

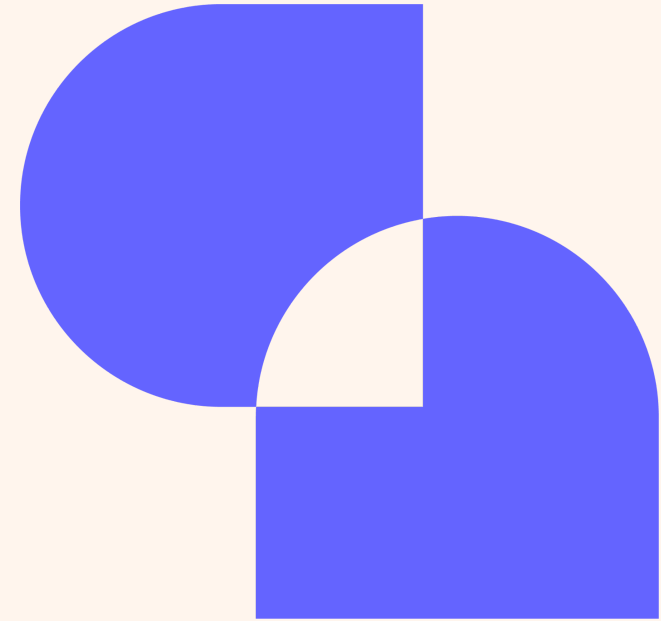
GoMind

“Agir avec l'esprit éveillé”

DOSSIER DE COMPÉTENCES

Développeur Sénior SwiftUI

9 ans d'expérience



PRINCIPALES EXPÉRIENCES

CIVITERRA : Développeur SwiftUI (2024 à aujourd'hui)
SPORTEASY : Lead Developer SwiftUI - UIKit (2021-2023)
AGILIDEE : Lead Developer SwiftUI- UIKit (2019-2021)
NUWED : Developer Swift-UIKit (2018-2019)
PUMPKIN: Developer Swift UIKit (2017-2018)
ALTRAN : Developer Swift-Objective C (2016-2017)

Autre expérience :

MENTOR OPENCLASSROOMS (2021-2024)

CONNAISSANCES TECHNIQUES

Langages : Swift, Javascript, Python , GO, SQL,
Typescript,Objective-C

Librairies/framework :

SwiftUI, UIKit, Combine, RxSwift, Swift TouchId, FacelId,
Data, Swift Package Manager, Cocoapods, Async/Await,
Foundation, Map Integration (MapKit, Location
Manager, Core Location), Gestion Multimedia
(AVFoundation, AVKit), Animation et Transition (Core
Animation, Core Graphics), Phrase/Localisation,
HealthKit

Architecture/Design Pattern : MVVM-C, VIPER, Clean Archi,
MVP, MVC, Observer (Delegate, KVO), Singleton, DI, DTO,
Coordinator, Adapter, SOLID, Clean architecture, TCA
architecture,

Base de données : CoreData, MySQL, Firebase,
PostgreSQL, pgAdmin 4, Realm

Bonnes pratiques : Code Review, Pair Programming,
Xctest, Test, TDD, KISS, DRY, SwiftLint,

Outils : Github, Gitlab CI/CD, Bitrise, Jenkins, Xcode, Swift
Package Manager, Cocoapods, Google Crashlytics, Firebase,
Instruments, Fastlane, IntelliJ, Terminal, Serveur Linux,
Caddy, Docker, Applications auto-hébergées, Portainer,
Grafana, Prometheus,

Gestion de projet : Agile, Scrum, Jira, Notion, Trello, Figma,
KANBAN

Veille : Point-Free,[Swift.org](https://swift.org) , Swift by Sundell, Vincent
Pradeilles, SwiftLee, WWDC, Lets Build That App, boot.dev,
Micode, Underscore, ThePrimeagen, Melkey, Stack Overflow,
ChatGPT, Claude, Mistral

EXPÉRIENCES TECHNIQUES

CIVITERRA : Développeur SwiftUI (2024 à aujourd'hui)

Projet de création d'application (Saas)

- Analyse fonctionnel
 - Marketing : Analyse de marché, interactions avec les différents acteurs
- Suivi des spécifications
- Rédaction d'un cahier des charges
 - Conception du cahier des charges
 - Interaction avec le design pour concevoir et modifier les interfaces
 - Choix des technologies
- Équipe de 2 personnes :
 - Documentation du code
 - Analyse des wireframe avec Figma
 - Ticketing avec Trello
- Mise en place de l'architecture MVVM :
 - Mise en place de la clean architecture : SOLID
 - Utilisation de la clean architecture : création de couche présentation, domaine, et data
 - Utilisation de Combine : mise en place des observable object, publisher, operator, cancellable
 - Mise en place de micro package
- Développement en SwiftUI :
 - Mise en place Swift Package Manager
 - Mise en place du Binding, Observer

- Gestion des appels réseaux en Async/Await
- Développement en GO :
 - Utilisation de Go 1.24 pour l'API
 - Base de données PostgreSQL
 - Gestion et mise à jour du serveur
- Versioning sous Git
 - Suivi du Gitflow
- Hébergement de l'application sur Gitlab
- CI/CD Gitlab : Déploiement sur un VPS Scaleway
- Bonnes pratiques :
 - Code review
 - Ecriture de la documentation
 - Refacto de code

Stack technique :

GO, SwiftUI, SOLID, SwiftLint, MVVM, EPOCH, Swiftcharts, SiriKit, API Binance, XCode13, Pair Programming, Review, SOLID, CI/CD Gitlab, Swift Package Manager, Gitlab, Go 1.24, PostgreSQL, TailwindCSS, NuxtJS (VueJS)

