

TECH LEAD ANDROID

13 ANS D'EXPERIENCE

Expert en Kotlin et Jetpack Compose

 +33 6 66 88 70 79 |  hatem.nouredine@gmail.com |  [hatem-noureddine](#)

PROFIL

Tech Lead Android expérimenté, spécialisé en Kotlin, Jetpack Compose et architectures modulaires (Clean Architecture, MVVM). J'allie une forte expertise technique à une solide expérience dans la gestion d'équipes pluridisciplinaires internationales. J'ai conçu et fait évoluer des applications mobiles et SDKs scalables, tout en garantissant qualité de code, performance et bonnes pratiques. Animé par le clean code, les tests et l'excellence technique, j'accompagne les équipes vers des solutions robustes et durables.

FORMATIONS

2012	Diplôme National d'ingénieur en informatique (génie logiciel). École supérieure privée d'ingénierie et de technologie.
2008	Diplôme d'un technicien supérieur en réseau Informatique. Institut supérieur d'informatique et des technologies de communications.

COMPETENCES

Architecture Mobile	Clean Architecture, MVVM, Architecture Modulaire, SOLID, KISS
Développement Android	Kotlin, Java, Android SDK, Jetpack libraries, Compose, room, Coroutines, LiveData, Koin, Gradle DSL, Google Maps, Ktor, Retrofit, CameraX, Mockkito, JUnit, Google Analytics
Bases de données :	MySQL, SQLite, Realm
Web Services :	REST, SOAP
Outils & Logiciels :	Jenkins, GitFlow, Junit, Sonar, Mockito, Robolectric, Glide, Fresco, Jackson, Moshi, Gson, Retrofit, Volley, OkHttp, Maven, Gradle, Artifactory, fastlane, Detekt, spotless, Docker, Charles, Sketch, Zeplin, figma, Mockflow, Espresso, Kaspresso, teamCity, Kotlin MultiPlatform, kover
Outil de gestion de projet :	Jira, Redmine, EasyRedmine, Confluence, Gitlab
Leadership technique & Gestion d'équipe :	Encadrement d'équipes, relecture de code, accompagnement technique, Recueil des besoins, rédaction de documentation technique, collaboration avec les équipes backend, définition des Features, backlog prioritisation, création des user story
Méthodologie :	SCRUM, SAFe,

EXPERIENCES PROFESSIONNELLES

TECH-LEAD Android

JUILLET 2022- JUIN 2025
STELLANTIS

➤ Piloter, assister et Développer la nouvelle application client « Space »

- Responsable du développement Android de l'application Space, le SDK « Framework » et le SDK « middleware » en collaboration étroite avec les équipes Produit, les équipes backends, l'équipe de design et QA pour livrer des fonctionnalités robustes sur plusieurs versions.
- Conception et implémentation d'un SDK « middleware » évolutive dédié aux services de véhicule non connecté intégrant l'orchestration réseau, la gestion du cache, la transformation des données et la gestion de session. Le SDK unifiait plusieurs backends Stellantis en normalisant les schémas de données, les formats d'API et les messages d'erreurs offrant ainsi une interface propre et cohérente tout en masquant la complexité d'intégration.
- Conception et implémentation d'un SDK « Framework » dédié aux composants graphiques partagé par différentes applications du groupe intégrant la gestion du wording à travers des fichiers Json et la mise à jour du visuel en temps réel suivant la marque sélectionnée offrant ainsi des composants graphiques optimisés et simples tout en masquant la complexité d'intégration.
- Définir avec l'équipe de design la structure des **tokens** générée par **Figma** afin de simplifier le maintien et le développement des composants graphiques pour toutes les marques.
- Contribution aux choix techniques, accompagnement de développeurs et réalisation de revues de code pour assurer un haut niveau de qualité.
- Mise en place d'une architecture modulaire basée sur les features selon les principes de la **Clean Architecture** et du **MVVM**, facilitant les tests unitaires, la maintenabilité du code et la scalabilité.
- Utilisation de Kotlin Coroutines et Flow pour des opérations asynchrones réactives et sensibles au cycle de vie.
- Intégrations de **Firebase Crashlytics** pour le monitoring des crashes, facilitant la détection, l'analyse et la résolution proactive des erreurs en production.
- Utilisation d'outils IA comme **GitHub Copilot**.
- Création de différents scripts Gradle pour générer des classes Kotlin à partir des design tokens partagés par l'équipe design, mise à jour des fichiers wording à partir du serveur et générer un binaire suivant la configuration de chaque marque.
- Maintien d'une couverture de code, avec des **tests unitaires**, supérieure à 85 %.
- Mettre en place des différentes tâches sur la CI/CD pour maintenir la qualité de code, respecter le coding style et simplifier le processus de release.
- Utiliser **Sonar** et **Detekt** pour faire respecter les bonnes pratiques Kotlin et uniformiser la qualité du code.
- Utiliser **ktlint** pour faire respecter le style de codage et avoir des pull-request propres.
- Gestion de livraison + gestion du **GIT**.
- Optimisation des performances avec **Android Profiler** (analyse CPU, mémoire, rendu) et surveiller les appels réseaux via **AppInspection**.
- Rédaction de spécifications techniques claires et structurées sur **Confluence** pour accompagner les implantations et l'alignement entre équipes.

