

# Hadi Lashkari Ghouchani

## Software Engineer

@ hi@hadilq.com    hadilq.com    Toronto, Canada  
github.com/hadilq    @hadilq@androiddev.social    linkedin.com/in/hadilq

## Summary

A curious developer with over 18 years of experience in software development, where almost 8 years of his experience covers Android development, over 2 years covers teaching algorithms and programming in universities, and over 8 years covers academic and self-employed experiences while he was a student. An Introvert person who has profound learning and contributing skills. A creative, value-driven developer who enjoys doing math, solving problems and creating scalable, maintainable, testable and efficient apps to make the world a better place to live. Advocate of bayesian mindsets. As an architect, he pursue reproducible, frictionless, consistent, and scalable structures. A person whose goal is to surround himself with creative, clever, motiveative and happy engineers, so he could continuously learn more and more, not just learning, of course. In fact, he has valuable knowledge and sturdy knowledge transfer skills to share. He's especially good at understanding, designing, and applying software architectures, all kind of automations, including CI/CD, deep diving into unknown areas, eg., fixing ANRs, creating tools, eg., improving productivity of engineers, optimizing software, eg., decreasing build time. Additionally, he's a big fan of SSOT, principle of ownership, isolation principle, SOLID principles especially DIP.

## Experience

### Productivity Engineer

at [MobileLIVE](#)    Aug. 2022 - Present    Toronto, Canada

Promoted to Productivity Engineer to make tools and automate processes of the development life cycle to improve productivity.

Mission:

- Investigated problems related to proxy and self-signed certificates in VPN networks, where by implementing a script, and by using a Jetpack Compose Desktop app to install them automatically increased the build count in GE dashboard by 164%.
- Raised and implemented the need for Gradle Conventional Plugins to reduce duplication and errors. This also applied SSOT on versions of external libraries, which reduced the runtime crashes due to version conflicts.
- Raised the idea and created a workflow POC in Gitlab to showcase how to achieve code quality by using Merge Trains and Merge request pipelines.
- Decreased the time of running tests by 18% by using Robolectric.

Additional Technologies:

- Gradle Enterprise, Gradle Conventional Plugin, Jetpack Compose Desktop.

### Software Engineer

at [MobileLIVE](#)    Apr. 2021 - Aug. 2022    Toronto, Canada

MobileLIVE is a Canadian tech-service provider unifying people, technology, and business to accelerate digital transformation. Over this time, he worked in Bell Canada as a full-time contractor. Bell Canada is subsidiary Bell Aliant provides services in the Atlantic provinces. It provides mobile service through its Bell Mobility (including Virgin Mobile Canada) subsidiary, and television through its Bell Satellite TV and Bell Fibe TV subsidiaries.

Mission:

- Introduced RFCs to make technical decisions and steps to applying on any new idea.
- Added value by teaching and discussing SOLID principles, and best practices of modularization, feature flags, and monorepo, etc.
- Raised and implemented the idea of automatically cloning and building all the source code in the multi-repo setup to reduce version conflicts, and improve code visibility for debugging purposes.

Technologies:

- Kotlin Coroutines / Mockk, Mockito, JUnit, etc.
- Agile, Sprint, Lane based branching strategy, CI/CD
- Atlassian Jira, Atlassian Confluence, Gitlab, Jenkins, Slack, Branch.io, inVision, Dynatrace, etc.

### Staff Android Developer

at [iFood](#)    Oct. 2020 - Mar. 2021    Barcelona, Spain

iFood was an ambitious project to build an entire food delivery ecosystem for all around the world. Currently, it was the biggest food delivery in Brazil, Colombia and Mexico, with 50+ million apps installed on Android devices alone. The Android project was highly modularized to make it possible for 50+ developers working on it frictionlessly.

