

COMPTE RENDU PROJET WEB

JONATHAN BOAKNIN
BENJAMIN FOURNIER
MACIL GUIDDIR
ALIX MATHIEU

me9tiie

[ENSIIE]
[21 / 05 / 2018]

SOMMAIRE

Répartition des tâches	2
Introduction	2
Fonctionnement du site	3
Difficultés surmontées	5
Conclusion	6

REPARTITION DES TACHES

Concernant la répartition des tâches, nous avons utilisé un Trello car cela permet d'avoir un document clair qui récapitule « qui fait quoi ? », « quelles sont les deadlines ? ».

Voici l'adresse de notre Trello : <https://trello.com/b/zfwxgaHx/projet-web>

INTRODUCTION

Dans le cadre de l'UE dédiée aux projets lors du 2ème semestre, un projet Web nous a été proposé. Pour celui-ci, voici les 4 sujets au choix : 3 sujets prédéfinis et le dernier qui correspond à un sujet libre. Notre groupe a choisi ce dernier. En effet, nous souhaitons laisser parler notre imagination et faire émerger une idée qui nous était propre. De plus, choisir ce sujet nous a permis aussi de nous démarquer et de proposer le concept, tel que nous l'avons imaginé et ce de façon inédite.

Est venu ensuite un moment de « Brainstorming » afin que chacun expose ses idées et qu'un sujet final soit décidé et accepté par l'ensemble de notre groupe. Le concept du site de rencontre a été notre première idée, suite à une conversation que nous avons eue quelques heures avant.

C'était en effet le sujet le plus apprécié car nous pensions que cela serait un concept vraiment fun au sein de l'école et qui pouvait être élargi à un plus grand public d'utilisateurs si l'idée allait plaire.

Cependant, dans un site de rencontres classique, chaque utilisateur a un profil contenant des informations explicites le concernant (nom, prénom, date de naissance, photo...) et qui est visible par tous les autres utilisateurs. Reprendre exactement le même concept n'allait pas être aussi drôle pour notre école. En effet, au vu de la taille des promotions (150 environ), le nombre d'étudiants n'est pas gigantesque et la plupart d'entre eux se connaissent. C'est pourquoi, nous nous sommes posés la question suivante :

« Comment adapter le concept du site de rencontre à l'ENSIIE ? »

FONCTIONNEMENT DU SITE

Le but du site est, bien entendu, que les utilisateurs s’amusent et qu’ils découvrent leurs camarades au travers de notre application et non pas le contraire (car si c’était le cas, les utilisateurs se lasseraient très vite de celle-ci).

❖ La page principale :

Un texte de présentation aux traits humoristiques ainsi qu’une image en arrière-plan sont les éléments incontournables de la vitrine de notre site. Ils ont pour but de pousser l’utilisateur à s’inscrire sur tous les plans : l’image pour la forme et le texte pour le fond. De même, le site fait appel aux “envies” de l’utilisateur en lui posant, d’emblée, la question suivante : “Vous recherchez...[Homme ou Femme]”. En plus, cette question est posée avant celle du sexe de l’utilisateur ! C’est une première, puisque les autres sites de rencontre - les plus connus, du reste - ne procèdent pas ainsi. De même, la couleur de fond (dégradé violet/fuschia) a pour but de mettre immédiatement l’utilisateur dans l’ambiance dans laquelle le site va le plonger tout au long de son expérience.

Aussi, un compteur qui recense le nombre de personnes inscrites sur le site, en temps réel, a été placé sur la page d’accueil. Il doit permettre de donner envie aux liens de s’inscrire, comme leurs collègues. Cette idée a été empruntée au site *Badoo*. Elle nous semblait être un moyen efficace de persuader un visiteur du site de sauter le pas.

❖ Le profil :

En ce qui concerne le profil (élément important puisqu’il est propre à chaque utilisateur), nous voulions qu’un utilisateur n’ait pas accès aux informations générales sur chaque autre utilisateur. Nous avons donc décidé qu’un numéro soit attribué à chaque utilisateur et que ce numéro serait son identifiant. De plus, aucune photo n’apparaîtra sur le profil visible par tout le monde.

