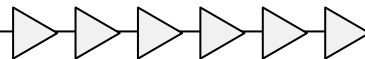




Exploration de Swift

TPALT, 2024

Alina Novikova



Introduction

Objectif du Projet:

- Explorer un domaine jusqu'alors non abordé par mes expériences précédentes - le langage Swift
- Développer une application qui répond à un besoin spécifique, en utilisant une architecture moderne et des technologies récentes

Pourquoi Swift ?

- Performance Optimale
- Sécurité Avancée
- Écosystème Riche

Le défi

Apprendre un tout nouveau langage tout en construisant une application fonctionnelle.

La solution

Application **SophistiSpend**, une démonstration d'architecture moderne et de programmation Swift.





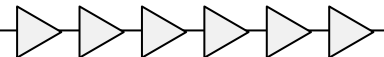
Choix du Sujet du Projet

- **Projet choisi** : Application de Gestion de Budget Simplifiée

Contraintes de temps :

7 semaines,
6 heures par semaine max
~ 48H

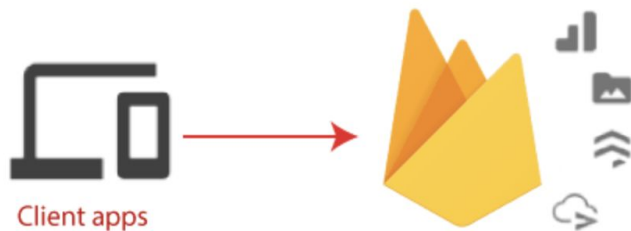
- Nécessité de gérer le temps avec d'autres matières
- **Travail individuel** : Projet réalisé seul, sans binôme
- **Simplicité requise** : Complexité adaptée à la situation
- **Utilité personnelle** : Usage personnel d'un système de gestion de budget



Architecture de Haut Niveau

Options habituelles :

Firebase



ou

SwiftData



Pourquoi non?

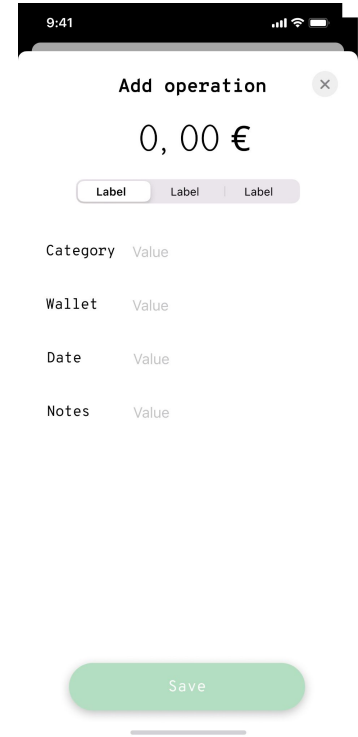
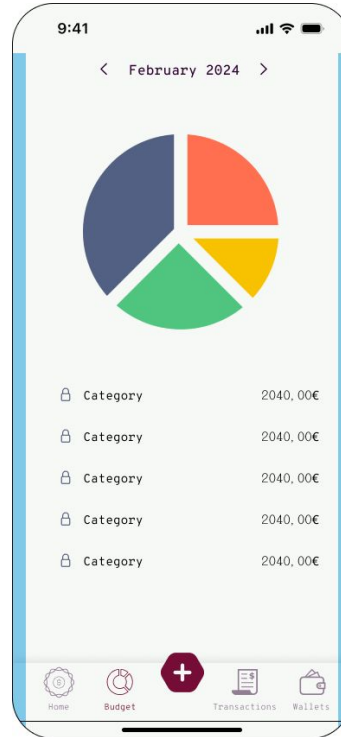
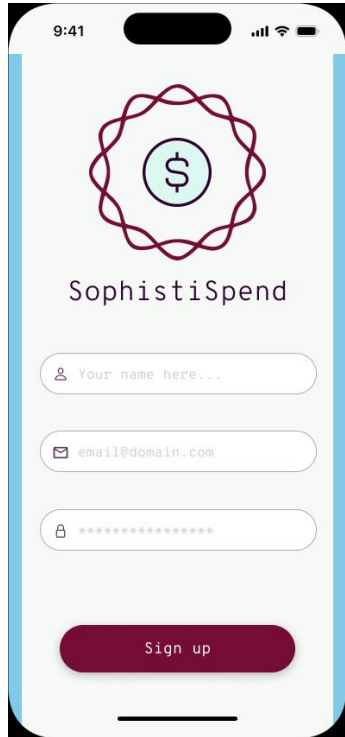
Expérience préalable avec Firebase et inadéquation de NoSQL pour ce type de données

Pourquoi non?

