TECH LEAD ANDROID
13 YEARS OF EXPERIENCE
Expert in Kotlin and Jetpack Compose

📞 +33 6 66 88 70 79 | ✉ hatem.nouredine@gmail.com | 💼 hatem-noureddine

PROFILE

Experienced Android Tech Lead, specializing in Kotlin, Jetpack Compose, and modular architectures (Clean Architecture, MVVM). I combine strong technical expertise with solid experience in managing international multidisciplinary teams. I have designed and developed scalable mobile applications and SDKs, while ensuring code quality, performance, and best practices. Driven by clean code, testing, and technical excellence, I guide teams toward robust and sustainable solutions.

TRAINING

2012	National Diploma of Computer Engineering (Software Engineering). **Private Graduate School of Engineering and Technology.**
2008	Diploma of a senior technician in computer networking. **Higher Institute of Computer Science and Communication Technologies.**

COMPETENCES

Mobile Architecture	Clean Architecture, MVVM, Architecture Modulaire, SOLID, KISS
Android Development	Kotlin, Java, Android SDK, Jetpack libraries, Compose, room, Coroutines, LiveData, Koin, Gradle DSL, Google Maps, Ktor, Retrofit, CameraX, Mockkito, JUnit, Google Analytics
Databases	MySQL, SQLite, Realm
Web Services	REST, SOAP
Tools and Software	**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
Project management tool	Jira, Redmine, EasyRedmine, Confluence, Gitlab
Technical Leadership and Team Management	Team management, code review, technical support, requirements gathering, writing technical documentation, collaboration with backend teams, defining features, backlog prioritization, creating user stories
Methodology	SCRUM, SAFe,

PROFESSIONAL EXPERIENCE

TECH-LEAD Android   **JULY 2022 – JUNE 2025**
                                                                        STELLANTIS

➢ **Manage, support and develop the new "Space" customer application**
  - Responsible for the Android development of the Space application, the Framework SDK, and the Middleware SDK, working closely with the Product, Backend, Design, and QA teams to deliver robust functionality across multiple releases.
  - Designed and implemented a scalable Middleware SDK dedicated to offline vehicle services, integrating network orchestration, cache management, data transformation, and session management. The SDK unified multiple Stellantis backends by standardizing data schemas, API formats, and error messages, providing a clean and consistent interface while masking integration complexity.
  - Designed and implemented a Framework SDK dedicated to graphic components shared by various group applications, integrating wording management through JSON files and real-time visual updates based on the selected brand, thus providing optimized and simple graphic components while masking integration complexity.
  - Define with the design team the structure of the tokens generated by Figma to simplify the maintenance and development of graphic components for all brands.
  - Contribute to technical choices, support developers, and conduct code reviews to ensure a high level of quality.
  - Implement a modular, feature-based architecture based on the principles of Clean Architecture and MVVM, facilitating unit testing, code maintainability, and scalability.
  - Use of Kotlin Coroutines and Flow for reactive, lifecycle-aware asynchronous operations.
  - Firebase Crashlytics integrations for crash monitoring, facilitating the detection, analysis, and proactive resolution of errors in production.
  - Use of AI tools such as GitHub Copilot.
  - Created various Gradle scripts to generate Kotlin classes from design tokens shared by the design team, updated wording files from the server, and generated a binary file based on each brand's configuration.
  - Maintained code coverage, with unit tests, above 85%.
  - Implemented various CI/CD tasks to maintain code quality, adhere to coding style, and simplify the release process.
  - Used Sonar and Detekt to enforce Kotlin best practices and standardize code quality.
  - Used ktlint to enforce coding style and maintain clean pull requests.
  - Managed releases and GIT.
  - Optimized performance with Android Profiler (CPU, memory, and rendering analysis) and monitored network calls via AppInspection.
  - Wrote clear and structured technical specifications on Confluence to support implementations and team alignment.

Technical environment: 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

➢ **Manage, assist and develop a modular white application that communicates with the different brands of the group.**

- Manage and develop the skills of a team of **2 developers**.
- Development of the application with **the MVVM architecture.**
- Integration of **Atomic Design** and development of graphic components
- Set up theme management in order to switch from one brand's theme to another in real time.
- Development of functionalities as **independent separate modules.**
- Use observers with **Kotlin-flow** to update the different screens in real time.
- Optimize the use of smartphone resources with the **Profiler**
- Detection of **memory leaks** with leak-canary and fix them.
- Added instrumentation tests and unit tests.
- Use **Kotlin-dsl** with **Gradle** to centralize module configuration
- Add a **Gradle script** to represent modules and their **intercommunications** in a schema using graphviz to represent it on the **Readme**.
- Add a **Gradle script** to generate a log with the **dependencies** that are **outdated** and the latest versions of the dependencies used.
- Add a **git-hook** to run the scripts that checks the **code quality** and formats the code before each commit.
- Integrate **Sonar, Detekt and Spotless** to maintain code quality