De même, pour alimenter un peu plus le profil visible par tout le monde, nous avons décidé d’y ajouter des tags : ces tags permettent à l’utilisateur de se décrire en quelques mots (Quel est son caractère ? Quelles sont ses passions ? Quels sont ses goûts ?). De plus, chaque utilisateur pourra se décrire tel qu’il le souhaite puisque les tags sont libres et non prédéfinis.

❖ Inscription et désinscription :

L’inscription se fait via les critères suivants : le sexe recherché, puis le sexe de l’utilisateur, l’adresse mail, la date de naissance et le mot de passe. Ensuite, un mail est envoyé à l’utilisateur, lui demandant de valider son inscription. Une fois inscrit, l’utilisateur peut se connecter à tout moment via le bouton “se connecter”, en haut à droite de la page d’accueil.

Quant à la désinscription, elle est précédée de quelques questions, écrites par les administrateurs. Ces questions peuvent être ajoutées, modifiées, supprimées à notre guise. Elles sont une sorte de feedback pour nous puisqu’elles nous permettent de

comprendre ce qui pousse l'utilisateur à se désinscrire. Ainsi, nous pourrions améliorer le site, bannir certaines personnes (si le tort a été prouvé) ou seulement faire des sondages. Par exemple, la question de désinscription "Avez-vous trouvé ce que vous recherchiez ?" nous permet de quantifier l'efficacité du site.

❖ Fonctionnalités :

Puis, nous avons réfléchi aux fonctionnalités que nous pourrions mettre en place pour notre application web, de sorte que l'utilisateur accroche à notre site et qu'il le recommande à ses amis tout en respectant les consignes imposées.

Ainsi, nous avons décidé de mettre en place des "matches" basés sur le même principe que le site *Tinder* : en effet, si un utilisateur X a un match avec un utilisateur Y, cela signifie que X est intéressé par le profil de Y et réciproquement.

Ces matches, vu la taille de l'école, ne se font pas via un système de recherche, mais pas un système de "suggestions". Ces suggestions sont basées principalement sur deux critères : sexe recherché et tags (adjectifs qualifiant l'utilisateur). Nous avons eu l'idée de plusieurs autres types de suggestion, tels que la ville d'origine. En effet, les liens viennent de toute la France. Cependant, nous avons pensé qu'une grande partie des utilisateurs auraient un nom de ville (ou village !) unique, ce qui pourrait les identifier d'emblée. Nous avons ensuite pensé élargir au département, ou à la région, mais nous avons conclu que ce critère restrictif irait à l'encontre de la volonté de connaître véritablement l'Autre.

Ainsi, tout utilisateur peut accéder à tous ses matches depuis un onglet "Tableau de chasse" (nous avons, une nouvelle fois, fait quelques traits d'humour à travers les titres des onglets, le but étant que ce site amuse l'utilisateur et qu'il veuille le partager à ses camarades). En outre, après avoir matché avec une personne, chaque utilisateur a la possibilité de discuter avec elle, en direct grâce au système de "chat" que nous avons mis en place. Cette fonctionnalité supplémentaire permet ainsi aux utilisateurs qui ont effectué la première étape de leur rencontre (par intérêt mutuel) de pouvoir apprendre à se connaître davantage. Toujours dans l'optique de garder une certaine forme d'anonymat, seul l'identifiant de l'utilisateur apparaîtra dans la fenêtre de chat. Cela permet ainsi aux utilisateurs de ne jamais connaître le nom, le prénom, ou d'autres informations sur la personne avec qui il discute.

Par ailleurs, une newsletter accueille l'utilisateur après sa connexion. Elle a pour but primitif d'informer le Meettier des mises à jour effectuées sur le site (elles seront nombreuses à venir !) ; mais elle permet aussi de faire quelques blagues et d'offrir quelques anecdotes sur l'utilisation du site, ou peut-être sur les performances remarquables de certains de ses utilisateurs !

