

Dans ce document vous devez présenter votre projet. Cette présentation, outre la description du projet, doit aborder plusieurs autres thématiques comme l'explication du lien entre vos futurs réalisations et la formation, vos motivations, des exemples de cas d'utilisations. Il est aussi important de détailler les points complexes de votre projet qui vont mettre en avant votre virtuosité. Ce projet doit montrer les compétences que vous avez acquises au cours de votre formation et/ou de votre alternance.

Ce projet consiste à développer une application mobile destinée à la gestion d'une bibliothèque personnelle. Cette application inclut des fonctionnalités telles que la recommandation de livres personnalisée, l'intégration de l'API Google Books pour la récupération des données, et une dimension sociale permettant de mettre en relation les lecteurs partageant des goûts similaires. L'objectif principal est de simplifier l'organisation des livres tout en offrant une expérience enrichissante grâce à des recommandations intelligentes et une fonctionnalité interactive. De plus, l'application vise à créer une communauté de lecteurs en permettant des interactions sociales basées sur les préférences littéraires.

Le projet s'inscrit dans un contexte de montée en compétences sur les technologies mobiles et les architectures cloud, éléments clés du parcours Master CCM. Il met en lumière les compétences en développement front-end et back-end acquises au cours de la formation, avec un accent particulier sur l'utilisation de services de synchronisation des données comme Firebase ou CloudKit, ainsi que sur la gestion des API, notamment l'intégration de Google Books. La mise en œuvre de ces fonctionnalités reflète une maîtrise des concepts de mobilité et des architectures distribuées, tout en exploitant le potentiel de SwiftUI pour le développement mobile.

L'initiative de ce projet est motivée par le désir de créer un outil qui facilite la gestion des bibliothèques personnelles et qui propose aux utilisateurs des suggestions de lecture adaptées à leurs goûts. L'intégration de fonctionnalités ludiques, comme la détection du secouement pour générer des recommandations aléatoires, ou encore un système de mise en relation entre lecteurs aux goûts similaires (façon "Tinder" pour lecteurs), renforce l'aspect interactif de l'application. Cela répond à la fois à un besoin personnel d'exploration de nouvelles lectures et à une volonté de créer une dimension communautaire autour de la littérature.

L'application permet par exemple à un utilisateur d'obtenir, à chaque ouverture, des recommandations de livres basées sur son historique de lecture. Elle offre également la possibilité de secouer son téléphone pour générer une suggestion aléatoire, ce qui ajoute une touche d'interaction physique amusante. En outre, deux lecteurs qui partagent plus de 80 % de compatibilité en termes de genres littéraires pourront échanger sur leurs lectures via une messagerie intégrée, enrichissant ainsi l'expérience sociale et littéraire.

Les aspects complexes du projet résident notamment dans l'intégration dynamique de l'API Google Books, afin de récupérer des informations sur les livres en temps réel. L'algorithme de recommandation, basé sur les habitudes de lecture des utilisateurs, représente également un défi technique majeur. De plus, la gestion des fonctionnalités liées au réseau, ainsi que l'authentification des utilisateurs via Firebase, sont des éléments cruciaux qui demandent une attention particulière pour garantir une expérience fluide et sécurisée.

Ces défis techniques sont fondamentaux pour assurer la robustesse et l'efficacité de l'application. L'intégration de l'API Google Books permet de rendre le contenu dynamique et à jour en fonction des derniers livres disponibles sur la plateforme, tout en assurant une gestion fluide de l'information. Cet aspect est essentiel pour offrir aux utilisateurs une expérience riche et pertinente. De plus, l'algorithme de recommandation doit non seulement analyser les préférences de lecture actuelles, mais aussi proposer des suggestions pertinentes, tout en tenant compte de l'évolution des goûts de l'utilisateur. L'adaptation et la personnalisation sont donc au cœur du projet pour garantir que les livres proposés correspondent aux attentes des utilisateurs.

Un autre point clé est l'aspect social de l'application, avec un système de correspondance entre les lecteurs (Ok, j'admets que c'est peut-être un peu trop, mais on va essayer...). Cet aspect implique une gestion des profils utilisateurs sophistiquée, basée sur les préférences de genres littéraires et les livres lus. La mise en place de la messagerie doit non seulement être fluide, mais aussi sécurisée, assurant ainsi un espace de communication agréable et privé pour les utilisateurs. Cette dimension sociale a pour but de transformer l'application en un véritable réseau de lecteurs passionnés, permettant aux utilisateurs de partager leurs découvertes littéraires et de discuter de leurs livres favoris avec des personnes partageant les mêmes goûts.

Enfin, le développement de l'application nécessite une architecture robuste, capable de gérer les données de manière efficace, que ce soit localement via CoreData ou en ligne via Firebase/CloudKit. La synchronisation des données entre différents appareils est un enjeu central, car les utilisateurs doivent pouvoir accéder à leurs bibliothèques personnelles quel que soit l'appareil utilisé. Le projet met donc en œuvre une gestion des données en temps réel, tout en respectant les meilleures pratiques en matière de sécurité et de gestion des identités.

En somme, ce projet s'inscrit parfaitement dans les objectifs de la formation en Cloud Computing et Mobilité. Il représente une occasion unique de démontrer les compétences acquises, notamment dans l'intégration d'API, la gestion des données dans des architectures cloud, le développement mobile via SwiftUI, ainsi que la mise en place d'interfaces utilisateur intuitives et performantes. Au-delà de l'aspect technique, ce projet met également en avant des compétences en gestion de projet et en collaboration, avec une gestion efficace des tâches, une répartition des rôles au sein de l'équipe et une planification rigoureuse pour garantir la livraison d'une application de qualité.