

# GoMind

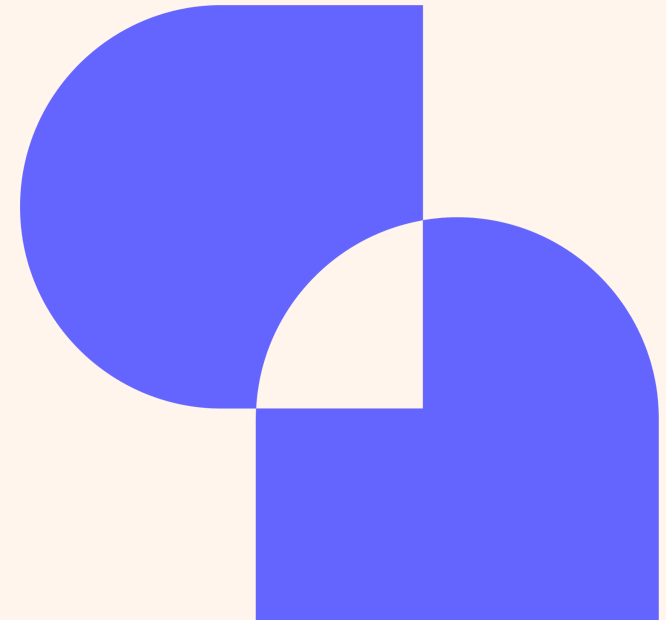
“Agir avec l'esprit éveillé”

---

## DOSSIER DE COMPÉTENCES

Développeur confirmé SwiftUI

Spécialisé : MVVM, SwiftUI



## PRINCIPALES EXPÉRIENCES

---

**BNP HELLO BANK :** Développeur Swift/SwiftUI (2025)

**CHANEL:** Développeur Swift/SwiftUI(2023 - 2024)

**NETATMO :** Développeur Swift/SwiftUI (2023)

**IPPON :** Développeur Swift/SwiftUI (2021-2022)

**Objectif :** *“Mon objectif est d'apporter une réel valeur ajoutée aux équipes de développement en apportant ma vision et mon expertise. Je souhaite aller au-delà de simple résolution de ticket et apporter ma pierre à l'édifice.”*

**Motivation :** *“En premier lieu, ma motivation repose sur un aspect technique, le développement est une passion pour moi. Aussi, évoluer dans des environnements avec différents types de profils, attise mon envie de découvrir et de partager mes idées.”*

**Passion :** *Course à pieds, échec et les jeux vidéos*

## CONNAISSANCES TECHNIQUES

---

**Langages :** Swift

**Librairies/framework :** SwiftUI, UIKit, Combine, Swift Package Manager, Async/Await, Foundation, Gestion Multimedia (AVFoundation, AVKit), Animation et Transition (Core Animation, Core Graphics), Widget, Phrase

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

**Base de données :** CoreData, Swift Data, GraphQL, API Rest

**Bonnes pratiques :** TDD, BDD, XCTest, TestFlight, Postman, TestUI system, Swiftlint, Pair Review

**Outils CI/CD :** XCode Cloud, Github Action, Fastlane, Buddy, Git, Gitflow, Crashlytics, Google Tag Analytics, Docker, SonarQube, Firebase distribution

**Gestion de projet :** Agile, Scrum, Figma, JIRA, Confluence, Notion

## EXPÉRIENCES TECHNIQUES

---

CHANEL: Développeur Swift/SwiftUI (2024)

Projet 1 : Développement from scratch d'une application dans le but de permettre aux utilisateurs de visualiser des produits de beauté sur leurs visages avec la VR.

- Analyse du besoin
- Ecriture du cahier des charges
- Écritures des spécifications fonctionnelles et techniques
- Échange avec la PO
  - Suivi des US avec l'aide de la PO
  - Suivi des maquettes déjà en place
- Binôme avec la PO
  - Développement supervisé par l'équipe iOS (4 personnes)
    - Code review
    - Pair programming
    - Ecriture de la documentation
  - Gestion de projet en Scrum
  - Participation cérémonies : weekly, démo, retro, sprint planning, sprint review, sprint refinement
  - Sprint de 2 semaines
- Mise en place de l'architecture MVVM :
  - Mise en place de la clean architecture : SOLID
  - Gestion de l'injection de dépendances

- Développement Swift :
  - Utilisation de AVFoundation pour la partie vidéo
  - Utilisation de la librairies OpenCV pour l'analyse du visage
  - Utilisation de Guards
- Développement SwiftUI :
  - Mise en place from scratch de SwiftUI
  - Intégration de vue en UIKit dans SwiftUI
  - Gestion de States Object
  - Gestion des Observed object
  - Gestion de la navigation
  - Gestion du Binding
  - Utilisation de Firebase pour du crashlytics et app distribution
- Versioning avec Git
  - Suivi du Gitflow
- Écriture des tests unitaires de XCTest :
  - Tester les ViewModels
- Utilisation de SwiftLint
- Utilisation de SonarQube pour l'analyse de code
- Suivie de la CI sur Azure DevOps

Projet 2 : Développement d'un catalogue from scratch pour la joaillerie de Chanel

- Analyse du besoin
- Ecriture du cahier des charges
- Écritures des spécifications fonctionnelles et techniques