DIFFICULTES SURMONTÉES

L'aspect du site a été majoritairement conçu sur la base du **framework Bootstrap**, ce qui nous a permis de gagner du temps au niveau du codage pour se concentrer plus sur le Back-End. Les fichiers de codes sont organisés selon le **pattern MVC**. Il permet une structuration claire du code en 3 parties simples. L'accès à la base de données se fait via **PDO**.

En ce qui concerne la partie Modèle, nous avons décidé de créer une classe pour chaque fonctionnalité.

La première difficulté rencontrée est l'utilisation de la fonction mail de **PHP** qui nécessite une configuration supplémentaire en fonction du fournisseur d'accès utilisé. Cette fonction est indispensable car elle permet de confirmer l'inscription et ainsi d'éviter d'avoir des faux comptes.

Ensuite, en ce qui concerne la validité des données, il a fallu utiliser plusieurs méthodes. Tout d'abord, les **regex** pour vérifier le bon format du mail fourni par l'utilisateur puis nous avons dû utiliser une **fonction de hachage** pour le mot de passe afin de le cacher en cas d'attaque par exemple dans la base de données. Pour chaque donnée reçue par l'utilisateur, on utilise une fonction **htmlspecialchars** pour éviter les failles de type **XSS**.

Une autre difficulté était de trouver une animation sympathique lorsqu'un utilisateur atteint un nouveau palier. Pour ce faire, nous avons d'abord téléchargé une image de coeur qui correspond bien à l'image globale du site. Puis, on a décidé de la faire tourner depuis le centre de l'écran vers les bordures pour annoncer à l'utilisateur que quelque chose arrive. Pour cela, on a utilisé un type de figure appelé spirale logarithmique. Il suffisait simplement de récupérer les équations déjà établies et de modifier la position à un instant donné.

Puis, évidemment, plusieurs détails techniques ont été corrigés. Par exemple, un bouton stylisé "se connecter" semblait alourdir le site, et il fallait appuyer trois ou quatre fois pour que la page de connexion soit envoyée. Nous avons donc enlevé le bouton et laissé un simple lien "se connecter", sans habillage stylistique, et le problème a été réglé.

Il fallait ensuite s'attaquer à la mise en place d'un chat. Nous avons donc utilisé **Ajax** en passant par le **framework JQuery**. Cependant, ce n'est pas un chat en temps réel car les requêtes Ajax ne sont pas instantanées. Pour chaque appel **Ajax**, on récupère des données de la base **PostgreSQL**. C'est un système lourd. De ce fait, on ne peut récupérer les messages qu'avec un retard de 2 secondes à peu près. On pourrait en effet réduire ce retard mais le serveur ne supporterait pas autant de requêtes HTTP et pourrait conduire à une erreur 502 ou bien considérer autant de requêtes HTTP comme une attaque du serveur. Il aurait fallu utiliser une technologie comme NodeJS ou un équivalent utilisant un système de socket pour créer un vrai chat en temps réel.

Un dernier problème rencontré a été le **Docker** lui-même. Travaillant sur Windows, même avec un **bash Ubuntu** installé, nous avons rencontrés des problèmes lors de l'installation. Par conséquent nous avons dû travailler dans un premier temps avec **MySQL**. Il a fallu ensuite transformer le script **MySQL** en **PostgreSQL**.

CONCLUSION

Pour conclure, ce travail mêle à la fois un esprit d'équipe et la mise en commun de compétences que nous avons acquises au cours de l'année. En effet, il a fallu répartir équitablement les tâches en fonction des connaissances de chacun.

De plus, ce fut une expérience vraiment enrichissante, tant du point de vue humain (travailler seul derrière son ordinateur, ce n'est pas aussi simple que travailler par équipe de 4, puisqu'il a fallu nous mettre d'accord à plusieurs reprises sur les choix auxquels nous avons été confrontés), que du point de vue technique : nous avons dû créer notre site de A à Z contrairement aux travaux pratiques que nous avons faits au cours de l'année, où il fallait juste compléter quelques "morceaux" de code.

Nous espérons maintenant démocratiser l'utilisation de ce site qui, sans ses utilisateurs, n'est rien. Il sera constamment amélioré et interagira avec l'utilisateur via la newsletter.