

Communauté de services - Backlog



Nous souhaitons créer un site internet **communautaire dédié aux échanges de services**.

Le principe est simple, un utilisateur s'inscrit tout d'abord en spécifiant un email, un mot de passe ainsi qu'un pseudo unique.

Lors de sa première connexion, il sera invité à compléter son profil. Par exemple son adresse, son numéro de téléphone, etc...

Les utilisateurs pourront rechercher les services suivant différents critères de recherche afin de proposer leur aide.

Lorsqu'un utilisateur souhaite rendre service à un autre utilisateur, il s'inscrit alors et le service en question ne sera plus visible dans la recherche. A partir de ce moment, l'utilisateur qui propose ses services peut envoyer et recevoir des messages à l'utilisateur qui demandait de l'aide.

Suite à cela, l'utilisateur devra alors se rendre au point de rendez-vous à l'heure et la date indiquée.

Nous souhaitons, dans un premier temps, créer uniquement la base de données et les requêtes associées.

Nous souhaitons une base de données qui sera au maximum optimisée afin que les différentes requêtes se fassent le plus rapidement possible.

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX créer une base de données MySQL

AFIN DE pouvoir ensuite créer et tester mes requêtes

Critères d'acceptation

- La base de données MySQL comprend les tables et les colonnes suivantes :
 - **Utilisateur**
 - Identifiant de la table – **Ne peut pas être NULL**
 - Email de l'utilisateur – **Unique – Ne peut pas être NULL**
 - Mot de passe – **Ne peut pas être NULL**
 - Pseudo – **Unique et ne peut pas être NULL**
 - Adresse postale – **Peut être NULL**
 - Code postale – **Peut être NULL**
 - Ville – **Peut être NULL**
 - Pays – **Peut être NULL**
 - Numéro de téléphone portable – **Peut être NULL**
 - Numéro de téléphone fixe – **Peut être NULL**
 - Date d'inscription – **Peut être NULL**
 - **Service**
 - Identifiant de la table – **Ne peut pas être NULL**
 - Identifiant de l'utilisateur – **Ne peut pas être NULL**
 - Nom du service – **Ne peut pas être NULL**
 - Description du service – **Ne peut pas être NULL**
 - Adresse du service – **Ne peut pas être NULL**
 - Code postale du service – **Ne peut pas être NULL**
 - Ville du service – **Ne peut pas être NULL**
 - Pays du service – **Ne peut pas être NULL**
 - Date et heure du service – **Ne peut pas être NULL**
 - Informations complémentaire – **Peut être NULL**
 - **Service_Utilisateur** (permet de savoir quel utilisateur est inscrit à quel service)
 - Identifiant de la table – **Ne peut pas être NULL**
 - Identifiant du service – **Ne peut pas être NULL**
 - Identifiant de l'utilisateur – **Ne peut pas être NULL**
 - Date et heure de l'inscription – **Ne peut pas être NULL**
 - **Message**
 - Identifiant de la table – **Ne peut pas être NULL**
 - Identifiant de l'expéditeur – **Ne peut pas être NULL**
 - Identifiant du receveur – **Ne peut pas être NULL**

- Contenu du message – **Ne peut pas être NULL**
- Date et heure d'envoi – **Ne peut pas être NULL**
- Vous veillerez à choisir le typage idéal pour chacune de colonnes.
- Différents fichiers sont créés pour stocker les requêtes :
 - « users.sql »
 - « services.sql »
 - « messages.sql »
 - « service_utilisateur.sql »
- La recherche d'un service se fera selon plusieurs critères. Lisez le backlog en entier afin de créer les index nécessaires à l'optimisation des recherches.

2ème Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX créer un jeu de données pour ma base de données

AFIN DE pouvoir ensuite créer et tester mes requêtes

Critères d'acceptation

- Le jeu de données comprendra au minimum :
 - 25 utilisateurs
 - 20 services
 - 10 inscriptions à un service
 - 30 messages envoyés
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

3^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX créer un utilisateur lors de la première inscription

AFIN DE pouvoir ensuite créer les requêtes pour proposer et participer à un service

Critères d'acceptation

- Lors de la première inscription, seul les champs suivant seront précisés :
 - Email de l'utilisateur
 - Mot de passe de l'utilisateur
 - Pseudo de l'utilisateur
 - Date d'inscription
- On imagine que le mot de passe sera ensuite hashé, il faut donc prévoir une chaîne de caractère d'une longueur suffisante (par exemple, 191 caractères). **Pour nos exemples, le mot de passe stocké en clair dans la base de donnée.**
- Le pseudo sera unique dans la base de données.
- L'adresse email sera unique dans la base de données.
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

4^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX mettre à jour mon profil lors de ma première connexion

AFIN DE pouvoir ensuite créer les requêtes pour proposer et participer à un service

Critères d'acceptation

- Le profil sera mis à jour avec les informations demandées à la 1^{ère} Story.
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

5^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX créer la requête pour créer un service et le proposer à la communauté

AFIN DE pouvoir ensuite créer la requête pour m'y inscrire

Critères d'acceptation

- Le profil sera mis à jour avec les informations demandées à la 1^{ère} Story.
- On considère ici que l'on connaît l'identifiant de l'utilisateur qui crée le service.
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

6^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX créer la requête qui permet de s'inscrire à un service

