

Ammar MOIZALY
Mehdi ABDALLAOUI
Lyess ICHALALEN
Youssef KANOUNI



RAPPORT DE PROJET WEB ENSIIE 1A 2019



CRÉATION D'UN SITE WEB POUR DISCUTER SUR DES TOPICS EN GROUPE AU SEIN DE L'ÉCOLE

Tuteurs du projet :

Thomas COMES - Nassim Kirouane - Rémi Parpaillon

Table des matières

1 - Présentation du projet	2
2 - Conception	3
Langages et choix des outils utilisés	3
Base de données	3
Les différentes étapes du développement	4
a) La Programmation	4
b) Le Design	4
Résultats obtenus	4
Problèmes rencontrés et solutions apportés :	6
3 - Propositions d'amélioration	7
4 - Répartition des tâches au sein de l'équipe	7
5 - Conclusion	8

1 - Présentation du projet :

Chaque année, l'ENSIIE reçoit une centaine d'étudiants qui généralement ne se connaissent pas, ce qui représente une gêne pour la majorité, pour les étrangers comme pour les français. Souvent, les premiers mois sont un peu durs puisqu'il est vrai qu'il faut à la fois s'adapter au niveau d'études mais aussi au niveau relationnel.

La solution "MEETIIE", comme son nom l'indique, consiste à mettre en place un chat real-time pour les nouveaux arrivants, afin de les aider de faire de nouvelles connaissances rapidement et plus facilement. En addition, elle met en disposition plusieurs salons leur permettant de discuter avec tous les liens de l'école sur la plateforme selon la nature du sujet.

La provenance de chacun de notre groupe met en évidence l'utilité de notre fameux projet, vu qu'on y retrouve déjà 3 nationalités : Maroc, Madagascar et Algérie, et donc 3 cultures distinctes. Grâce à cette solution, nous contribuons à la facilité de s'ouvrir à de nouvelles cultures.

Finalement, et en concordance avec nos acquis de formation, nous avons pu mettre en lumière quelques observations en titre de conclusion sur quelques points concernant le développement web d'une solution aussi importante comme la nôtre.

En premier lieu, nous présentons le projet. En deuxième lieu, nous nous focalisons sur sa conception et les propositions d'amélioration. En dernier lieu, nous concluons par les tâches effectuées.

2 - Conception :

Langages et choix des outils utilisés :

Combinaison de PHP, Javascript, HTML et CSS, en implémentant le design pattern MVC.

On a utilisé la bibliothèque socket.io pour implémenter les discussions en temps réel, c'est l'une des bibliothèques les plus prisées par les développeurs Node.js, de laquelle on a pu tirer profit après avoir lancé la commande **"npm start"** sur le terminal de Docker.

Base de données

```
CREATE TABLE "member" (  
  id SERIAL PRIMARY KEY,  
  firstname VARCHAR(30) NOT NULL,  
  lastname VARCHAR(30) NOT NULL,  
  email VARCHAR(255) NOT NULL,  
  password VARCHAR(40) NOT NULL,  
  admin boolean not null default false,  
  banned boolean not null default false  
);
```

La base de données contient une table "member" dédiée à tous les liens inscrits sur notre plateforme.

Elle contient les attributs classiques : "id", "firstname", "lastname", "email", "password".

En addition, on retrouve 2 booléens : "admin" qui représente le statut de chacun des utilisateurs, et également "banned" pour savoir si un certain utilisateur est banni ou pas.

```
create table "chat" (  
  id SERIAL PRIMARY KEY,  
  member1 integer references member(id),  
  member2 integer references member(id),  
  startDate timestamp  
);  
  
create table "message" (  
  id SERIAL PRIMARY KEY,  
  sender integer references member(id),  
  chat integer references chat(id),  
  sendTime timestamp not null,  
  message VARCHAR(255) NOT NULL DEFAULT ''  
);
```

Différentes étapes du développement :

A) Programmation

Etape I : Programmation du front-end, tout ce qui concerne le design et donc l'IHM.

Etape II : Vérification et validation des champs des formulaires avec Javascript.

Etape III : Programmation du chat en temps réel.

B) Design

Pour la couleur de l'arrière plan, nous avons choisi une image avec un dégradé du vert au bleu. Nous avons choisi cette couleur car elle sort du lot, ainsi ça permet de donner une identité à Meetiie.

