

COTE-COLISSON Romane

HERRERO Valentin

TRAVERSIER Mathys



Développement mobile 2023

Exploitation d'une API Web



Année 2022/2023

Projet encadré par Baptiste PESQUET

TABLE DES MATIÈRES

Introduction	3
Installation du projet	3
Description de l'application	3
Liste des fonctionnalités développées	3
Langage et environnement	4
Sélection des informations pour l'application	4
Stockage des informations	4
Architecture de la solution	5
Architecture du dossier	5
Architecture de l'application	6
Organisation du travail en équipe	8
Conclusion et perspectives	9

Introduction

Au cours de notre formation à l'Ecole Nationale Supérieure de Cognitique, nous avons eu l'opportunité de mettre en pratique nos compétences en développement mobile en participant à un projet visant à créer une application mobile utilisant une API web externe. Notre choix s'est porté sur une API portant sur la répartition des familles au sein de l'ENSC, projet réalisé précédemment par Mathys et Valentin au semestre 7. L'API *familyTriz* est disponible à l'adresse suivante : <https://familytriz.azurewebsites.net/>.

Le but de notre application est de fournir aux utilisateurs une interface claire et conviviale permettant de consulter les données relatives aux familles de l'école, y compris les liens de parenté entre les étudiants, les scores des familles en temps réel et les informations sur les étudiants eux-mêmes. Pour ce faire, nous avons créé trois onglets distincts : le premier affiche les cinq familles de l'ENSC, le deuxième permet de consulter les scores de chaque famille, et le dernier onglet est dédié aux étudiants de l'école.

Notre projet nous a permis de mettre en pratique les connaissances et compétences acquises tout au long de notre module de formation, et de nous familiariser avec les différentes étapes du développement d'une application mobile basée sur une API web externe.

Installation du projet

Voici les étapes pour notre projet fonctionne correctement :

- Cloner le projet grâce à cette adresse depuis gitHub :
 - https://github.com/ensc-mobi/projet-2023-cote-colisson_gr3

Description de l'application

Liste des fonctionnalités développées

Code	Description
Famille	
F_01	En tant qu'utilisateur, j'ai accès à la liste de chaque membre de chaque famille
F_02	En tant qu'utilisateur, j'ai accès au nombre de points de chaque famille
Points	
F_03	En tant qu'utilisateur, j'ai accès au diagramme des points entre chaque famille
Élèves	

F_05	En tant qu'utilisateur, j'ai accès à la fiche d'un étudiant
F_06	En tant qu'utilisateur, je peux ajouter un étudiant
F_08	En tant qu'utilisateur, je peux supprimer un étudiant
Recherche	
F_09	En tant qu'utilisateur, je peux effectuer une recherche d'élèves par nom, prénom.

Langage et environnement

Pour développer notre application, nous avons utilisé le framework React Native. De plus, nous avons aussi utilisé Prettier qui est un formateur de code, c'est-à-dire que lorsque certaines règles ne sont pas respectées, il va formater le code pour qu'il respecte ces règles. Nous avons choisi de coder sur l'éditeur de code Visual studio Code. Notre code est disponible à l'adresse suivante sur GitHub :

https://github.com/ensc-mobi/projet-2023-cote-colisson_gr3

Sélection des informations pour l'application

Pour notre projet d'application mobile, nous avons choisi de ne présenter que quelques étudiants de l'école pour des raisons pratiques et de simplicité. En effet, l'ENSC compte environ 250 étudiants, ce qui aurait rendu la présentation de toutes les familles et de leurs liens de parenté très complexe et difficile à naviguer pour l'utilisateur final.

De plus, nous avons également considéré que pour une première version de l'application, il était préférable de se concentrer sur quelques étudiants sélectionnés de manière aléatoire pour permettre une démonstration efficace des fonctionnalités principales de l'application.

Enfin, il était également important de prendre en compte les limites de temps et de ressources pour la réalisation du projet. Ainsi, en limitant le nombre d'étudiants à présenter dans l'application, nous avons pu nous concentrer sur la qualité de l'interface utilisateur, la facilité de navigation et la fluidité des interactions avec l'API web externe.