Environment technique: Clean Architecture, MVVM, Multi-module, SOLID, KISS, Kotlin, Ktor, Koin, Room, Coroutine, flow, jetpack compose, jetpack navigation, Gradle, Charles, Leakcanary, Proguard, GitHub Copilot, JUnit, Sonar, TeamCity, python, GIT, Github, Atomic design, Figma, StyleDictionary

TECH-LEAD Android

JANVIER 2021- JUIN 2022
PSA

➤ **Piloter, assister et Développer une application blanche modulaire qui représente les différentes marques du groupe.**

- Piloter et faire monter en compétences une équipe de **4 développeurs**.
- Développement de l'application selon les principes de la **Clean Architecture** et du **MVVM**, assurant des transitions d'état prévisibles et une séparation claire des modules.
- Intégration de **l'Atomic Design** et développement des composants graphiques qui prend en charge la charte graphique de chaque marque.
- Mettre en place une gestion de thème afin de passer du thème d'une marque à une autre en temps réel.
- Développement des fonctionnalités sous forme de **modules séparés indépendantes**.
- Utilisation des observateurs avec **Kotlin-flow** afin de mettre à jour les différents écrans en temps réel.
- Optimiser l'utilisation des ressources du smartphone avec le **Profiler**
- Détection des **fuites mémoires** avec leak-canary et les corriger.
- Ajout des tests d'instrumentations et des tests unitaires.
- Utiliser le **Kotlin-dsl** avec **Gradle** pour centraliser la configuration des modules
- Ajouter un **script Gradle** pour représenter les **modules** et leurs **intercommunications** suivant un schéma en utilisant graphviz.
- Ajouter un **script Gradle** pour générer un log avec les **dépendances** qui sont **obsoletes** et les dernières versions des dépendances utilisées.
- Ajouter un **git-hook** pour exécuter les scripts qui vérifie la **qualité de code** et formate le code avant chaque commit.
- Intégrer **Sonar, Detekt et Spotless** pour maintenir la qualité de code

Environment technique: Kotlin, MVVM, Coroutine, Flow, Livedata, Room, Koin, Retrofit, Moshi, Timber, Proguard, Gradle, Kotlin-dsl, Charles, Leakcanary, Atomic design, kaspresso, Junit, Sonar, Mockito, Robolectric, GIT, Github, Scrum.

DEVELOPPEUR ANDROID SENIOR

SEPTEMBRE 2020-DECEMBRE 2020
PSA

➤ **Refonte du SDK et des composants communicants avec les voitures des différentes marques du groupe.**

- Refonte du SDK de communication avec les véhicules du groupe, avec adoption de Kotlin et Coroutines, pour une exposition réactive des données.
- Optimiser l'utilisation des ressources du smartphone avec le **Profiler**
- Détection des **fuites mémoire** avec leak-canary et les corriger
- Ajouter une **documentation** détaillée sur l'utilisation du SDK.
- Ajout **des tests unitaires**.
- Intégrer **Sonar, Detekt et Spotless** pour maintenir la qualité de code

Environment technique : Kotlin, Coroutine, Flow, Room, Retrofit, Moshi, Proguard, Gradle, Kotlin-dsl, Charles, Junit, Sonar, Mockito, Robolectric, GIT, GitLab, Scrum.