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

SENIOR ANDROID DEVELOPER   **SEPTEMBER 2020-DECEMBER 2020**
PSA Group

➢ **Redesign of the SDK and the components communicating with the cars of the different brands of the group.**

- Redesign of the SDK for communication with the group's vehicles, adopting Kotlin and Coroutines for responsive data exposure.
- Optimize the use of smartphone resources with the **Profiler**
- Detecting and correcting **memory leaks** with leak-canary
- Add detailed **documentation** on how to use the SDK.
- Added **unit tests**.
- Integrate **Sonar, Detekt and Spotless** to maintain code quality

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

SENIOR ANDROID DEVELOPER   **SEPTEMBER 2018 – August 2020**
                                    JOHN PAUL – CONCIERGERIE ACCOR HOTELS

➢ **Development of a new modular application for JOHN PAUL, the concierge company of the ACCCOR HOTELS Group.**

- Development of the application with **the MVVM architecture.**
- Integration of **Atomic Design** and development of graphic components
- Development of functionalities in the form of separate modules**.**
- Development of the **Login and Register modules.**
- Generating a form from a **JSON Config file**
- Redesigned the network layer to use **Coroutines.**
- Use **Kotlin-dsl** with **Gradle** to centralize module configuration
- Integrate **Sonar, Detekt and Spotless** to maintain code quality

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

➢ **Development of new features for the VISA Apac, VISA Platinum, VISA Platinum Business and Toyota StartConnect applications**

- Redesign of the common code in the form of modules**.**
- Add error handling and cache management for the network layer**.**
- Migration of all code to **Kotlin.**
- Migration to **Androidx**
- Bug fixes
- Correct proguard rules.
- Development of new journeys around the creation of a request (booking).
- Encrypt sensitive information, exchange with the server (PCI certification preparation)
- Add pagination and filter for content (list of offers, list of privileges...)
- Contribution to the improvement of the proofreading process, development and the Gitalb CI integration chain.

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

SENIOR ANDROID DEVELOPER   **DECEMBER 2017 – SEPTEMBER 2018**
                                                LCL – CREDIT LYONNAIS

➢ **Development of new features for the LCL banking application – My Accounts**

- Code refactoring and moving to an **MVP architecture.**
- Migrated 30% of the code to **Kotlin.**
- **Optimizing Layouts Using ConstraintLayouts**
- Migration of the network layer from **Volley to Retrofit** and addition of **SSL pinning** based on **certificate hashes.**
- Added functionality to make standing and deferred transfers.
- Added a new "news and dialog box" category to highlight the new features of the application to users.

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

LEAD DEVELOPPEUR ANDROID   **JANUARY 2012 – NOVEMBER 2017**
                                                                    PROXYM-IT AGENCY

Intervention as a senior developer / lead dev for several of the agency's clients.

SCANTOPAY SDK PROJECT (3 MONTHS)

➢ **Design and development of an SDK for digitizing check deposits:** Development of a check deposit SDK for various banks.

- Technical feasibility study and implementation of the application architecture
- Integration of the A2IA SDK for check recognition
- Configuration of the design rendering based on real-time WS feedback
- Retrieval of check deposit history
- Display of deposit details with the ability to retrieve a check image asynchronously in Bitmap format.
- Creation of a component to display the camera with runtime permission management and document detection (drawing the detected area).
- Check recognition with image size optimization.
- Code obfuscation with Proguard with javadoc documentation generation and a binary retrievable from a Local Maven repository with a Sample.

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

ALLIANZ CONNECT PROJECT (9 MONTHS)

➢ **Design and development of the Allianz connect application (Insurance application):** allows you to report claims and accidents, request assistance, have a private cloud storage space.

- Participation in the drafting of specifications, proposal and validation of the graphic charter.
- Added customer-validated screens and requirements to the **Mockflow and Zeplin** by sharing the information with the iOS team.
- Technical feasibility study and **implementation of the application architecture.**
- Use of **a modular architecture** to simplify the integration of modules for other projects.
- Development and design of custom widgets.
- Design and development of the modules for managing markers on Google Maps, **managing Web Services REST calls**, synchronizing images with **Amazon S3 and cache management with the Realm database.**
- Optimization of images before upload in order to reduce the time it takes to send data.