- Échange avec la PO
  - Suivi des US avec l'aide de la PO
  - Suivi des maquettes déjà en place
- Équipes de 3 personnes
  - 1 tech lead, 1 développeur iOS et un PO
  - Développement supervisé par l'équipe iOS (4 personnes)
    - Code review
    - Pair programming
    - Ecriture de la documentation
  - Gestion de projet en Scrum
  - Participation cérémonies : weekly, démo, retro, sprint planning, sprint review, sprint refinement
  - Sprint de 2 semaines
- Mise en place de l'architecture MVVM :
  - Mise en place de la clean architecture : SOLID
  - Gestion de l'injection de dépendances
- Développement Swift :
  - Utilisation de AVFoundation pour la partie vidéo des pièces Chanel
  - Utilisation de Guards
- Développement SwiftUI :
  - Mise en place from scratch de SwiftUI
  - Intégration de vue en UIKit dans SwiftUI
  - Gestion de States Object
  - Gestion des Observed object
  - Gestion de la navigation
  - Gestion du Binding

- Utilisation de Firebase pour du crashlytics et app distribution
- Versioning avec Git
  - Suivi du Gitflow
- Écriture des tests unitaires de Swift Test
- Utilisation de SwiftLint
- Utilisation de SonarQube pour l'analyse de code
- Suivie de la CI sur Azure DevOps

Projet transverse Participation et accompagnement ponctuel sur le développement de différent projets

- Développement Swift :
  - Gestion du Lazy
  - Utilisation de Guards
- Développement SwiftUI :
  - Gestion de States Object
  - Gestion des Observed object
  - Gestion du Binding
  - Gestion du tracking avec Analytics
  - Gestion d'animations et de transitions
  - Gestion de gestures
- Développement UIKit :
  - Création d'un carrousel
  - Mise en place de la navigation

Environnement technique :

Swift, SwiftUI, AVFoundation, OpenCV, Lazy, Guards, UIKit, States Object, Observed Object, Binding, Firebase, MVVM,

SOLID, Git, Gitflow, XCtests, Swift Test, ViewModels, SwiftLint, SonarQube, Azure DevOps

---

#### NETATMO : Développeur Swift/SwiftUI (2024)

Participation à la migration du nouveau design de l'application Home+Security

- Suivi du besoin
- Suivi du cahier des charges
- Suivi spécifications fonctionnelles et techniques
- Équipe de 3 développeurs :
  - Un tech lead et deux développeurs
  - Gestion de projet en Scrum
  - Participation cérémonies : Daily, démo, retro, sprint planning, sprint review, sprint refinement
  - Sprint de 1 semaines
- Suivie de l'architecture MVVM :
  - Suivi de la clean architecture : SOLID
  - Suivi des designs patterns Factory
  - Gestion de l'injection de dépendances
- Développement Swift
- Développement en Swift UI :
  - Mise en place de gestures
  - Mise en place d'un carroussel
  - Utilisation de StateObject
  - Utilisation d'une API REST
- Versioning avec Git

- Suivi du Gitflow
- Écriture des tests unitaires de XCtests :
  - Tester les ViewModels
- Suivie de la CI sur Gitlab CI
- Bonne pratiques :
  - Code review
  - Écriture de la documentation

Environnement technique :

Swift, SwiftUI, State Object, API Rest, MVVM, SOLID, Factory, Git, Gitflow, XCTests, ViewModels, Gitlab CI

---

#### IPPON : Développeur Swift/SwiftUI (2022-2024)

Mission chez Chanel : Développement from scratch d'une application à destination des clients VIP de la partie joaillerie de chez Chanel

- Analyse du besoin
- Ecriture du cahier des charges
- Écritures des spécifications fonctionnelles et techniques
- Échange avec les PO
  - Suivi des US avec l'aide de la PO
  - Suivi des maquettes déjà en place
- Équipes de 5 personnes
  - 1 développeur iOS, un PO mobile, 2 développeurs full stack et 1 PO web
  - Gestion de projet en Scrum

- Participation cérémonies : daily, démo, retro, sprint planning, sprint review, sprint refinement
  - Sprint de 2 semaines
- Mise en place de l'architecture MVVM :
  - Mise en place de la clean architecture : SOLID
  - Gestion de l'injection de dépendances
- Développement Swift :
  - Utilisation de AVFoundation pour la partie vidéo
  - Utilisation de Guards
  - Utilisation de Tuist
  - Mise en place de cache
  - Utilisation de Combine
- Développement SwiftUI :
  - Mise en place from scratch de SwiftUI
  - Intégration de vue en UIKit dans SwiftUI
  - Gestion de States Object
  - Gestion des Observed object
  - Gestion de la navigation
  - Gestion du Binding
  - Utilisation de Firebase pour du crashlytics et app distribution
  - Mise en place de la gesture
  - Utilisation d'une API Rest
- Versioning avec Git
  - Suivi du Gitflow
- Écriture des tests unitaires de XCTest :
  - Tester les ViewModels
- Utilisation de SwiftLint

- Utilisation de SonarQube pour l'analyse de code
- Suivi de la CI sur Azure DevOps
- Bonnes pratiques
  - Mise en place de la documentation

Environnement technique :

Swift, SwiftUI, AVFoundation, Guards, Tuist, Combine, UIKit, States Object, Observed Object, Binding, Firebase, API Rest, MVVM, SOLID, Git, Gitflow, XCTest, ViewModels, SwiftLint, SonarQube, Azure DevOps

## FORMATION

---

INSA DE ROUEN - Diplôme d'ingénieur informatique

## LANGUES

---

Français

Anglais