AFIN DE pouvoir ensuite communiquer avec l'utilisateur concerné

Critères d'acceptation

- On considère ici que l'on connaît l'identifiant de l'utilisateur qui souhaite s'inscrire.
- On considère ici que l'on connaît l'identifiant du service auxquels s'inscrire.
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

7ème Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX créer la requête qui permettra d'envoyer un message

AFIN DE pouvoir ensuite communiquer avec l'utilisateur concerné

Critères d'acceptation

- On considère ici que l'on connaît l'identifiant de l'utilisateur qui souhaite s'inscrire, à savoir son identifiant unique (id).
- On considère ici que l'on connaît l'identifiant du service auxquels s'inscrire, à savoir son identifiant unique (id).
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

8ème Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX afficher les conversation d'un utilisateur

AFIN DE pouvoir ensuite communiquer avec l'utilisateur de mon choix

Critères d'acceptation

- On considère ici que l'on connaît l'identifiant de l'utilisateur connecté
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.
- Les conversations seront affichées par ordre du dernier message reçu.

9ème Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX afficher les messages d'une conversation

AFIN DE pouvoir ensuite communiquer avec l'utilisateur de mon choix

Critères d'acceptation

- On considère ici que l'on connaît l'identifiant des 2 utilisateurs qui communiquent.
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.
- Les messages seront affichés par ordre de création (du plus récent au plus ancien).

10^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX rechercher la liste de tous les services que proposent le site

AFIN DE pouvoir ensuite m'y inscrire ou afficher le détail du service

Critères d'acceptation

- La recherche peut se faire suivant différents critères :
 - Le nom du service
 - Le code postal du service
 - La ville du service
 - Le pays du service
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.
- On affichera les services dont la date n'est pas encore passés.
- Si un utilisateur est déjà inscrit au service, il ne s'affichera pas dans la recherche.
- Les résultats seront affichés suivant leur date, ainsi que la ville. Du plus récent au plus ancien et par ordre alphabétique.

11^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX afficher le détail d'un service

AFIN DE pouvoir ensuite m'y inscrire

Critères d'acceptation

- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.
- On affichera les informations suivantes :
 - Toutes les informations du service
 - Le nom de l'utilisateur qui propose le service

- Le prénom de l'utilisateur qui propose le service
- Le numéro de téléphone de l'utilisateur qui propose le service
- Le nom et prénom de l'utilisateur inscrit au service **SI CELA EST LE CAS.**
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

12^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX supprimer un service proposé

AFIN DE de prévenir une indisponibilité

Critères d'acceptation

- On considère que l'on connaît l'identifiant du service
- La suppression du service entraîne la suppression de la ligne de l'utilisateur inscrit à ce service.
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

13^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX supprimer mon inscription à un service

AFIN DE de prévenir une indisponibilité

Critères d'acceptation

- On considère que l'on connaît l'identifiant du service
- On considère que l'on connaît l'identifiant de l'utilisateur qui souhaite supprimer son inscription.
- La suppression du service entraîne la suppression de la ligne de l'utilisateur inscrit à ce service.
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

14^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory
JE VEUX supprimer mon profil
AFIN DE de ne plus avoir de lien avec la communauté

Critères d'acceptation

- On considère que l'on connaît l'identifiant de l'utilisateur
- La suppression de l'utilisateur entraîne la suppression de **tous ses services, toutes ses inscriptions** ainsi que **tous ses messages envoyés**.
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

15^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory
JE VEUX supprimer un message
AFIN DE de corriger une erreur de frappe

Critères d'acceptation

- On considère que l'on connaît l'identifiant du message à supprimer
- Les requêtes en lien avec la messagerie seront placées dans le même fichier.

16^{ème} Story

Story

EN TANT QU'étudiant de la Coding Factory
JE VEUX récupérer l'historique des services auxquels un utilisateur aura participé.
AFIN DE pouvoir suivre l'utilisation que je fais du site internet.

Critères d'acceptation

- On affichera les informations suivantes :
 - Toutes les informations du service

- Le nom de l'utilisateur qui propose le service
- Les informations de l'utilisateur inscrits
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.
- Les résultats seront affichés suivant leur date, ainsi que la ville. Du plus récent au plus ancien et par ordre alphabétique.
- Le nombre total de service auxquels l'utilisateur a participé.

17ème Story

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX récupérer les informations du premier service auquel un utilisateur a participé.

AFIN DE pouvoir suivre l'utilisation que je fais du site internet.

Critères d'acceptation

- On affichera les informations suivantes :
 - Toutes les informations du service
 - Le nom de l'utilisateur qui propose le service
 - Les informations de l'utilisateur inscrits
- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.

18ème Story (Ultra Bonus)

Story

EN TANT QU'étudiant de la Coding Factory

JE VEUX récupérer le nombre de services participés sur une année par mois pour un utilisateur

AFIN DE pouvoir suivre l'utilisation que je fais du site internet.

Critères d'acceptation

- Les requêtes de création du jeu de données seront stockées dans le fichier adéquat.
- Le résultat s'affichera de la façon suivante :

Mois	Pseudo utilisateur	Participations total
01	Michel	5
02	Michel	6
02	Michel	20
04	Michel	0

05	Michel	1
06	Michel	6
07	Michel	5
08	Michel	7
09	Michel	8
10	Michel	2
11	Michel	1
12	Michel	3