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

BOZER PROJECT (4 MONTHS)

➢ **Design and development of the BOZER application (VOIP application):**

- Technical feasibility study and implementation of the application architecture.
- Costing of functionalities and development of the financial offer.
- Implementation of the architecture and specific components for the application.
- Integration of the **Sinch SDK** for VOIP with audio management for ringtones.
- Create a ringtone from voice with a duration of no more than 10 seconds.
- Application of **Design material** with vector resources to lighten its size.
- Social media sharing features

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

VANCLEEF & ARPEL PROJECT (5 MONTHS)

➢ **Design and development of the VanCleef & Arpel Android application (Guide to an exhibition application):** to accompany visitors during their visits.

- Technical feasibility study of the new features of the application.
- Implementation of the architecture and specific components for the application.
- Addition of an audio guide system that is triggered only by the presence of a pillow kit.
- Development of mini-games either with drag & drop, or use of the camera with canvas to simulate augmented reality, etc...
- Interaction with beacons to determine the position in rooms.
- Social media sharing features.

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

PROJECT TCHAPPER (7 MONTHS)

➢ **Design and development of the Tchapper application (Chat application)** allowing you to exchange messages, either privately or in groups, in multimedia form.

- Implementation of the architecture and specific components for the application.
- Creation of custom extensions for exchanges with XMPP according to the customer's needs such as sending text, image, photo, video or audio messages, sending an ephemeral message.
- Encrypt and decrypt messages with touchID or visual code protection.
- Import of contacts with recognition of those with the Tchapper application.

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

DIOR PROJECT (2 MONTHS)

➢ **Design and development of the DIORTV application (advertising application**) allowing you to view videos according to the different categories of the DIOR brand.

- Technical feasibility study, Implementation of the architecture and specific components for the application with support for more than 5 languages.
- Implementation of favorites management, offline access mode and a video player with subtitle support with the notion of **Cast with ChromeCast, AirPlay**, ...

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

GOSSIP PROJECT (2 MONTHS)

➤ **Design and development of the Gossip application** to exchange gossips, in the form of a rumor or visual evidence, with the contacts in your Facebook directory and/or account.

- Technical feasibility study, implementation of the architecture and specific components.
- Image and video processing with ffmpeg.
- Validation of authentication by SMS (Parse).
-

Environnement technique: Android Studio, Crashlytics, Google Analytics, **Facebook Api**, Google Plus, GitHub, **Parse**, **Digits**, **GIT**, Trello, FFMPEG.

PROJECT BETERISE (2 MONTHS)

➤ **Design and development of the Betterise application (m-health application)** to take care of one's health by providing advice from the information collected.

- Feasibility study, implementation of the architecture and specific components for the application.
- Use of Google Fit to collect step count data.

Technical environment: Android Studio, REST Webservices, **Retrofit**, **Robospice**, Google Analytics, C2DM, Google Fit, Json, **SVN**.

MYDESIGN PROJECT (3 MONTHS)

➤ **Design and development of the MyDesign application (m-commerce application)** allowing the customization of a store's products such as T-Shirt, Case, Cushion ...

- Feasibility study and implementation of the architecture and specific components.
- Implementation of payment methods such as **PayPal and PayBox.**
- Planning management.

Technical environment: Robospice, Eclipse **SVN**, Retrofit PayPal, PayBox, Prestashop, Image processing, Google Analytics, C2DM.

CONSOSMART PROJECT (7 MONTHS)

➤ **Design and development of a promotions and advice application:** allows you to better consult the current offers on different products with the management of your winnings.

- Addition of scans/entries of barcodes, the taking of several proofs of purchase, ...

- Manage winnings and distribute them via bank transfer.
- Display of stores and offers based on geolocation.

Technical environment: Eclipse, SVN, Android Simulator, QR Code, barcode, multiscreen, Google Analytics, b, **Retrofit**, **Robospice**, Gson, Camera, **GPS**.

PORT OF NICE PROJECT (2 MONTHS)

➢ **Design and development of a tour guide app**
- Addition of a customized, geotagged map that displays the area of the port of Nice.
- Addition of tourist places related to the Nice region (texts, images, videos, ...)

Technical environment: Android, **Eclipse**, **Robospice**, **Retrofit**, JSon, SQLIte, GPS, Android Simulator, SVN, OpenStreetMap.

WALK IT PROJECT (7 MONTHS)

➢ **Design and development of a communication application:** allows you to track and participate in events based on the user's geolocation.
- Participation in circuit events e.g. festivals, walks, demonstrations, etc.
- Exchange with participants (aggregation of flows related to the event).
- The geolocation of friends on the event circuit (checkpoints).

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