Mission:

- Proposed and implemented composite build to handle dependency management in Gradle. It decreased the incremental build time while updating a library from more than 13 min to less than a minute.
- Implemented Android side of metadata project. Its purpose was gathering all information of the highly modularized Android/iOS projects in a dashboard along backend information.
- Proposed and applied a concept to move out the legacy code from the root module. It decreased the incremental build time by 55%.
- Detected many ANRs and resolved them by applying architectural principles.
- Wrapped up external libraries to better handling panic situations.

Technologies:

- Dagger 2, Room, Kotlin Coroutines, Moshi, WorkManager / Mockk, Robolectric, JUnit, etc.
- Agile, Sprint, Github Flow, CI/CD, Gradle Enterprise
- Atlassian Jira, Atlassian Confluence, Github, Jenkins, Slack, Databricks, Firebase, Bugsnag, Logzio, etc.

### Senior Android Developer

at [Letgo](#)    Oct. 2019 - Mar. 2021    Barcelona, Spain

Letgo was one of the biggest classified marketplaces in US, and also, the biggest player in Turkey with 15+ million apps installed on Android devices alone. TL;DR Mission: Improved the performance of loading feeds by at least a minute.

### Lead Android Developer

at [Sheypoor](#)    Jan. 2018 - Jan. 2019    Tehran, Iran

Sheypoor was one of two leading classified marketplaces in Iran with 1.5+ million users per day, 13+ million overall users and 344+ million sessions. TL;DR Mission: Increased the number of users per day from 0.5+ million to 1.5+ million in a year.

### Senior Android Developer

at [Myket](#)    Sep. 2015 - Des. 2017    Tehran, Iran


Myket was one of two leading app stores in Iran with over 8+ million users. TL;DR Mission: Implemented features in the Myket app store, started TDD, installed Jenking's pipeline to handle CI/CD.


### Java and C# developer

at [Soha System](#)    Mar. 2015 - Aug. 2015    Tehran, Iran


SOHA was formed to focus on solutions based on smart cards. TL;DR Mission: Implemented Android and desktop apps.


## Article / Repository


 2022 Sample App: Refactored a Mastodon client app to a modularized Jetpack Compose-driven architecture, LTA, which is designed by himself.


 2023 Post: Dependency Inversion Principle


 2023 Contributing to nixpkgs/androidenv: Nix is a language that is designed to have reproducibile output.


 2022 Post: Logic Tree in Android


 2022 Sample App: Molecule sample app to solve some scoping problems in Android development.


 2022 Post: Reproducibility


 2021 ProAndroidDev post: Are feature flags, build variants, and sample apps related concepts?


 2021 iFood Engineering post: Modularization tip: optimize the Root Module.


 2021 ProAndroidDev post: Happy Railway. An improvement for Railway Oriented Programming in Kotlin.


 2021 Library: Happy is a library to auto-generate Kotlin code to make calling APIs more concise.


 2020 POC: Rust Flutter Reactive is an idea to use Flutter as UI and Rust for the main project.


 2020 Make Log4k Multiplatform: Simple logging library for Android, iOS, Mac, etc.


 2020 ProAndroidDev post: A/B Testing Myths


 2020 Sample App: Provide a full implementation for a No-Framework Dependency Injection in Kotlin


 2020 Contributing on Arrow Library: Arrow is a library for Typed Functional Programming in Kotlin


 2020 Library: A Kotlin compiler plugin that generates trackable method


 2020 Medium post: Save State By Using RxJava or Kotlin Flow


 2020 Medium post: RxJava instead of LiveData in MVVM


 2020 Library: A glue between the lifecycle of LifecycleOwner in Android and upstreams in RxJava


 2020 Library: A glue between the lifecycle of LifecycleOwner in Android and Kotlin Coroutines


 2020 Library: Simplifier for unnecessary complex lifecycles


 2019 Sample App: Functional Programming + Clean Architecture


 2019 Sample App: Movie Challenge


 2019 Math Library: Modularized, mathematical and multiplatform Kotlin library


 2018 ProAndroidDev post: Logging in Kotlin


 2018 Log4k Library: Easy logging library, but unfortunately it's in Sheypoor repository

 2018 ProAndroidDev post: Reactive MVVM architectural pattern





 2018 Sample App: Clean Architecture + RMVVM

 2018 ProAndroidDev post: Loading entities from database and server API by using RxJava

 2018 ProAndroidDev post: LiveData with single events

 2018 LiveEvent library: It's useful to propagate single event to View

## Education

Sharif University Of Technology	M.Sc	in Physics	 Feb. 2010	 Tehran, Iran
Payame Noor University	B.Sc	in Physics	 Jul. 2007	 Mashhad, Iran

