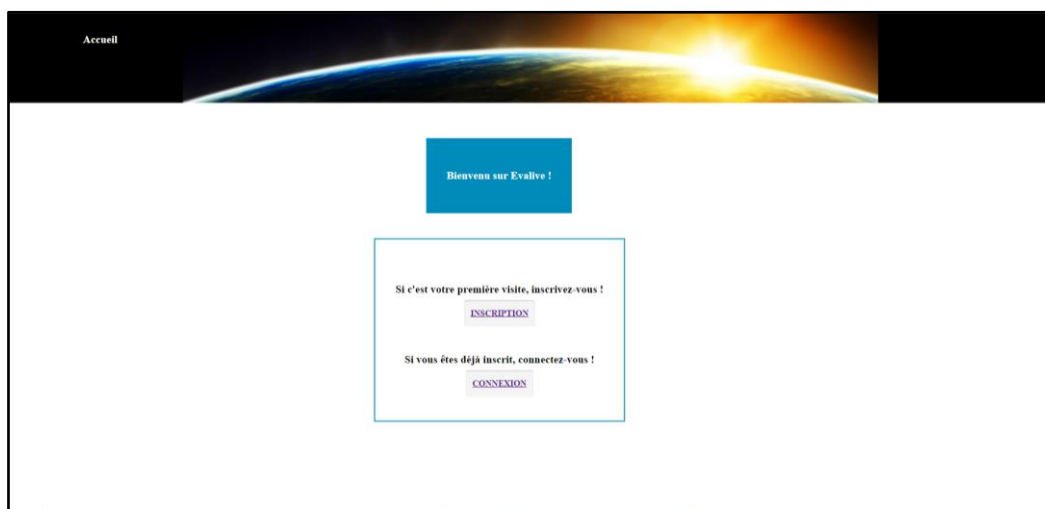


Figure 1: Page d'accueil

Lorsque l'on ouvre l'application web, on arrive sur la page d'accueil qui propose de s'inscrire ou de se connecter.

Accueil

Nom

Prénom

Identifiant

Mot de passe

Récopiez votre Mot de Passe

Email

Enseignant  
Etudiant  
Enseignant

Confirmer

Figure 2: Inscription

Si l'utilisateur choisi de s'inscrire, il est redirigé vers cette page qui lui demande plusieurs informations nécessaires à l'inscription à Evalive. L'utilisateur devra redonner son identifiant et son mot de passe lors de sa prochaine visite pour se connecter.

L'adresse permettra dans une version plus développer de mettre en place un système de communication entre les enseignants et les apprenants. Pour l'instant elle n'est pas utilisée.

Le statut (étudiant ou enseignant) permet de proposer un affichage différent en fonction du profil.

Accueil

Identifiant

Mot de Passe

Connexion

Figure 3: Connexion

Lorsque l'utilisateur n'en est pas à sa première visite, il peut directement se connecter. Il doit alors redonner son identifiant et son mot de passe renseigné lors de l'inscription.

