

INFO-F-308 : Application Vivre ensemble

Plan de travail

Abdelkefi Amin	Baudoux Nicolas	Bauwin Lucie
Boulif Ilias	Dewit Maxime	Minhas Prabhdeep
Requena Carlos	Van Palm Geoffrey	Zhou Mi

Année académique 2016-2017

1 Calendrier d'avancement

Tâche | Equipe | Date

2 Description des tâches

2.1 Back-end général

2.1.1 Algorithme de match

L'algorithme affichant les personnes à rencontrer se basera sur les informations fournies lors des questionnaires remplis précédemment par les utilisateurs. Il établira un panel de 5 personnes ayant un pourcentage de différences plus ou moins grand. Nous définirons 5 intervalles de pourcentage (0%-20%, 21%-40%, 41%-60%, 61%-80% et 81%-100% de différences). Dans le cas où l'algorithme est dans l'incapacité de trouver au moins une personne appartenant à un interval précis, il est prévu d'agrandir cet interval afin de faire correspondre une autre personne. Le choix des personnes se ferait aussi de manière à ne pas pouvoir retomber sur une personne déjà rencontrée.

2.1.2 Limitation du nombre de choix

Par jour, l'utilisateur aura la possibilité de choisir parmi 5 fiches de personnes pour désigner la personne qu'il aimerait rencontrer. Ces 5 fiches sont données par l'algorithme de match présenté ci-dessus. Celui-ci renvoie des personnes avec un pourcentage différent pour permettre à l'utilisateur de choisir parmi des personnes ayant de nombreux points en commun ou non.

Sur chaque fiche, l'utilisateur verra l'avatar de la personne accompagné de son surnom, de son niveau dans le jeu, ainsi que le pourcentage de ressemblance avec celui-ci. Ce sont donc les informations disponibles sur la fiche qui permettront à l'utilisateur de choisir sa prochaine rencontre.

Une fois la personne choisie, cette-dernière recevra une notification pour lui demander si elle accepte cette rencontre ou non.

2.1.3 Classement

Après avoir effectué une rencontre, l'utilisateur sera crédité d'un montant de points. Le nombre de points étant en lien avec le pourcentage de différence (plus de différences, plus de points). La somme des points donnera à l'utilisateur une position dans un classement. L'accumulation de ces points donnera accès à des "niveaux".

L'équipe Back-end implémentera donc différentes fonctions calculant les points marqués, actualisant le nombre total de points des utilisateurs et leur niveau.

2.1.4 Confirmation de rencontre

Afin de confirmer une rencontre entre deux utilisateurs, ceux-ci devront remplir un quiz supplémentaire. Ce quiz a pour but de savoir si l'image qu'ils s'étaient fait l'un de l'autre est correcte.

Ensuite, ils reçoivent chacun un code et doivent se le partager pour valider la rencontre. Une fois la rencontre validée, ils sont ajoutés dans leur liste d'amis respective où ils auront accès à une messagerie instantanée et au profil de l'utilisateur rencontré afin de conserver un contact si la rencontre leur a plu.

2.2 Base de données

La base de donnée sera stockée en ligne sur un serveur à distance, en langage SQL. Celle-ci sera composée de différentes informations sur les utilisateurs (nom, prénom, sexe, études, loisirs, ...).

A l'inscription, chaque utilisateur remplit un formulaire de base, a une liste d'amis vide et génère un score de 0. Un pseudo aléatoire lui est également attribué. Les utilisateurs de cette application ont la possibilité de modifier ses données personnelles. Après chaque rencontre, les données sont actualisées. Si le score est assez élevé, le joueur augmente de niveau.

De plus, la base de données est l'endroit où sont enregistrés les lieux de rencontre, ainsi que les 3 choix de personnes à rencontrer par jour.

2.3 Front-end

Dans l'optique de donner un visage à notre application, nous allons utiliser divers frameworks tels que Bootstrap et React.

La collection d'outils Bootstrap nous permet de créer des interfaces 'mobile-first' avec 'responsiveness', c'est à dire, qui réagit bien aux différents appareils et leur caractéristiques telles que la résolution.

React est une librairie JavaScript qui facilite la construction des composants web.

Pour le système de communication, le service externe Twilio sera utilisé. Twilio offre la possibilité aux utilisateurs de communiquer entre eux à travers leur API, en utilisant leur infrastructure.

Le système d'authentification bénéficiera des données de l'ULB pour restreindre l'accès à la communauté universitaire.