## Tools / Skills

### Android

Partially Open Source Mobile  
Lifecycles/LiveData/Navigation/Paging/ViewModel  
Multidex/Test/Data Binding/WorkManager/Room  
Android Architecture Components/JetPack  
Anvil/Dagger 2/RxJava 2/Retrofit 2/...  
Kotlin Multiplatform/Kotlin Coroutines  
Kotlin Compiler Plugin/KSP/...  
Compose/Data Binding/View Binding/...

TL;DR why he's interested:

- It's partially open source OS.
- Generally interested on Mobile Technologies.

### Flutter

StatelessWidget/StatefulWidget  
StreamController/Isolates  
FFI

TL;DR why he's interested:

- Generally interested on all cross-platform and multi-platform solutions.

### Rust

tonic/prost/openssl/tokio/flatbuffers/thrift  
Rc/Arc/Mutex  
FFI

TL;DR why he's interested:

- Rust's type system is what you need for scale-up.
- Borrow Checker restrictly stops any memory leaks.
- Bevy, and Dioxus frameworks look like good candidates for cross-platform solutions.

### Docker

Buildkit/Compose/Cache

TL;DR why he's interested:

- Reproducibility is the key to productivity.
- Isolation is the key to reproducibility.

### Nix

NixOS/Nix Darwin  
mkDrivation/mkShell

TL;DR why he's interested:

- Reproducibility is the key to productivity.
- Isolation is the key to reproducibility.
- Due to respecting to purity, it provides better caching experience than Docker.

### Architecture

Clean Architecture/Hexagonal Architecture  
Unidirectional data flow Architecture  
Logic Tree Architecture  
MVVM/MVI/MVP/RMVVM

### Principle

Clean Code/SOLID  
Object Oriented Programming  
Functional Programming  
Stable Dependencies Principle  
Stable Abstractions Principle

### Design Pattern

Factory/Builder/Singleton/...  
Adapter/Facade/Proxy/Visitor/Delegate/...  
Strategy/Chain of responsibility/...

### Android test library

JUnit/Mockito/Espresso/JaCoCo/...

### Build Tool

Gradle/Cargo/Sbt/Maven/Ant/Make/...

### Command Line Tool

grep/sed/netstat/awk/docker/...  
adb/sdkmanager/avdmanager/...

### Development Tool

Pivotal Tracker/Jira/Jenkins/Travis/CircleCi/...

### Programing Language

Java/Kotlin/Rust/Scala/Haskell/Python  
Groovy/Dart/Javascript/Ruby...

### Workflow

Trunk Based development/Github flow/Git Flow  
Cross-functional team

### Operating System

Linux(Nix OS/Fedora/Ubuntu/Debian), OS X

## Special Honor

- 2020 Had an offer from MobileLIVE, Toronto, Canada
- 2020 Had an offer from iFood, Remote
- 2019 Had an offer from Kik, Kitchener, Canada
- 2019 Had an offer from Letgo, Barcelona, Spain
- 2019 Had an offer from CRITICAL Techworks, BMW, Porto, Portugal
- 2019 Had an offer from The Next Big Thing, Berlin, Germany
- 2018 Received result-oriented badge in Sheypoor
- 2007 Ranked 44th among more than five thousand participants in the National Graduate Entrance Exam in Physics

## Language

English: Advance  
Farsi: Native

The latest version of this resume is available in [github](#)

Built on November 19, 2023