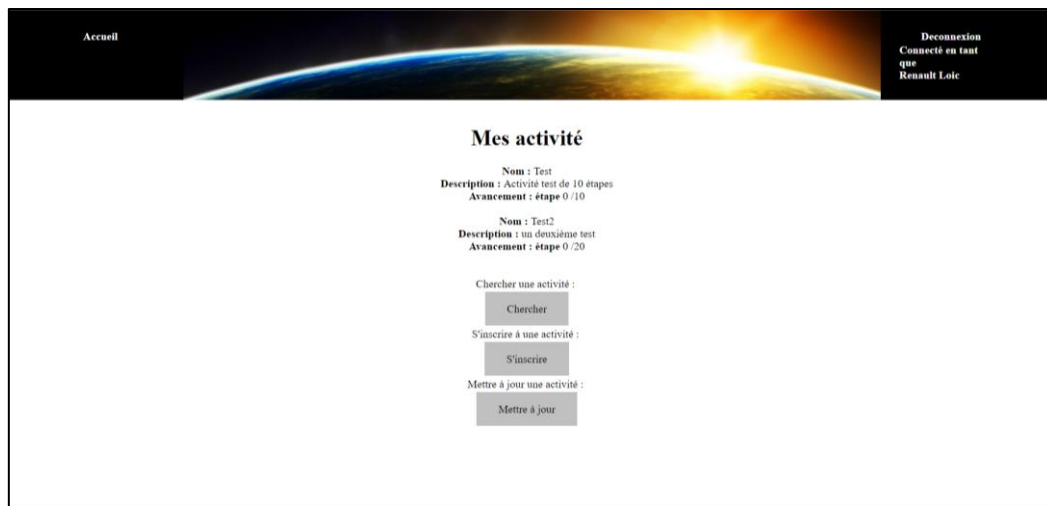


Figure 4: Tableau de bord d'un étudiant

Quand un étudiant se connecte, il voit apparaître son tableau de bord. Cet affichage lui permet de visualiser l'ensemble des activités auxquelles il est inscrit ainsi que les différentes fonctionnalités auxquelles il a accès. Pour chaque activité auxquelles il participe, le site affiche le nom de celle-ci ainsi que sa description et il affiche l'avancement de l'étudiant.

Ensuite l'étudiant a la possibilité de rechercher une activité, de s'inscrire à une activité ou bien de mettre à jour l'une de ses activités. Cette option n'est pas encore fonctionnelle.

Dans le bandeau supérieur, l'affichage est adapté et permet de confirmer que l'utilisateur est connecté sur le bon compte. En plus de cela, il y a un bouton « Déconnexion » pour quitter la session en cours.

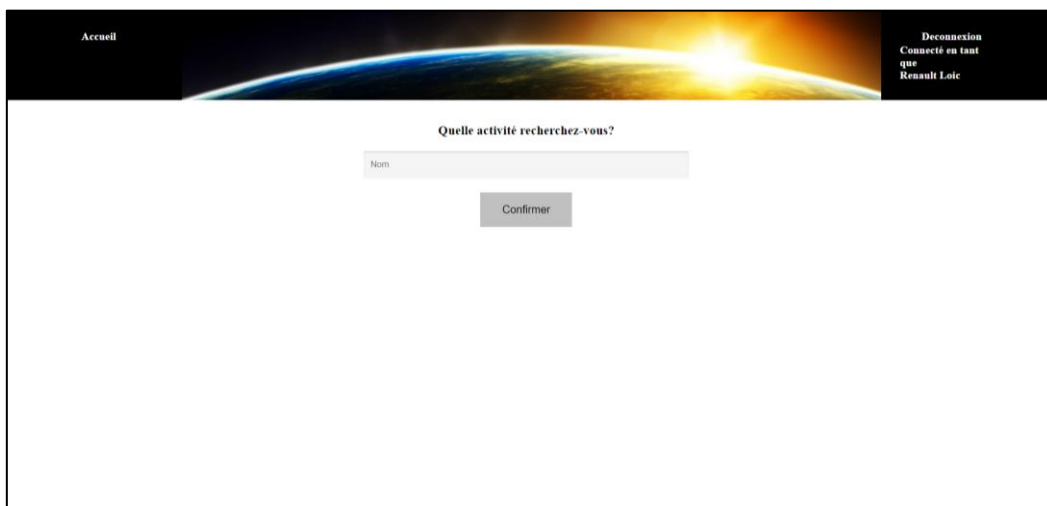


Figure 5: Rechercher une activité

Si l'étudiant souhaite rechercher une activité dont il connaît déjà le nom, il peut cliquer sur le bouton rechercher et arriver sur cette page (Figure 5). Il ne lui reste plus qu'à taper le nom et lancer la recherche. Si l'étudiant est déjà inscrit à cette activité alors l'écran va afficher le résumé le concernant (qui identique à celui du tableau de bord) sinon il a la possibilité de s'y inscrire.

