

Création d'un réseau social d'entreprise : Cahier des charges

Principales fonctionnalités

- Création d'utilisateurs (CRUD)
- Les utilisateurs pourront
 - Voir les derniers posts de tous les utilisateurs
 - Voir les derniers posts d'un utilisateur spécifique
 - Publier des images
 - Publier du texte
- Sur chaque posts, les utilisateurs pourront
 - Liker / disliker
 - Commenter
 - Supprimer leur propre post/comment/like
- Modération par des administrateurs qui peuvent
 - Supprimer les posts/commentaires qu'ils jugent inappropriés.
 - Donner/enlever les droits d'admin à un autre utilisateur
- Autres
 - Session persistante au rechargement de la page
 - Système de notification

Pages

- Login
- Sign-in
- Agora = feed d'actualité
 - Liste des derniers posts
 - Like / Comment
- User = utilisateurs
 - Liste les utilisateurs
 - Barre de recherche d'utilisateur
- User/:id = profil d'un utilisateur
 - Modifier les infos
 - CRUD photo de profil
 - Supprimer le compte
- Notification = Page des notifications d'un utilisateur
 - répertorie les derniers commentaires et likes à l'un de ses posts

Organisation de travail

- AGILE
- Une première version basique (MVP) sera présentée au client
- Le développement se fera au fil de l'eau selon les requêtes du client
- L'objectif de ce projet est de créer ce premier MVP

Versionning

- Git et GitHub utilisés
- Workflow:

Technologies Utilisees :

BackEnd

- NODE : Version 16.13.1
- express : Version 4.17.2
- bcrypt : Version 5.0.1
- body-parser : Version 1.19.1

- cors : Version 2.8.5
- dotenv : Version 8.6.0
- helmet : Version 4.6.0
- jsonwebtoken : Version 8.5.1
- multer : Version 1.4.4
- mysql : Version 2.18.1
- mysql2 : Version 2.3.3
- crypto-js : Version 4.1.1
- joi : Version 17.5.0
- morgan : Version 1.10.0
- cookies : Version 0.8.0

FrontEnd

> Dependencies

- @angular/animations : Version 13.1.0
- @angular/common : Version 13.1.0
- @angular/compiler : Version 13.1.0
- @angular/core : Version 13.1.0
- @angular/forms : Version 13.1.0
- @angular/platform-browser : Version 13.1.0
- @angular/platform-browser-dynamic : Version 13.1.0
- @angular/router : Version 13.1.0
- ngx-image-cropper : Version 3.2.1
- ngx-infinite-scroll : Version 9.1.0
- rxjs : Version 7.4.0
- tslib : Version 2.3.0
- zone.js : Version 0.11.4

> DevDependencies

- @angular-devkit/build-angular : Version 13.1.2
- @angular-devkit/build-angular : Version 13.1.2
- @angular/cli : Version 13.1.1
- @angular/compiler-cli : Version 13.1.0
- @types/jasmine : Version 3.10.0
- @types/jasminewd : Version 22.0.8
- @types/node : Version 14.18.2
- codelyzer : Version 6.0.2
- jasmine-core : Version 3.10.0,
- jasmine-spec-reporter : Version 5.0.0
- karma : Version 6.3.9
- karma-chrome-launcher : Version 3.1.0
- karma-coverage-istanbul-reporter : Version 3.0.3
- karma-jasmine : Version 4.0.0
- karma-jasmine-html-reporter : Version 1.7.0
- protractor : Version 7.0.0
- ts-node : Version 10.4.0
- tslint : Version 6.1.0
- typescript : Version 4.5.3

Démarrage de l'application :

BackEnd

- Installez nodemon. Exécutez npm install, Toujours dans le répertoire « backend »

- Bien Évidement, après avoir créer votre BDD en suivant la doc existante dans le repertoire /Docs_&_Database/Database, Il faut créer un dossier caché .env dont voici mon exemple :

```
-----
DB_HOST='localhost'
DB_BASENAME='groupomania_walid'
DB_USER='walid'
DB_PASSWORD='baSE8533@)@+)'
CRYPT_USER_INFO='mot_de_passe_secret_pour_cryptoJS'
JWT_KEY='FC895214DR5214HTFEDS5G8V7VFRDXWA596447HYG'
COOKIE_KEY='encore_un_mdp_pour_le_cookie'
-----
```

- Exécutez "node server".
- Le serveur doit fonctionner sur http://localhost:3000.

FrontEnd

- Dans le dossier « frontend », exécutez npm install
- Puis, Tjrs dans le dossier frontendn exécutez la commande "ng serve" Cela devrait à la fois exécuter le serveur local et lancer votre navigateur directement sur le port 4200. -- http://localhost:4200.

NB : La création de votre premier Utilisateur le considerera comme Admin