SPORTEASY : Lead Developer SwiftUI - UIKit (2021-2023)

Refactoring des fonctionnalités de l'application iOS SportEasy

- Analyse fonctionnel
- Echange avec les équipes Android et Web
- Suivi des spécifications
- Documentation de code
- Équipe de 5 personnes :

- Gestion de projet en Scrum
 - Sprint de 2 à 3 semaines
- Mise en place de l'architecture MVVM :
 - Mise en place de DRY, Kiss,
- Développement Swift 4 :
 - Utilisation de AVFoundation
 - Gestion du Lazy
 - Utilisation de Guards
- Développement SwiftUI :
 - Gestion de States Object
 - Gestion des Observed object
 - Gestion de la navigation
 - Gestion du Binding
 - Utilisation d'URLSession
 - Utilisation de Firebase
 - Gestion des permissions Caméra
 - Gestion offline avec CoreData
 - Ajout d'une pagination et pull to refresh sur la liste des installations :
 - Gestion des types, affichage et mise à jour d'une installation
 - Correction de bug :
 - Navigation style sur iPad
 - Gestion du tracking avec Analytics
 - Rapport sur Prometheus pour remonter les bug
- Développement UIKit :
 - Mise en place des UI table View
 - Utilisation de Core Animation/Lottie
 - Utilisation du NavigationController

- Gestion de la base de données :
 - Gestion de Core Data
 - Test d'API avec Postman
- Versioning avec Git
 - Suivi du Gitflow
- Écriture des tests unitaires de XCTest :
 - Tester les ViewModels, tests des services (Network)
- Pratique du TDD
- Utilisation de SwiftLint
- Déploiement continu avec Firebase app distribution
- Déploiement avec Fastlane
- Bonne pratiques :
 - Code review
 - Écriture de la documentation
 - Refacto de code

Stack technique :

Swift4, MVVM, SwiftUI, Figma, UIKit, Activity Indicator View, Photo Library, XCode13, Git, Agile, Pairprogramming, SOLID, Swift Package Manager et Cocoapods

AGILIDEE : Lead Developer UIKit (2019-2021)

Développement de A à Z de l'application myProman

- Analyse fonctionnel
- Suivi des spécifications
- Analyse des wireframe avec Figma

- Équipe de 4 personnes :
 - Rôle de Lead
 - Définition de l'API Design
 - Échange sur l'architecture
 - Choix technologique
- Gestion de projet en SCRUM
 - Sprint de deux semaines
- Mise en place de l'architecture MVVM :
 - Mise en place de DRY, Kiss
 - Utilisation du TDD
 - Mise en place de SOLID
- Développement Swift 4
- Développement Ulkit :
 - Mise en place des UI table View
 - Mise en place de RxSwift
 - Utilisation d'Alamofire
- Gestion de la base de données :
 - Gestion de Core Data
 - Test d'API avec Postman
- Versioning avec Git
 - Suivi du Gitflow
- Écriture des tests unitaires de XCtests :
 - Tester les ViewModels, tests des services (Network)
- Pratique du TDD
- Utilisation de SwiftLint
- Utilisation de Bitrise pour la CI/CD
- Bonne pratiques :
 - Code review
 - Écriture de la documentation

- Refacto de code

Stack technique :

MVVM, Ulkit, TDD avec XCTest, Review, SOLID, CI/CD
Gitlab,RxSwift,

[NUWED](#) : Developer Swift-Ulkit (2018-2019)

Création from scratch d'un réseau social type SnapChat

- Analyse fonctionnel
- Suivi des spécifications
- Documentation de code
- Équipe de 2 personnes
- Mise en place de l'architecture VIPER
- Développement Ulkit :
 - Développement d'écrans
 - Gestion du profil
 - Intégration de la Caméra avec AVFoundations
 - Utilisation d'Alamofire
 - Utilisation de R.Swift
 - Utilisation de RxSwift
- Versioning avec Git
 - Suivi du Gitflow
- Utilisation de GitlabCI
- Ecriture de tests unitaires avec XCTest
 - Tester les ViewModels, tests des services (Network)
- Bonne pratiques :

- Écriture de la documentation
- Refacto de code

Stack technique :

Swift4, MVVM, SwiftUI, Figma, UIKit, Activity Indicator View, Photo Library, XCode13, Git, Agile, Pairprogramming, SOLID,Swift Package Manager et Cocoapods

PUMPKIN: Developer Swift-UIKit (2017-2018)

Développement de l'application Pumpkin Business

- Analyse fonctionnel
- Suivi des spécifications
- Rédaction d'un cahier des charges
- Analyse des wireframe avec Figma
- Documentation du code
- Équipe de 8 personnes
 - Méthodologie Scrum & Kanban
 - Sprint de deux semaine
- Mise en place de l'architecture VIPER :
- Développement UIKit :
 - Développement d'écrans
 - Gestion du profil
 - Intégration de la Caméra avec AVFoundations
 - Utilisation d'Alamofire
 - Utilisation de R.Swift
 - Utilisation de RxSwift
 - Développement d'un scan de QR code natif

- Versioning avec Git
 - Suivi du Gitflow
- Écriture des tests unitaires de XCTest :
 - Tests et validation avec ViewInspector pour view SwiftUI
 - Relecture de code
 - Respect des standard
- Bonne pratiques :
 - Écriture de la documentation
 - Refacto de code

Stack technique :

Swift4, MVVM, SwiftUI, Figma, UIKit, Activity Indicator View, Photo Library, XCode13, Git, Agile, Pairprogramming, SOLID,Swift Package Manager et Cocoapods

FORMATION

Supinfo International University (2017)

LANGUES

Français courant

Anglais C1

Portugais brésilien A2