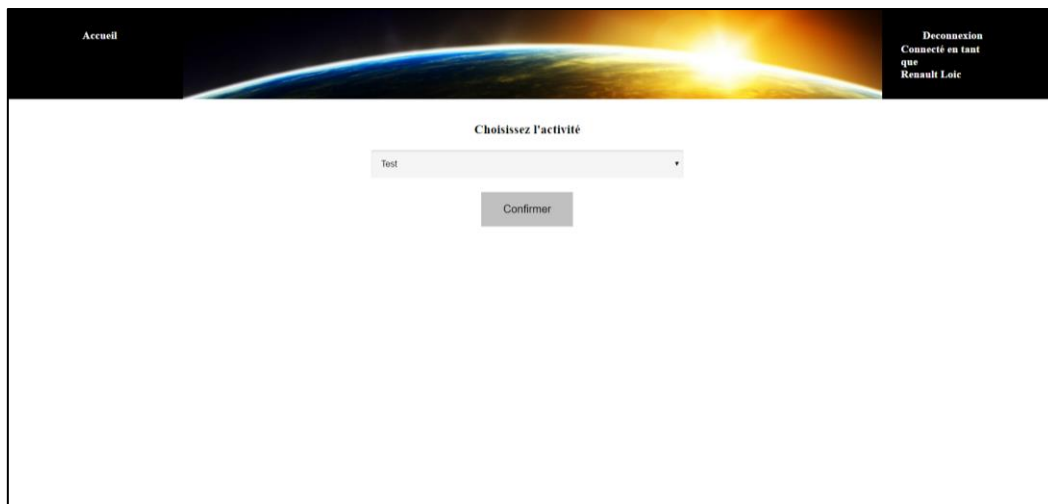


Figure 6: Inscription

Si l'étudiant choisi de cliquer sur le bouton « S'inscrire », il arrive sur un page où il a la possibilité de choisir dans un menu déroulant l'activité à laquelle il souhaite s'inscrire. L'inscription se fait automatiquement dès que l'étudiant clique sur le bouton de validation.

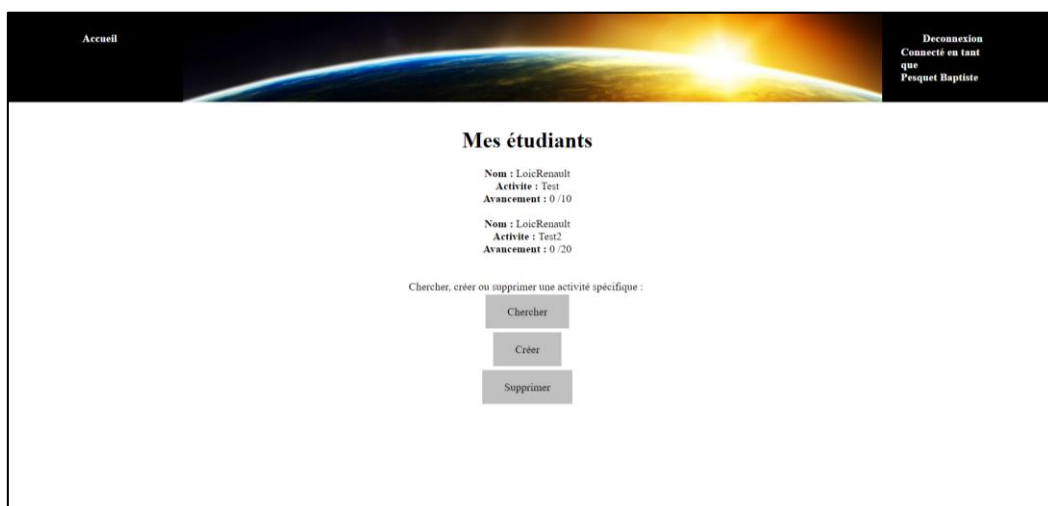


Figure 7: Tableau de bord d'un enseignant

Voici le tableau de bord d'un enseignant (Figure 7). Il est très similaire à celui d'un étudiant à ceci près que l'enseignant a accès à la liste des étudiants qui sont inscrits dans les activités dont il est le responsable. Ainsi pour chaque étudiant, il visualise son nom, l'activité en question ainsi que son avancement.