Cependant, il est important de noter que l'application pourrait facilement être étendue pour inclure tous les étudiants et familles de l'ENSC, une fois que la version initiale aurait été testée et validée.

Stockage des informations

Les principales informations sur lesquelles nous avons travaillé sont donc

- ★ Les différentes familles

- ★ Les différents étudiants
- ★ Le nombre de points obtenus par chaque famille

Par exemple, la liste des étudiants inclus dans une famille est récupérée sous forme de tableau de chaînes de caractères.

Architecture de la solution

Architecture du dossier

Pour regrouper les informations, nous avons réparti les différents fichiers .tsx dans les différents dossiers :

- le dossier **assets** comporte les différentes images utilisées (les images de chaque famille, les icônes...)
- le dossier **api** se compose des différents services permettant d'accéder aux données venant des API et de les traiter.
- le dossier **components** rassemble différents types de composants :
 - les composants *List* permettent de présenter des listes d'éléments
 - les composants *Item* définissent le rendu de chaque élément de liste
- le dossier **navigation** comporte deux fichiers :
 - *app-stacks* permet d'utiliser la navigation sous forme de piles de screens
 - *tab-navigation* définit le rendu de la barre de menu
- le dossier **screens** rassemble les différentes pages de l'application
- le dossier **thème** comporte :
 - *colors.js* : fichier qui permet de modifier la couleur du header
 - *styles.js* : fichier qui permet de définir le style global de l'application

Architecture de l'application

Voici l'architecture de l'application de manière plus schématisée. Ainsi, nous retrouvons en violet les différentes pages (screen) de l'application, qui incluent des composants principaux de type liste, eux-même composés d'éléments plus individuels (souvent de type Item).

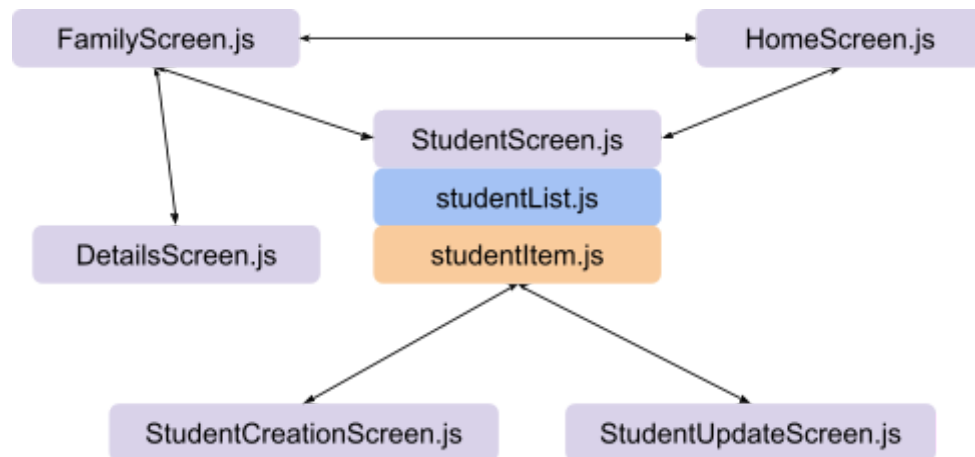


Schéma de l'architecture de l'application

Navigation :

- ★ Une page d'accueil avec le diagramme des points : *HomeScreen.js*
- ★ Une page permettant d'accéder aux détails des 5 familles de l'ENSC : *FamilyScreen.js*
- ★ Une page permettant d'avoir accès aux détails d'une famille en particulier : *DetailsScreen.js*
- ★ Une page donnant accès à tous les étudiants de l'ENSC : *StudentScreen.js*
- ★ Une page d'ajout d'un étudiant : *StudentCreationScreen.js*
- ★ Une page de modification d'un étudiant : *StudentUpdateScreen.js*

A l'ouverture de l'application, l'utilisateur arrive sur la page d'accueil, *HomeScreen*. À partir de celle-ci, il a la possibilité de naviguer sur l'onglet "Familles" et l'onglet "Élèves" via les onglets.