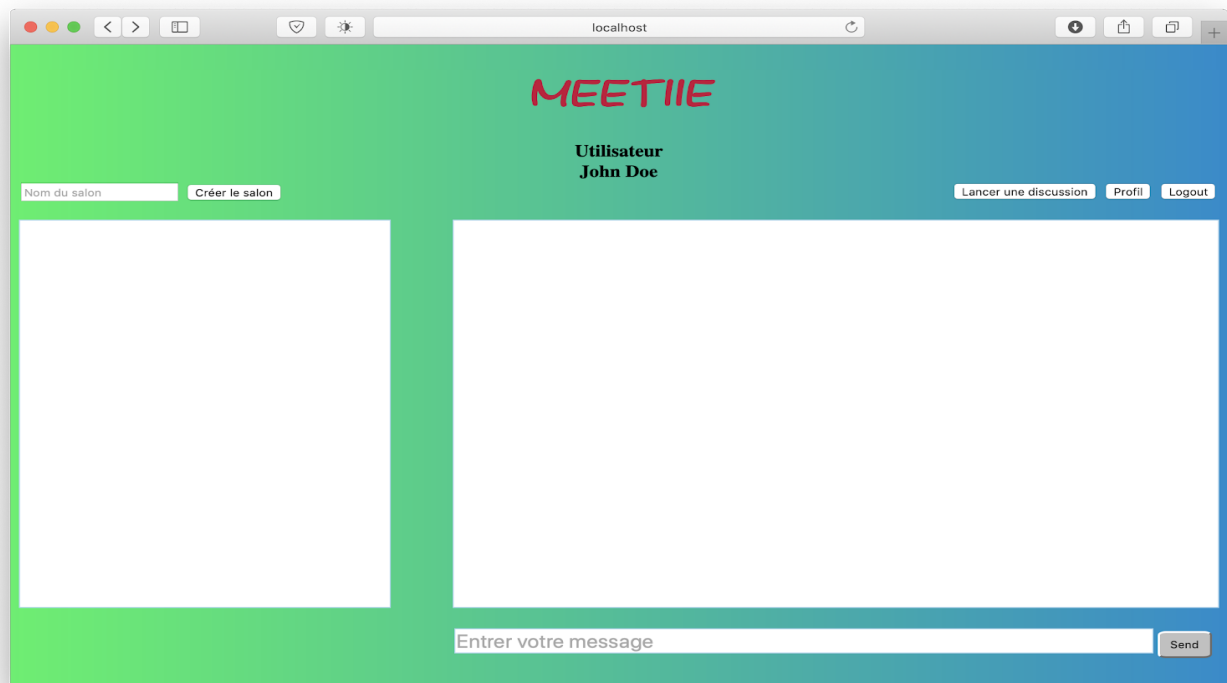
La page principale contient une barre de recherche pour faciliter la navigation à l'utilisateur. Elle contient également deux zones distinctes, la première est conçue pour l'affichage des salons actuellement ouverts et ainsi permettre de participer à une discussion en salon avec un simple clic. La deuxième zone est dédiée aux messages, sa partie supérieure affiche les messages et l'autre inférieure sert à envoyer un message.

Résultats obtenus :