Ensuite, il a la possibilité de faire une recherche d'activité ou d'étudiant de la même façon que les étudiants font une recherche d'activité. Mais il peut aussi créer une nouvelle activité.

Figure 8: Création d'une nouvelle activité

Lorsqu'un enseignant souhaite créer une nouvelle activité, il doit remplir un certain nombre de champs tout indispensable. Tout d'abord le nom de l'activité, le module auquel elle est associée, le nombre d'étapes ainsi que sa description.

## IV. Exigences

Au début du projet, des exigences/fonctionnalités ont été définies comme étant indispensable dans la solution finale. Il s'agissait des actions minimales que devait pouvoir réaliser les futurs utilisateurs, qu'ils soient enseignants ou apprenants. Elles ont toutes été consignées dans le cahier des charges établi à cette période.

Par conséquent, ces fonctionnalités ont été les premières à avoir été codées. Voici un bilan qui récapitule la présence ou non des fonctionnalités exigées, dans l'application finale.

EF = Exigence fonctionnelle

ENF = Exigence non fonctionnelle

### 1. Exigences générales:

Référence	Description	Etat
EF_1	Création d'un profil utilisateur	Fait
EF_2	Connexion différentielle en fonction du profil de l'utilisateur (enseignant ou apprenant)	Fait
EF_3	Rechercher une activité	Fait
ENF_2	Avoir un code couleur en fonction de l'avancement dans l'activité	Non fait
ENF_3	Hébergement en ligne de l'application	Non fait
ENF_4	Fonctionnement de l'application sur tout type de support (responsive design)	Fait

L'hébergement en ligne du site a été abandonné en cours de projet car il s'agissait d'une manipulation trop compliquée à mettre en place par le développeur. A l'origine, il était prévu que l'application soit déployée sur un hébergement en ligne de type SaaS.

## 2. Exigences spécifiques :

### a. Concernant les enseignants :

Référence	Description	Etat
ENF_1	Interface affichant l'avancement des apprenants	Fait
EF_4	Gestion des activités (ajout, modification, suppression)	Fait
EF_5	Gestion des étapes d'une activité (ajout, modification, suppression)	Fait
EF_5	Créer un module	Fait
EF_6	Supprimer une activité	Fait
EF_8	Valider ou non l'inscription d'un étudiant à une activité	Non fait
EF_9	Afficher les détails d'avancement d'un seul étudiant	Non fait

### b. Concernant les apprenants :

Référence	Description	Etat
ENF_3	Interface affichant l'avancement de chaque activité en cours	Fait
EF_10	S'inscrire à une activité	Fait
EF_11	Mettre à jour son avancement dans une activité en cours	Non fait

## V. Ressources

Au début du projet, il existait déjà plusieurs solutions sur le marché proposant ce que nous souhaitions apporter. Le seul problème est qu'elles sont souvent très complexe à utiliser et qu'il y a une certaine maîtrise pour pouvoir naviguer dedans. De plus, il était impossible de mettre en place une telle solution du fait de leur complexité, tant au niveau de l'architecture que du code utilisé derrière. Nous avons donc opté pour une solution plus simple dans laquelle un utilisateur lambda n'aura aucun mal à trouver son chemin entre les différentes pages.

En ce qui concerne les ressources personnelles du développeur nous sommes partis sur les acquis en programmation Web de l'année passée. Nous verrons un peu plus loin pourquoi ce point précis a causé beaucoup de retard sur la phase de développement du projet.

