

Documentation Technique -> Application de type Réseau Social

1- Fonctionnalités Implémentées

a. Connexion/Inscription Utilisateur

La fonctionnalité principale de l'application qui est nécessaire pour 90% des autres features et la Connexion/Inscription Utilisateur.

Cette fonctionnalité permet à l'utilisateur de se créer un compte ou de se connecter au travers d'un formulaire « POST » où doit être renseigné l'email et le mot de passe. Lors de la création de compte l'utilisateur reçoit un lien de validation dans sa boîte mail pour confirmer la création de son compte.

Cette fonctionnalité fait appel à Supabase, le formulaire récupère les informations de l'utilisateur et les retourne à Supabase en tant que demande d'inscription. A la suite de ça le mail est envoyé. 2 fonctions supabase sont utilisées.

Pour la connexion : **supabase.auth.signInWithPassword(data)**

Et pour la création de compte : **supabase.auth.signUp(data)**

Une session est donc ouverte grâce au Token et la session est stockée dans les cookies.

b. Déconnexion Utilisateur

Fonctionnalité permettant à l'utilisateur de se déconnecter de sa session grâce à la méthode de supabase : **supabase.auth.signOut()**

c. Mise à jour du profil utilisateur

L'utilisateur a un accès à son profil qu'il peut compléter et modifier seulement s'il est connecté. Il peut renseigner son Nom et Prénom, son nom d'utilisateur, son site web et il peut ajouter une photo de profil. Ses données sont envoyées à Supabase et sont renseignées dans la table « profiles »

d. Création et publication de messages en temps réel

Un utilisateur connecté peut rédiger un message et le publier. Ce message sera vu de tous les utilisateurs connectés ou non connectés.

Il peut rédiger un message grâce à un formulaire. Le message est récupéré et transmis à Supabase qui l'insère dans la table « posts »

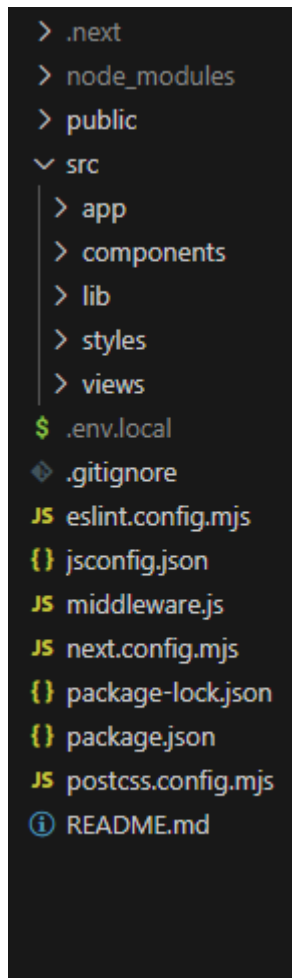
Les messages sont alors publiés en bas de la page d'accueil en temps réel grâce aux fonctionnalités WebSockets de Supabase notamment la fonctionnalité **supabase.channel** qui est lié directement à la table posts et qui permet

d'afficher la liste à jour des messages insérés dans la table puisqu'elle écoute en temps réel ce qu'il se passe au sein de cette table.

Un utilisateur connecté peut seulement modifier ou supprimer ses messages.

2- Architecture du Projet

Arborescence du projet :



Dans le dossier **src** nous allons retrouver le code principal de notre application. Notre code principal est divisé en 4 dossiers. Le dossier **app** où nous retrouvons les différentes pages de l'application. Le dossier **components** où se situent les composants en JSX que nous appelons dans nos pages. Le dossier **utils** est composé des fichiers qui permettent de paramétrer les fonctions utilitaires et de pouvoir justement lier notre backend avec notre front-end. Enfin, dans le dossier **views** nous retrouvons les différents fichiers qui permettent d'avoir une interface utilisateur (le front-end). C'est dans les fichiers contenus dans **app** et **components** que nous allons retrouver les différentes interactions entre Supabase et Next.JS.

3- Justification des règles de sécurité RLS mises en place.

a. Allow all users to read profiles

Cette RLS indique que tous les utilisateurs peuvent accéder aux données des profils utilisateurs (username, fullname, avatar_url, website). Cette règle est en place car certaines des données du profil sont affichées publiquement notamment pour savoir qui a rédigé quel message.

b. Allow profile updates for authenticated users

Règle de sécurité qui permet aux utilisateurs connectés SEULEMENT de modifier leur profil. Un profil utilisateur peut être modifier seulement si l'utilisateur est connecté. Cela permet de ne pas avoir de modification de la part de tiers et donc de protéger les données des utilisateurs

c. Users can insert their own profile.

Règle de sécurité qui permet aux utilisateurs connectés SEULEMENT de créer leur profil. Cela permet de contrer l'injection de BOT et de n'avoir que de vrais utilisateurs.

d. Users can update their own profile.

Règle de sécurité qui permet aux utilisateurs connectés SEULEMENT de ne modifier que leur profil. Cela permet de ne pas avoir de modification de la part d'autres personnes et donc de protéger les données des utilisateurs

e. Allow all users to read posts

Les messages peuvent être lus de tous les utilisateurs de l'application connectés ou non.

f. Insert posts only for authenticated users

Seuls les utilisateurs connectés peuvent rédiger des messages cela permet de tracer les différents messages notamment si des posts sont blessants.

g. User can delete only his posts

Les utilisateurs ne peuvent supprimer que leurs messages. Cela irait à l'encontre de la liberté d'expression de supprimer les messages des autres utilisateurs.

h. User can only edit his posts

Les utilisateurs ne peuvent modifier que leurs messages. Cela irait à l'encontre de la liberté d'expression de modifier les messages des autres utilisateurs, cela pourrait permettre la désinformation pour les autres utilisateurs, ou des répercussions pour l'utilisateur victime de cette modification.