

Plan d'accompagnement mobile et back



Semaine 1 : Bases & Setup du Projet



Objectif : Être à l'aise avec les outils, l'environnement et la structure du projet



Frontend (React Native)

- Installer l'environnement (Expo ou React Native CLI)
- Apprendre les bases :
 - Composants (`View`, `Text`, `Button`, `TextInput`)
 - Navigation (`react-navigation`)
 - Hooks (`useState`, `useEffect`)
- Créer des écrans simples : Accueil, Connexion, Inscription

Notion à approfondir: routing, usage des hooks, création de hook personnalisé



Backend (Laravel)

- Installer Laravel
- Routes, contrôleurs, modèles, migrations
- Apprendre l'authentification avec Sanctum
- Créer les endpoints : `POST /register`, `POST /login`, `GET /user`

Notion à approfondir: processus de réinitialisation de mot de passe, model avec plusieurs relations



API & Connexion Front/Back

- Installer Axios dans React Native
- Tester les appels API depuis le mobile vers Laravel
- Gérer les erreurs et loading

Notion à approfondir: gestion des erreurs, loading



Semaine 2 : Authentification et Sécurité



Objectif : Mettre en place un système d'authentification sécurisé



Backend

- Finaliser le système d'authentification (Sanctum + tokens)
- Ajouter les middlewares `auth:sanctum`
- Créer endpoint `logout`, `refresh`

Notion à approfondir: gestion de droits et permission, autre système d'authentification



Frontend

- Stocker le token (AsyncStorage)
- Gestion de session (auto-login, logout)
- Redirection conditionnelle (si non connecté → login)
- Affichage des erreurs d'authentification

Notion à approfondir: session, localStorage, mise en place et usage de context + routing



Semaine 3 : Gestion de l'épargne



Objectif : CRUD sur une entité métier (épargnes)



Backend

- Migration/Model : **Savings** (montant, objectif, user_id, nom, etc.)
- Routes API : **GET /savings**, **POST /savings**, **PUT**, **DELETE**
- Validation avec **FormRequest**
- Liens entre utilisateurs et leurs épargnes (**hasMany**)



Frontend

- Écran liste des épargnes
- Formulaire de création/modification
- Suppression avec confirmation
- Rafraîchissement dynamique

Notion à approfondir: formRequest personnalisé, approfondissement des types de validation, traduction des erreurs de request, usage de uuid, rafraîchissement dynamique et conditionnelle



Semaine 4 : Gestion de budget + Finalisation



Objectif : Ajouter la logique de budget, corriger, tester et documenter



Backend

- Entité **Budgets** :
 - Catégorie (alimentation, transport...)
 - Montant alloué, montant dépensé
 - Dates (mois en cours, etc.)
- Routes : **GET /budgets**, **POST** , **PUT**, **DELETE**



Frontend

- Vue mensuelle du budget
- Ajout/modif des budgets
- Affichage des dépenses vs budget (barres, couleurs)
- Intégration avec la logique d'épargne (ex : % épargné automatiquement)



Finalisation

- Ajouter **Loader**, **Toasts**, validation UI
- Teste les erreurs réseau, expirations de session, mauvais identifiants
- Documente l'API avec Swagger ou Postman
- Nettoyage du code

Notion à approfondir: codes propres, lecture de documentation....