## VI. Choix technologiques

Dès le début du projet, en accord avec le client, il a été décidé de développer l'application en PHP et JavaScript, le tout étant relié à une base de données via un protocole SQL. Il s'agissait là d'une bonne occasion de découvrir un nouveau langage de programmation : le JavaScript.



## VII. Choix d'architecture

D'un point de vue architectural, l'application est organisée de la sorte : l'interaction avec l'utilisateur se fait via les pages HTML qui sont projetées sur l'écran. Ces mêmes pages font appel à deux éléments distincts : d'une part des scripts CSS qui permettent de mettre en place une connexion avec la base de données et de communiquer avec celle-ci via des requêtes SQL, et d'autre part un script CSS qui, lui, est chargé de l'aspect esthétique de l'application. C'est notamment lui qui gère la position des différents éléments, leur couleurs, leur taille, etc.

Le diagramme suivant permet de mieux apprécier l'architecture de l'application ainsi que les différences entre les types d'utilisateurs, surtout d'un point de vue fonctionnalités accessibles.

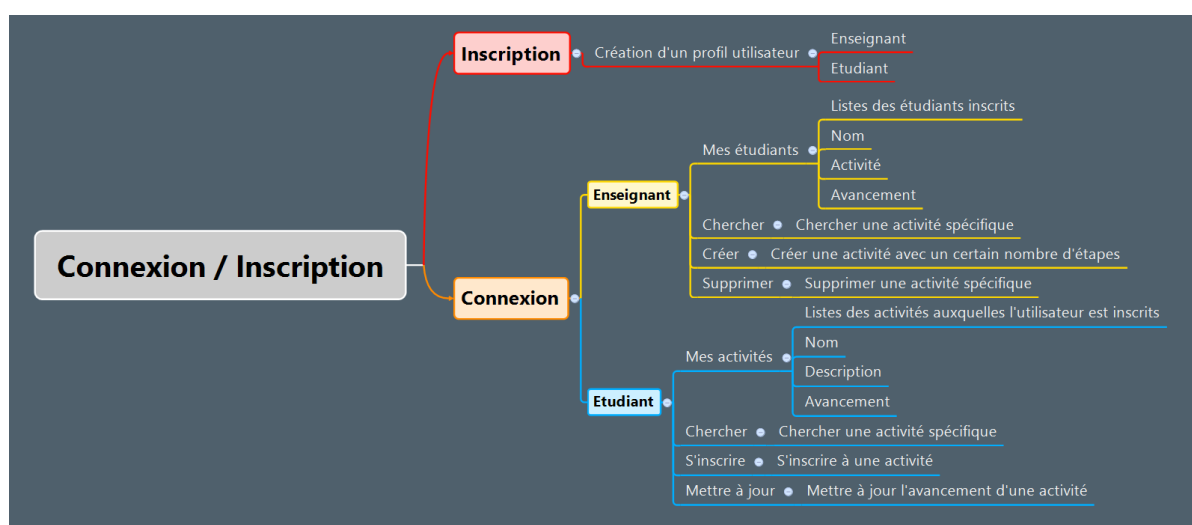


Figure 9: diagramme des fonctionnalités de l'application Web

## VIII. Base de données

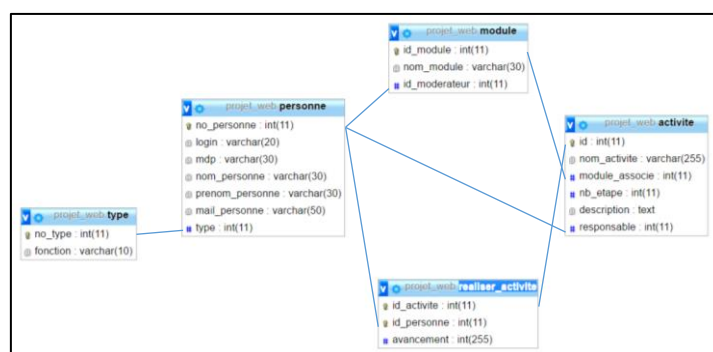


Figure 10: Diagramme de la BDD

Voici le diagramme représentant l'organisation de la base de données.

