



RAPPORT DE PROJET

Application de Gestion de Tâches

UE : Développement d'application client

Chargé de TP Jean-Luc Lamotte

Réalisé par le binôme :

- [22309694] – REZGUI Léna
- [22312242] – LAMAIRI Mohamed Yassine

01 Décembre 2025

1 Introduction et Vision du Projet

Dans le cadre de cette UE , nous avons développé une application de "Todo List" connectée à GraphQl. Au-delà des fonctionnalités de base imposées , nous avons implémenté des fonctionnalités supplémentaires et ce afin de permettre une expérience client plus fluide et agréable.

Ce rapport détaille l'ensemble des fonctionnalités que nous avons implémentées .

The image shows a mobile-style sign-in interface. At the top, the word "Sign In" is displayed in a dark font. Below it is a large, light-colored rectangular area containing two input fields. The first field is labeled "Username" and the second is labeled "Password". Both fields have a thin black border. Below these fields is a large, solid blue rectangular button with the words "Sign In" in white. At the very bottom of the screen, there is a thin horizontal bar with two links: "Sign In" on the left and "Sign Up" on the right, each preceded by a small downward-pointing arrow.

FIGURE 1 – Interface d'authentification et d'inscription

1.1 Authentification Sécurisée

Nous avons implémenté le cycle complet :

- **Inscription et Connexion** : L'utilisateur peut se connecter et créer un compte accédant par la suite à ses données personnelles sécurisées via l'API GraphQL.
- **Gestion des erreurs** : L'application informe visuellement l'utilisateur en cas d'erreur d'authentification ou manque de données de connexion(champ vide lors de la soumission).
- **Déconnexion** : Un bouton accessible permet de se déconnecter tout en affichant un message de confirmation avant.

2 Fonctionnalités supplémentaires implémentées

2.1 Feedback Visuel : La Barre de Progression

Plutôt que d'afficher un simple compteur textuel (ex : "2/5"), nous avons développé une **barre de progression graphique**.

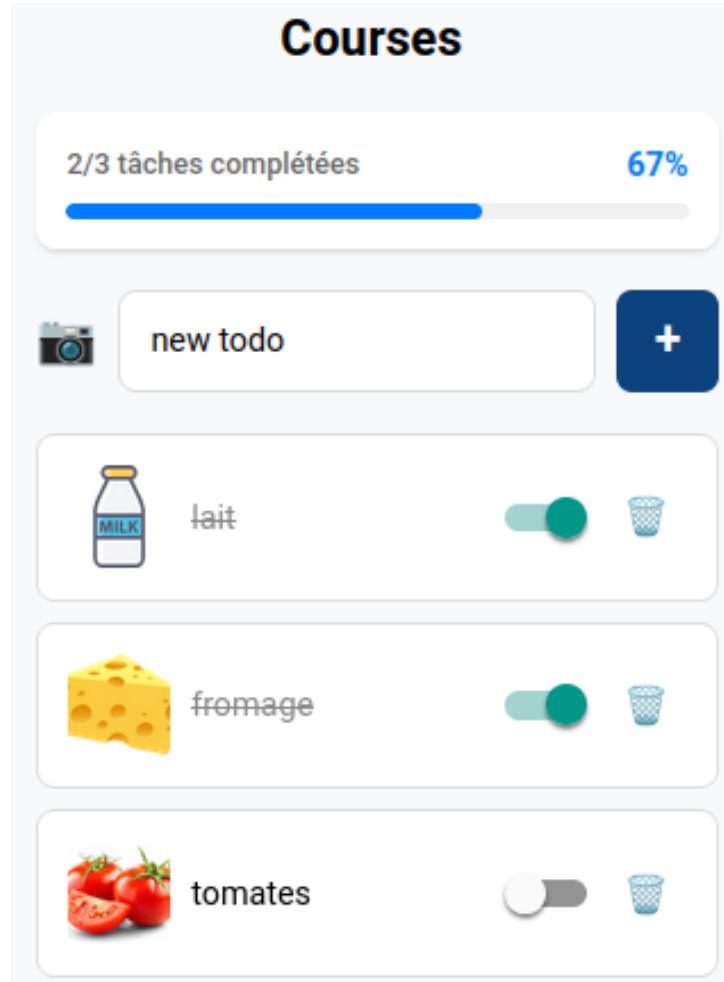


FIGURE 2 – Liste des tâches avec barre de progression et intégration d'images

Comme illustré sur la figure 2, la barre se remplit dynamiquement en fonction du ratio de tâches accomplies. Cela offre une gratification visuelle immédiate à l'utilisateur lorsqu'il coche une case.

2.2 Enrichissement Média

Nous avons étendu le modèle de données pour permettre l'ajout d'images aux tâches (visible ci-dessus). Cela permet, par exemple, d'illustrer une liste de courses ou de visualiser une tâche spécifique, rendant l'interface plus vivante.

3 Administration

3.1 Suppression Sécurisée de Compte



FIGURE 3 – Mécanisme de sécurité (Warning) avant suppression

Pour éviter les accidents (clics involontaires), nous avons mis en place une "Barrière de sécurité" :

- L'appui sur "Supprimer mon compte" ne lance pas l'action immédiatement.
- Une fenêtre de confirmation personnalisée (implémentée via le composant `Modal`) se superpose à l'écran. (voir figure 3).
- Ce n'est qu'après validation que la requête API est envoyée pour détruire les données.

4 Conclusion

Ce projet nous a permis de consolider nos connaissances en React Native (Props, State, Hooks) tout en explorant des concepts avancés comme la persistance via API. Nous sommes fiers de rendre une application non seulement fonctionnelle, mais aussi agréable à utiliser et complète.