DEVELOPPEUR ANDROID SENIOR

SEPTEMBRE 2018 – Août 2020

JOHN PAUL – CONCIERGERIE ACCOR HOTELS

➤ Développement d'une nouvelle application modulaire pour JOHN PAUL la société de conciergerie du Groupe ACCCOR HOTELS.

- Développement de l'application avec une architecture modulaire basée sur les features selon les principes de la **Clean Architecture** et du **MVVM**, facilitant les tests unitaires, la maintenabilité du code et la scalabilité.
- Intégration de l'**Atomic Design** et développement des composants graphiques
- Génération d'un formulaire à partir d'un fichier de **Config JSON**
- Refonte de la couche réseau pour utiliser les **Coroutines**.
- Utiliser le **Kotlin-dsl** avec **Gradle** pour centraliser la configuration des modules
- Intégrer **Sonar**, **Detekt** et **Spotless** pour maintenir la qualité de code

Environment technique: Kotlin, MVVM, Coroutine, Livedata, Koin, Retrofit, Moshi, Atomic design, Junit, Sonar, Mockito, Robolectric GIT, GitLab, Proguard, Gradle, Kotlin-dsl, Charles, Fastlane, Sentry, Leakcanary, Scrum.

➤ Développement de nouvelles fonctionnalités pour les applications VISA Apac, VISA Platinum, VISA Platinum Business et la nouvelle application StartConnect de Toyota

- Refonte du code commun sous forme de modules.
- Ajouter une gestion d'erreurs et la gestion du cache pour la couche réseau.
- Migration de tout le code en **Kotlin**.
- Migration vers **Androidx**
- Corriger les règles de proguard.
- Développement des nouveaux parcours autour de la création d'une requête(booking).
- Crypter les informations sensibles échanger avec le serveur (préparation à la certification PCI)
- Ajouter de la pagination et du filtre pour le contenu (liste des offres, liste des priviléges...)
- Contribution à l'amélioration du processus de relecture, développement et la chaîne d'intégration Gitalb CI.

Environnement technique : Kotlin, Yaml, Junit, Mockito, Robolectric, GIT, GitLab, Proguard, Gradle, Charles, Fastlane, Slack, Stripe SDK, Adyen SDK, Scrum.

DEVELOPPEUR ANDROID SENIOR

DECEMBRE 2017 – SEPTEMBRE 2018

LCL – CREDIT LYONNAIS

➤ Développement de nouvelles fonctionnalités pour l'application bancaire LCL – Mes Comptes

- Refactorisation du code et passage à une **architecture MVP**.
- Migration de 30% du code en **Kotlin**.
- **Optimisation des Layouts en utilisant des ConstraintLayout**

- Migration de la couche réseau de **Volley vers Retrofit** et ajout du **SSL pinning** en se basant sur **les Hash des certificats**.
- Ajout de la fonctionnalité permettant de faire les virements permanents et différés.
- Ajout d'une nouvelle catégorie « news et dialogue box » pour mettre en avant les nouvelles fonctionnalités de l'application aux utilisateurs.

Environnement technique : Kotlin, Jenkins, Junit, Sonar, Mockito, Robolectric GIT, GitLab, Proguard, Gradle, Confluence, Jira, Charles.

LEAD DEVELOPPEUR ANDROID

JANVIER 2012 – NOVEMBRE 2017
PROXYM-IT AGENCY

Intervention en tant développeur senior / lead dev pour plusieurs clients de l'agence.

PROJET SDK SCANTOPAY (3 MOIS)

- **Conception et développement d'une SDK pour dématérialiser les dépôts des chèques :**
Développement d'un SDK de dépôt des chèques pour différentes banques.

- Étude de faisabilité technique et mise en place de l'architecture de l'application
- Intégration du SDK A2IA pour la reconnaissance des chèques.
- Configuration du rendu du design suivant le retour du WS en temps réel.
- Récupération de l'historique des dépôts de chèques
- Affichage des détails du dépôt avec la possibilité de récupérer l'image d'un chèque en mode asynchrone sous format Bitmap.
- Création d'un composant permettant d'afficher la caméra avec la gestion des permissions en runtime et la détection d'un document (dessiner la zone détectée).
- Reconnaissance d'un chèque avec optimisation de la taille des images.
- Assombrissement du code avec Proguard avec génération de la documentation javadoc et un binaire récupérable depuis un dépôt Maven Local avec un Sample.