## IX. Gestion de projet

### 1. Communication avec le client

La communication avec le client aurait dû être plus présente, c'est une évidence. Il y a eu 3 rencontres avec le client au cours du projet :

- La première : au début du projet afin d'établir le cahier des charges initial.
- La seconde : à la moitié du projet pour discuter des difficultés rencontrées dans la maîtrise des langages HTML et PHP.
- La troisième : peu de temps après le deuxième pour faire un point sur l'avancement.

Tout au long du projet, le code source de l'application était mis en partage sur la plateforme GitHub. Ainsi le client pouvait, à tout moment, connaître l'avancement du projet. C'était également l'occasion pour le développeur de laisser des commentaires sur les différentes parties du code source.

### 2. Planning

Ce projet a débuté le 13 janvier 2017 et s'est terminé le 3 mai 2017 par une soutenance orale. Ce projet a été ponctué de 4 deadlines imposées :

- Jeudi 26 Janvier : rendu du cahier des charges
- Vendredi 31 Mars : rendu de l'état d'avancement du projet
- Jeudi 27 Avril : rendu du projet final
- Mercredi 3 Mai : soutenance orale du projet

Dès le début la planification s'est faite en trois phases :

- *La phase d'approche* : établissement du cahier des charges et formation au nouveau langage.
- *La phase de développement* : codage de l'application.
- *La phase finale* : dernière partie permettant de faire les dernières améliorations, de rédiger le rapport final et de préparer la soutenance.

Ci-dessous vous pouvez voir le planning prévisionnel établi au début du projet.

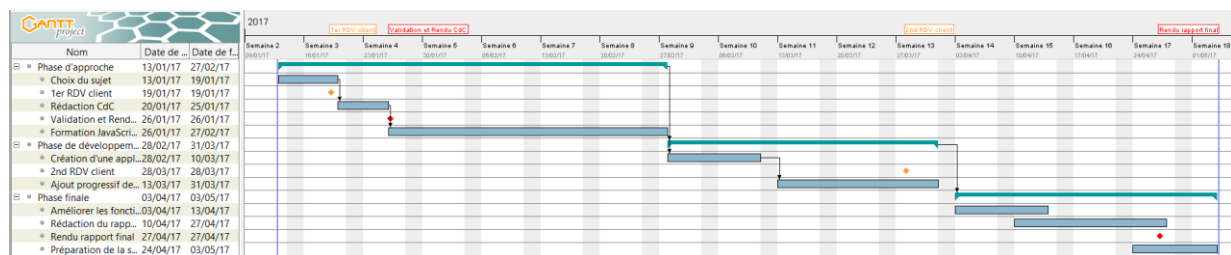


Figure 11: planning prévisionnel du projet

Dans le chapitre suivant, je vous explique pourquoi ce planning n'a pas été complètement respecté.

## X. Mise en œuvre technique

### 1. Règles de code

Afin d'avoir un code source le plus lisible possible, il a été décidé d'adopter une convention de nommage des différentes variables. De plus les différents scripts HTML et PHP sont nommés de façon suffisamment claire pour qu'un développeur externe au projet puisse naviguer sans mal entre les différents scripts.

## 2. Problèmes rencontrés

Durant la réalisation de ce projet, j'ai rencontré quelques problèmes, causant des contretemps et des difficultés plus ou moins importants.

Le plus difficile à résoudre fut la prise en main des langages HTML et PHP. En effet, lorsque durant la phase d'approche je me suis penché sur le langage JavaScript, je me suis vite rendu compte qu'il fallait parfaitement maîtriser les langages HTML et PHP. Bien que ces langages aient été abordés l'année passée en cours de développement Web, je me suis rendu compte que je n'avais acquis que les bases. J'étais totalement incapable d'afficher la moindre page web sans me replonger dans mes cours. Face à mon incompetence, j'ai rapidement pris la décision de me reformer complètement en HTML et PHP et de laisser de côté le JavaScript pour quelques temps.