Depuis l'onglet "Familles", on accède directement à la page *FamilyScreen* où l'on peut accéder aux détails de la famille souhaitée en cliquant sur le bouton correspondant. De là, l'utilisateur accède à la page *DetailsScreen* de la famille sélectionnée.

Enfin, depuis l'onglet "Élèves", l'utilisateur accède à la page *StudentScreen* où il voit la liste des élèves de l'ENSC. Il peut alors interagir avec cette liste en cliquant sur différents icônes positionnés dans chaque Item représentant un élève. L'icône "poubelle" sert à supprimer l'élève et l'icône "notes" sert à le modifier. En le modifiant, l'utilisateur accède alors à la

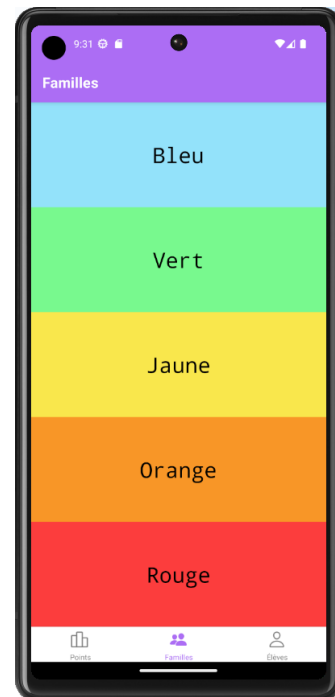
page *StudentUpdateScreen*. Il peut aussi décider d'ajouter un élève en cliquant sur le bouton “+” en haut à droite de la page. Il accède alors à la page *StudentCreationScreen*.

Interfaces

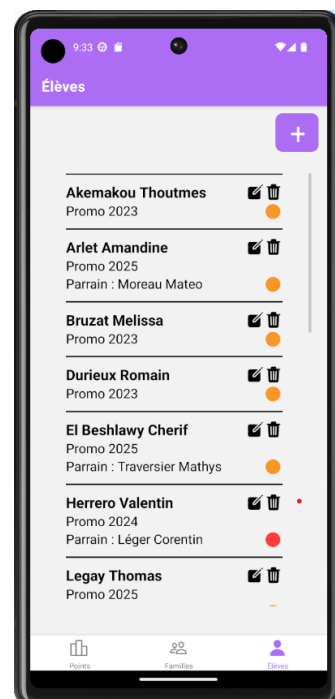
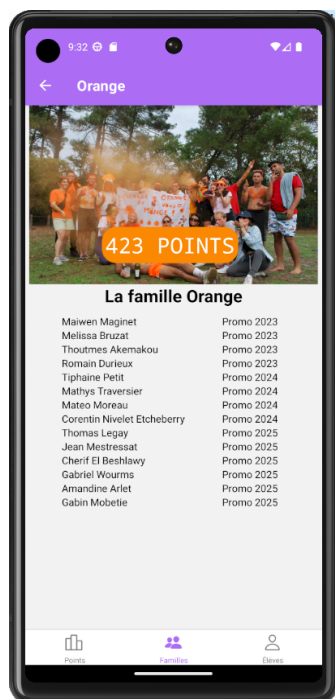
Voici les interfaces de l'application :