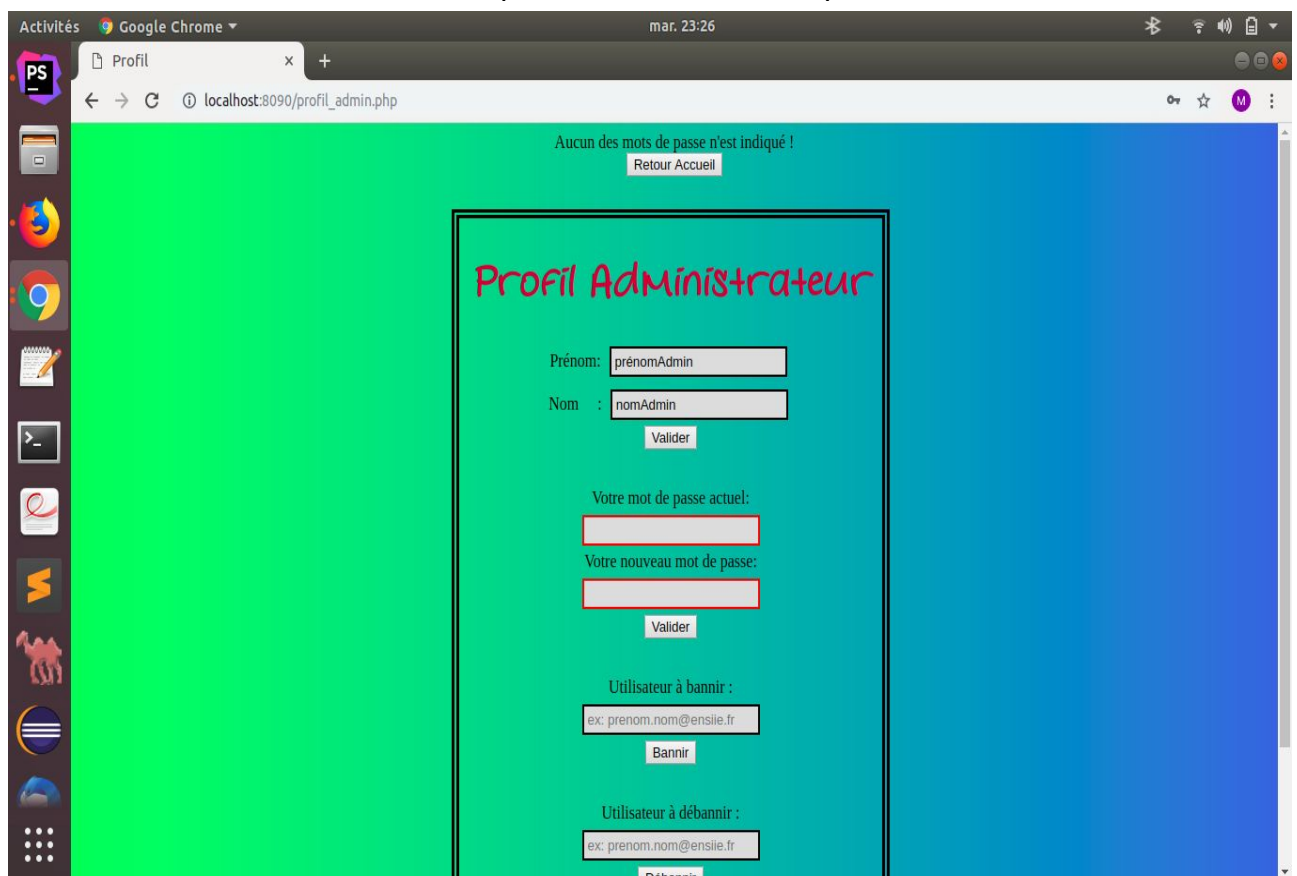
Les objectifs principaux de notre projet ont été réalisés avec succès. Cependant, d'autres améliorations sont en cours de construction telle que la possibilité de créer des salons pour discuter des différents sujets souhaités dans la page principale.

Nous avons consacré du temps à la recherche et l'apprentissage des nouvelles technologies afin d'implémenter un chat en temps réel.

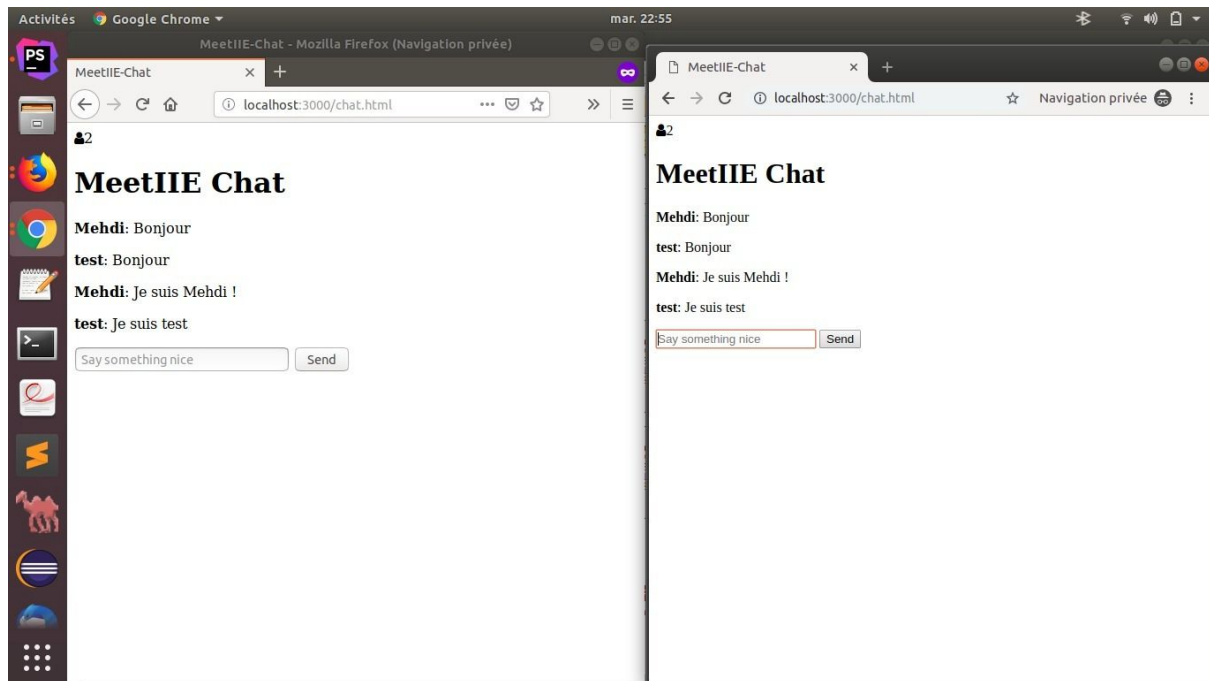
Capture de la page principale après l'authentification :



