

Hadi Lashkari Ghouseh

Software Engineer

@ hi@hadiiq.com

📍 hadiiq.com

📍 Toronto, Canada

📄 github.com/hadiiq

✂ x.com/@hadiiq

in linkedin.com/in/hadiiq

Summary

A humble and curious developer with over 19 years of experience in software development, where 9 years of his experience covers professional 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 introverted person who has profound 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 a software architect, he pursue reproducible, frictionless, consistent, and scalable structures. A person whose goal is to surround himself with creative, clever, motivative and happy engineers. He's especially good at understanding, designing, and applying software architectures. Additionally, he's a big fan of Single Source of Truth principle(SSOT), principle of ownership, isolation principle, pure functions/stateless design, SOLID principles especially DIP. Since his study time in university, he was deeply interested in the machine learning and its applications in human life, so he has been accumulating his knowledge in this area for a long time. On the other hand, Android development becomes more boring over time, thus, he's actively chasing to focus on ML more than ever. He believes an open source resume would contribute to his credibility, even though, because of that the bullet points in this resume are degrading faster than usual in the market, therefore, just reach out. Maybe we'll be good friends.

Experience

Productivity Engineer

at [MobileLIVE](#)

📅 Aug, 2022 - Present

📍 Toronto, Canada

Promoted to "Productivity Engineer" of Mobility team in Bell Canada 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%.

- The single point of training for the build, and supporting for its related issues in the team of 50+ Android developers.

Additional Technologies:

- Developcity(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

Java and C# developer

at [Soha System](#)

📅 Mar. 2015 - Aug. 2015

📍 Tehran, Iran

Article / Repository

- 🔗 2024 Hair Segmentation: A ML model to detect and segment hair in images
- 🔗 2024 dragon detector: A simple ML model to detect dragon in images
- 🔗 2024 Sample Rust App: A sample app to crawl an API and put the data in a database.
- 🔗 2024 Nix Effect pod: Modularize effects of all kinds, and stay in-control, with NixOS and podman.
- 🔗 2023 Sample App: Refactored a Mastodon client app to a modularized Jetpack Compose-driven architecture.
- 📄 2023 Post: Dependency Inversion Principle
- 🔗 2023 Contributing to nixpkgs/androidenv: Nix is a language that is designed to have reproducible 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

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
Lifecycle/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.

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 restricts any memory leaks.

Podman/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
mkDrvation/mkShell

TL;DR why he's interested:

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

Software 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

Nix/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

Kotlin/Rust/Python/Java/Scala
Groovy/Dart/JavaScript/Ruby...

Workflow

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

Operating System

Linux(NixOS/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 June 15, 2024