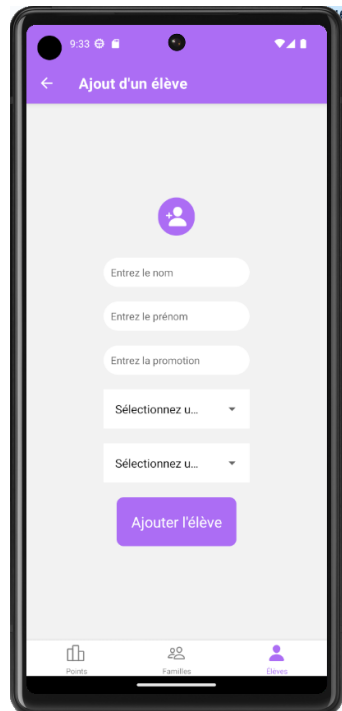
Page d'accueil de l'application



Page “familles” de l'application

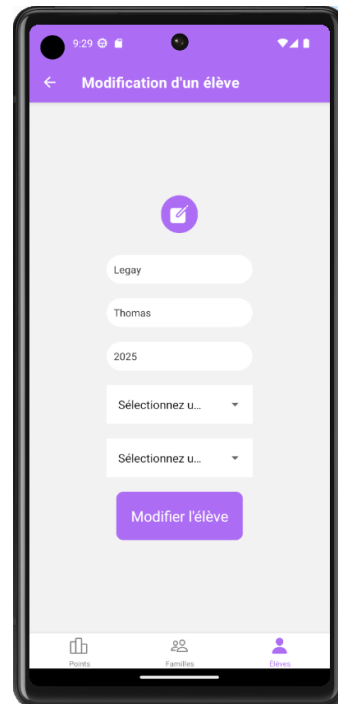


Page "Détails" sur la famille orange



Page "Ajout d'un élève"

Page "Elèves"



Page "Modification d'un élève"

Organisation du travail en équipe

Notre travail en équipe a débuté par une phase de travail collaboratif en présentiel, au cours de laquelle Mathys et Valentin ont commencé à esquisser les maquettes de l'application. C'est à ce moment-là que Romane nous a rejoint, après que sa camarade de projet soit partie en semestre à l'étranger. Les maquettes ont permis de visualiser et de structurer le travail à réaliser.

Nous avons ensuite réparti les tâches entre nous pour travailler efficacement. Pour favoriser les échanges et la coordination de nos efforts, nous avons communiqué régulièrement via des applications telles que Messenger, étant donné que Valentin était souvent occupé avec ses entraînements de handball. Cette approche nous a permis de travailler de manière synchrone et de faire avancer le projet efficacement.

Enfin, nous avons également utilisé le pair programming via GitHub pour résoudre les problèmes rencontrés au cours du développement de l'application. Cette technique de travail en trinôme nous a permis de travailler plus efficacement ensemble en résolvant rapidement les problèmes et en partageant nos connaissances techniques. Nous faisons tout de même attention à ne pas travailler sur les mêmes fichiers en même temps, afin d'éviter les conflits.

Conclusion et perspectives

Grâce à ce projet, nous avons eu l'opportunité de renforcer nos connaissances sur React Native, en appliquant les concepts abordés en cours et en essayant de nous affranchir des tutoriels. Nous sommes fiers d'avoir développé une application fonctionnelle dotée d'une interface intuitive et facile à utiliser. Ce projet a donc été bénéfique pour notre apprentissage de la programmation mobile avec React Native.

Ce projet nous a cependant posé certaines difficultés, notamment au niveau des erreurs dans le code. Nous avons notamment rencontré un problème avec l'API de la liste des élèves. Après avoir tenté de résoudre le problème sans succès, nous avons dû reprendre le code de notre projet de Génie Logiciel pour trouver la source de l'erreur et y remédier. Finalement, en corrigeant l'API, nous avons pu rétablir la fonctionnalité de la liste des élèves.

En affrontant les défis et les obstacles qui ont surgi lors de ce projet, nous avons pu acquérir de précieuses compétences en matière de résolution de problèmes. Ces compétences nous seront bénéfiques à l'avenir, car elles nous aideront à mieux surmonter les contraintes et les difficultés dans les projets à venir. En d'autres termes, grâce à cette expérience, nous sommes mieux équipés pour relever les défis futurs avec plus de confiance et de compétence.

Ainsi, des perspectives d'améliorations peuvent être envisagées afin d'enrichir davantage l'application. Il aurait été intéressant de créer un système de connexion, et d'ajouter une page de gestion des différents événements. Afin de compléter l'application, il aurait été bien d'avoir la possibilité de modifier les informations des élèves.