Vous l'aurez compris ce problème m'a réellement empêché d'avancer dans le projet. Mais petit à petit j'ai pu créer les bases du projet tel qu'il est aujourd'hui ; à savoir la base de données sur PhpMyAdmin, une interface de connexion, un affichage différencié, etc.

Le temps passait et l'heure était venue de faire un point sur l'avancement du projet. Le client (Mr Pesquet), m'a rassuré sur mes capacités et m'a conseillé d'utiliser différents outils auxquels je n'avais pas pensé ou dont je n'avais pas connaissance. Ces outils, tels que Bootstrap, m'ont permis de réellement aller plus vite, surtout dans l'élaboration du CSS.

Face à cette énorme perte de temps, j'ai dû abandonner l'idée de terminer ma formation sur JavaScript ainsi que de mettre en place un hébergement en ligne de type SaaS du site. A l'heure actuelle, le site est hébergé en serveur local.

## 3. Pistes d'amélioration

Par conséquent, il y a encore beaucoup de points à améliorer pour cette application. Le principal étant de pouvoir héberger le site en ligne et ainsi éviter. D'autre part, l'application est loin d'être parfaitement fonctionnelle. Il existe de nombreuses pistes d'amélioration, à commencer par la mise en place des dernières fonctionnalités comme la possibilité de valider ou refuser l'inscription d'une personne à une activité, ou de sélectionner un étudiant et afficher les détails le concernant : toutes les activités auxquelles il est inscrit, le degré d'avancement pour chacune de ses activités. Il serait également intéressant de faire apparaître plus de détails concernant l'avancement d'un étudiant dans une activité : sa vitesse d'apprentissage, sa régularité dans le travail, etc.

Lors de l'inscription à une activité, il serait préférable que l'étudiant puisse avoir accès à la description de l'activité avant de finaliser son inscription.

D'autre part, l'aspect esthétique de l'application laisse encore beaucoup à désirer. Il est très important de faire un affichage qui suit une échelle colorimétrique de façon à permettre à l'enseignant de repérer d'un seul coup d'œil les étudiants en difficultés. De même, cela permettrait à l'étudiant de repérer d'un seul coup d'œil les activités dans lesquelles il est le moins avancé.

## XI. Conclusion

Ce projet informatique m'aura indubitablement appris qu'il ne faut rien prendre pour acquis. La gestion du projet de A à Z fut quelque peu périlleuse mais j'ai finalement réussi à redresser la barre juste à temps. J'ai ainsi pu prendre du recul par rapport à la gestion de projet et ses différentes étapes. Avant de commencer j'étais un peu réticent à l'idée de réaliser un projet entièrement seul. Je suis finalement content de l'avoir fait. Et ce qui est sûr c'est que je n'aborderais pas les prochains projets en solitaire de la même façon.

Si je dois tirer une leçon de mes erreurs sur ce projet, c'est qu'à l'avenir je dois instaurer une communication plus active, régulière et efficace avec le client. Je dois d'abord aborder le projet dans sa globalité et entrer de plus en plus dans le détail au fur et à mesure de sa réalisation. Chose qui n'est pas facile à faire car j'ai tendance à me concentrer sur les points sur lesquels je bloque.

D'un autre côté, j'espère avoir plus de temps au quotidien pour les projets que je réaliserais à l'avenir. Parce que durant ce semestre, nous devons parfois jongler entre 4-5 projets en parallèle, dans des langages qui n'ont parfois rien à voir (C#, Prolog, HTML, etc.) et je trouve cela extrêmement compliqué et perturbant, surtout lorsqu'on ne les maîtrise pas (Prolog, JavaScript, etc.).

Pour finir, j'aimerais juste remercier Monsieur Pesquet qui m'a apporté son soutien quand je doutais sérieusement de mes capacités à pouvoir fournir ne serait-ce qu'une seule page Web.