Vérification de la validité des champs à l'aide du JavaScript :



Capture du chat en temps réel sur deux navigateurs différents avec deux comptes utilisateurs différents :



Problèmes rencontrés et solutions apportés :

Trouver une technologie pour les discussions en temps réel :

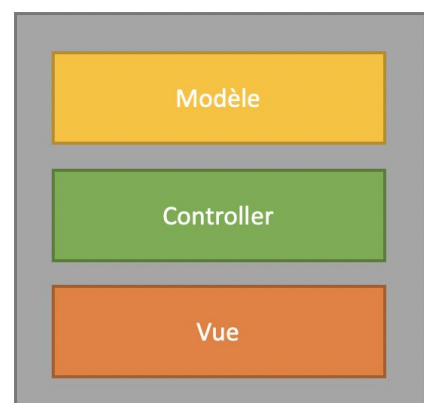
Comme aucun de nous n'avait jamais fait un site avec un échange de donnée en temps réel, nous avons consacré du temps pour le choix de la technologie. On c'est vite rendu compte que Websocket de la bibliothèque socket.io de Node.js était la plus utilisée pour cette solution. Nous avons donc choisi cette solution et appris à utiliser Websocket chacun de notre côté.

Configuration du serveur SMTP :

Configuration du serveur SMTP qu'on avait réussi à mettre en place sur le localhost, mais on a pas eu assez de temps pour l'implémenter sur Docker. La raison pour laquelle on l'a utilisé est d'envoyer le mot de passe à la boîte email de l'utilisateur une fois il clique sur mot de passe oublié et qu'il remplit le formulaire et valide par reset.

Structuration du code :

Au début, chacun avait programmé à sa manière et on s'est vite rendu compte quand arrivait plus à se lire entre nous. Ce qui fait, après la formation de MVC sur openclassroom dont notre encadrant nous avait conseillé de lire, et ses conseils, on a décidé de programmer en se basant sur le Design Pattern



MVC : Les accès à la base de données en PHP en haut de la page, puis le contrôle des boutons et formulaires en PHP et en Javascript pour le côté client, et enfin en HTML l’affichage de la page.

3 - Propositions d’amélioration :

- Possibilité d’activer la webcam lors d’une discussion.
- Possibilité de créer des groupes de projets pour collaborer avec des outils sur le site. Par exemple programmer des réunions de groupes.
- Ajouter un champ dans la table SQL afin de confirmer la lecture des messages reçus.

4 - Répartition des tâches au sein de l’équipe :

Nom de tâche	effectué par :
Front-end	Youssef et Lyes
Back-end	
Formation en MVC sur openclassroom	Ammar + Mehdi
Page de connexion	Mehdi
Création de compte	Ammar
Logout	Ammar
Page principale	Mehdi
Profil	Ammar
Base de données	Ammar + Mehdi

5 - Conclusion :

Grâce à ce projet on a pu consolider notre niveau en php, html, javascript, postgresSQL. Mais également pu découvrir de nouvelle technologie tel que les Websocket, connaître un peu plus sur les serveurs SMTP et la programmation en suivant le design pattern MVC.

Nous avons également pu avoir une expérience en plus du travail en équipe. On a su partager le travail selon les points forts de chacun et travailler ensemble pour résoudre les problèmes les plus compliqués.

En conclusion, ce projet fut d'une grande aide car il nous a dans une certaine mesure préparé à travailler en collaboration pour des projets en entreprise.