Nouveauté du sujet nécessitant un temps d'apprentissage supplémentaire

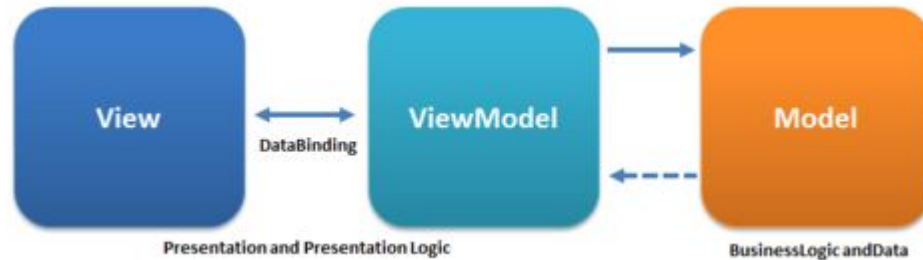
Solution envisagée : Architecture combinant Swift, backend Go, et PostgreSQL

Wireframes



Architecture Détaillée - Frontend Swift

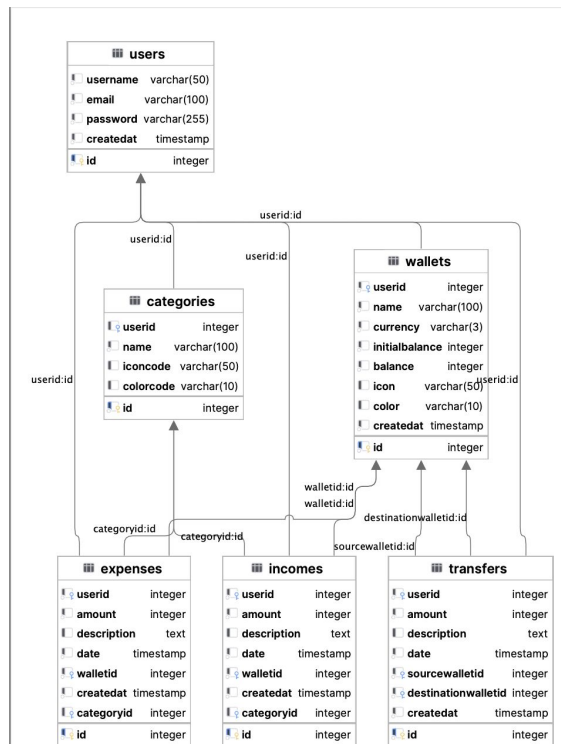
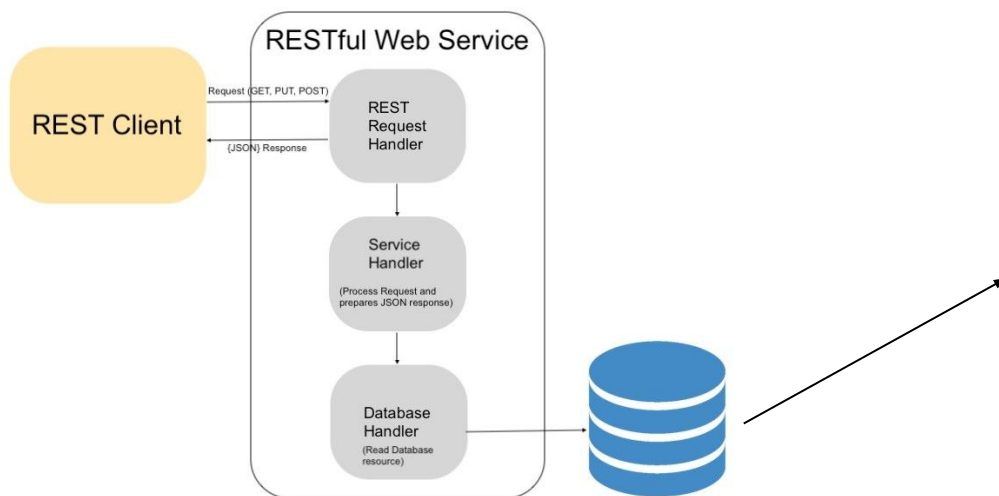
- Frameworks et Patterns : SwiftUI et MVVM
- Fonctionnalités **SwiftUI** : Interface utilisateur déclarative et réactive
- Architecture **MVVM** : Séparation claire entre la logique d'affichage et la logique métier
- Interaction avec le Backend : Communication via **REST API** pour échanger des données en temps réel

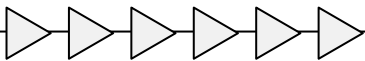


MVVM

Architecture Détaillée - Backend Go

- Choix de Go : Performance, concurrence, simplicité
- Structure : Serveur REST API pour la communication avec le frontend
- Sécurité : Authentification JWT, validation des requêtes
- Intégration avec PostgreSQL : Gestion efficace des données relationnelle



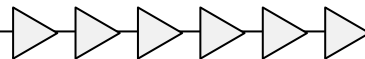


Leçons Apprises et Retours

- Appréciation de Swift : Éléance et éléments prêts à l'emploi
- Points Forts : Intuitivité et richesse de l'écosystème Apple
- Défis : Courbe d'apprentissage et intégration avec des technologies non-Swift

**Conclusion : Une expérience enrichissante et
motivante**

- Développement Continu : Ajout de nouvelles fonctionnalités et optimisations
- Perspectives : Potentiel de publication sur l'App Store
- Réflexion : Une fondation pour des projets futurs plus ambitieux





Merci!