Environnement technique : Javadoc, EasyRedmine, A2IA, FCM, Proguard, Gradle, MavenLocal., GIT, Retrofit, Crashlytics, Eclipse, SVN, Robospice, Camera, Google Analytics, SQLite.

PROJET ALLIANZ CONNECT (9 MOIS)

- **Conception et développement de l'application Allianz connect (Application d'assurance) :**
permet de déclarer des sinistres et des accidents, demander de l'assistance, avoir un espace de stockage cloud privé.

- Participation à la rédaction des spécifications, à la proposition et à la validation de la charte graphique.
- Ajout des écrans et des besoins validés par le client au **Mockflow et Zeplin** en partageant les informations avec l'équipe iOS.
- Étude de faisabilité technique et **mise en place de l'architecture de l'application**.
- Utilisation d'une **architecture modulaire** afin de simplifier l'intégration des modules pour d'autres projets.
- Développement et conception des composants graphiques personnalisés.
- Conception et développement des modules de gestion des marqueurs sur Google Maps, de gestion des appels Web Services REST, de synchronisation d'images avec **Amazon S3 et de gestion de cache avec la base de données Realm**.

- Optimisation des images avant l'upload afin d'alléger le temps d'envoi des données.

Environnement technique : Architecture modulaire, Crashtytics, FCM, MVP, EventBus, Fresco, GoogleMaps, Calligraphy, GPS, Google Analytics, Realm, Amazon S3, ButterKnife, GIT, EasyRedmine,

PROJET BOZER (4 MOIS)

➤ Conception et développement de l'application BOZER (Application de VOIP) :

- Étude de faisabilité technique et mise en place de l'architecture de l'application.
- Chiffrage des fonctionnalités et élaboration de l'offre financière.
- Mise en place de l'architecture et des composants spécifiques pour l'application.
- Intégration de la **SDK Sinch** pour VOIP avec gestion d'audio pour les sonneries.
- Création d'une sonnerie depuis la voix d'une durée qui ne dépasse pas les 10 secondes.
- Application du **matériel Design** avec des ressources vectorielles pour alléger sa taille.
- Fonctionnalités de partage sur les réseaux sociaux

Environnement technique : Android Studio, Crashtytics, Retrofit, ButterKnife, GCM, Fresco, GIT, Jenkins, EasyRedmine, Calligraphy, Google Analytics, Realm, Gradle, Sinch, SVG.

PROJET VANCLEEF & ARPEL (5 MOIS)

➤ Conception et développement de l'application VanCleef & Arpel Android (Application de guide à un expo) : permettant d'accompagner les visiteurs dans leurs visites.

- Étude de faisabilité technique des nouvelles fonctionnalités de l'application.
- Mise en place de l'architecture et des composants spécifiques pour l'application.
- Ajout d'un système de guide audio qui se déclenche seulement à la présence d'un kit oreiller.
- Développement des mini-jeux soit avec des drag & drop, soit utilisation de la caméra avec du canvas pour simuler de la réalité augmentée, etc...
- Interaction avec les beacons pour déterminer la position dans les salles.
- Fonctionnalités de partage sur les réseaux sociaux.

Environnement technique : Android Studio, Crashtytics, Estimote, Calligraphy, Google Analytics, NirbeeSDK, Gradle, ButterKnife, GIT, EasyRedmine,

PROJET TCHAPPER (7 MOIS)

➤ Conception et développement de l'application Tchapper (Application de chat) permettant d'échanger des messages, soit en privé soit en groupe, sous forme multimédia.

- Mise en place de l'architecture et des composants spécifiques pour l'application.
- Création des extensions personnalisées pour les échanges avec XMPP suivant le besoin du client comme l'envoi des messages texte, image, photo, vidéo ou audio, l'envoi d'un message éphémère.
- Cryptage et décryptage des messages avec une protection par touchID ou par un code visuel.
- Import des contacts avec reconnaissances de ceux disposants de l'application Tchapper.

Environnement technique : Android Studio, Crashtytics, Google Analytics SQLite, SMACK XMPP, Retrofit2, Glide, DBFlow, ButterKnife. TouchID, GIT, EasyRedmine,

PROJET DIOR (2 MOIS)

➤ **Conception et développement de l'application DIORTV (application publicitaire)** permettant de visualiser des vidéos suivant les différentes catégories de la marque DIOR.

- Étude de la faisabilité technique, Mise en place de l'architecture et des composants spécifiques pour l'application avec prise en charge de plus de 5 langues.
- Implémentation de la gestion des favoris, mode accès offline et d'un lecteur vidéo avec prise en charge des sous-titres avec la notion du **Cast avec le ChromeCast, AirPlay, ...**

Environnement technique : Android Studio, Retrofit, Robospice, Exoplayer, Facebook Api, GIT, Trello, Crashtytics, Google Analytics, SQLite, , Twitter, Google Plus, ChromeCast, AirPlay, Github

PROJET GOSSIP (2 MOIS)

➤ **Conception et développement de l'application Gossip** permettant d'échanger des gossips, sous forme d'une rumeur ou d'une preuve visuelle, avec les contacts de votre répertoire et/ou compte Facebook.

- Étude de faisabilité technique, mise en place de l'architecture et des composants spécifiques.
- Traitement des images et des vidéos avec ffmpeg.
- Validation de l'authentification par SMS (Parse).
-

Environnement technique : Android Studio, Crashtytics, Google Analytics, Facebook Api, Google Plus, Github, Parse, Digits, GIT, Trello, FFMPEG.

PROJET BETERISE (2 MOIS)

➤ **Conception et développement de l'application Betterise (application m-health)** permettant de prendre soin de sa santé en fournissant des conseils à partir des informations collectées.

- Étude de la faisabilité, mise en place de l'architecture et des composants spécifiques pour l'application.
- Utilisation de Google Fit pour récolter les données liées au nombre de pas.

Environnement technique : Android Studio, REST WebServices, Retrofit, Robospice, Google Analytics, C2DM, Google Fit, JSON, SVN.

PROJET MYDESIGN (3 MOIS)

➤ **Conception et développement de l'application MyDesign (application m-commerce)** permettant la personnalisation des produits d'une boutique comme T-Shirt, Étui, Coussin ...

- Étude de la faisabilité et mise en place de l'architecture et des composants spécifiques.
- Mise en place des moyens de paiements comme **PayPal et PayBox**.

- Gestion de planning.

Environnement technique : Robospice, Eclipse, SVN, Retrofit PayPal, PayBox, Prestashop, Traitement d'images, Google Analytics, C2DM.

PROJET CONSOSMART (7 MOIS)

- **Conception et développement d'une application de promotions et de conseils :** permet de mieux consulter les offres du moment sur différents produits avec la gestion de ses gains.
 - Ajout des scans/saisies des codes-barres, la prise de plusieurs preuves d'achats, ...
 - Gestion des gains et les dépagnote via un virement bancaire.
 - Affichage des magasins et les offres en se basant sur la géolocalisation.

Environnement technique : Eclipse, SVN, Simulateur Android, QR Code, code à barres, multiscreen, Google Analytics, b, Retrofit, Robospice, Gson, Camera, GPS.

PROJET PORT DE NICE (2 MOIS)

- **Conception et développement d'une application de guide touristique**
 - Ajout d'une carte personnalisée et géolocalisée qui affiche la zone du port de Nice.
 - Ajout des endroits touristiques liés à la région niçoise (textes, images, vidéos, ...)

Environnement technique : Android, Eclipse, Robospice, Retrofit, JSON, SQLite, GPS, Simulateur Android, SVN, OpenStreetMap.

PROJET WALK IT (7 MOIS)

- **Conception et développement d'une application de communication :** permet de faire le suivi et la participation aux événements en se basant sur la géolocalisation de l'utilisateur.
 - Participation aux évènements de circuit ex : festival, marche, manifestation...
 - Échange avec les participants (agrégation des flux liés à l'évènement).
 - La géolocalisation des amis sur le circuit de l'évènement (checkpoints).

Environnement technique : Eclipse, Java, Android SDK, Retrofit, Rebospice, GPS, GoogleMaps, JSON, Web services et SVN