VOL. X | CURRICULUM VITÆ | 2026 EDITION

Raphael Bernardo

Senior Android Engineer · *iOS & tvOS with SwiftUI*
Ex-Uber (contract) · Kotlin and Jetpack Compose for phone, tablet and TV, native JNI/C++, my own apps on Google Play and the App Store, and applied AI from coding agents to a pay-per-call API for AI agents.

MAIL raphaelbgr@gmail.com TEL +55 21 98885-6697 GIT github.com/raphaelbgr
IN linkedin.com/in/raphaelbgr

LOC Rio de Janeiro · GMT−3 CIT Brazilian + Portuguese (EU) dual citizen
LANG Portuguese (native) · English (fluent) · Spanish (intermediate) AVAIL Open to a new engagement
ENG Offshore B2B contractor (own CNPJ) · full US Eastern overlap

§ 01 Summary

PROFILE · POSITIONING

Senior Android engineer for phone, tablet and TV, with my own iOS and tvOS apps on the App Store and applied AI. Ten-plus years shipping production Android apps — Uber's Maps SDK (JNI/C++), payments and offline-first data, and OTT streaming with 4K DRM-protected playback (Widevine, PlayReady) across Android TV, Fire OS, webOS and Tizen. I build and publish apps for iPhone, iPad and Apple TV in Swift and SwiftUI, and build AI systems end to end: an AI-native workflow with coding agents and automated review, agent orchestration across my own machine fleet, and a live pay-per-call API for AI agents.

§ 02 Experience

ENGAGEMENTS · 2014 → PRESENT

Nexstar Media Group, Inc.

Aug 2025 — Present · current

Senior Fire OS / Android TV Engineer · full remote CONTRACT

- Contributed to the architecture of an enterprise **OTT streaming framework** powering **multiple TV station brands** across Fire TV and Android TV using **MVVM + Clean Architecture** with a plugin-based integration layer for third-party services.
- Built five specialised **Media3 / ExoPlayer** implementations covering VOD, live, and audio playback with HLS adaptive bitrate.
- Shipped DRM-protected playback (**Widevine, PlayReady**) with both server-side and client-side ad insertion.
- Integrated the platform's analytics and audience-measurement stack under **GDPR / CCPA** compliance.
- Delivered TV-optimised auth (device-based), entitlement & flexible paywall, and zero-downtime feature rollouts across a large multi-brand flavor matrix via remote config; built Leanback TV UI with full d-pad navigation.
- Refactored callback chains to reactive Kotlin **SharedFlow**; resolved fragment-lifecycle leaks; wired **Firebase Crashlytics + New Relic** with JUnit, Mockk, Robolectric, Espresso coverage.

- Run a **client-approved agent-driven delivery workflow** end to end: an agent picks up the tracker issue with its acceptance criteria, works the change on an isolated branch, runs the test suite alongside agent-driven end-to-end checks against a running build, and opens a **GitHub** pull request carrying the issue context and test evidence. **GitHub Actions** runs the pipeline; a second model audits the diff against the original request before anything is reported done.

`Kotlin` `Coroutines / Flow` `Media3 / ExoPlayer` `Dagger 2` `Retrofit2` `Moshi` `Glide` `Android Leanback` `Jetpack (Nav, ViewModel, Room)` `Firebase` `Google IMA` `New Relic` `JUnit · Mockk · Espresso`

FreeCast, Inc.

Mar 2021 — Jul 2025 · 4 yrs 5 mo

Android / Android TV / Fire OS Engineer · part-time · EST hours · full remote CONTRACT

- Led the **full migration of the legacy Java codebase to Kotlin**; introduced Use-Case + Clean Architecture and improved end-user UX.
- Implemented **HLS / DASH** playback for **live linear TV and VOD** with Widevine DRM-protected streaming; integrated Google Cast and a paid subscription gateway.
- Built and maintained the Select TV and FreeCast Watch products across **Android, Android TV, Fire OS, webOS and Tizen**.
- Integrated **OTA tuner platforms** (HDHomeRun, AirTV, Alticast, PDAQ) with GStreamer-based media pipelines for secure live broadcast over HLS / DASH.
- Owned **DRM, EPG, tuner orchestration, transcoding**, and multi-device streaming end-to-end.
- Shipped on Google Play (*Select TV Android*, *FreeCast Watch*) and Amazon (*FreeCast Watch Fire OS*).
- Used **AI-assisted development** on the Java-to-Kotlin migration and on widening test coverage, later with **coding agents as a primary development method** on migration and refactor work, with automated verification in CI before review.

`Kotlin` `Jetpack Compose` `Hilt` `Coroutines / Flow` `Media3` `Room · DataStore` `Retrofit2` `Picasso` `Firebase Analytics · Crashlytics · FCM` `Mixpanel` `webOS · Tizen` `HDHomeRun · AirTV · Alticast · PDAQ` `GStreamer` `Widevine DRM`

Claro TV+

Apr 2024 — Dec 2024 · 8 mo

Android TV / Fire OS Engineer · part-time · full remote FREELANCE VIA GLOBAL HITSS

- Greenfield Android TV / Fire OS app for one of Brazil's largest pay-TV operators; **4K streaming** via DASH / HLS with multi-DRM.
- Contributed to the data layer as a shared **Kotlin Multiplatform (KMP)** submodule consumed by Android, webOS and Tizen clients.
- Multi-module **MVVM** architecture; **Bitmovin Player** playback; analytics across Firebase + Google Analytics 4.
- Applied **AI-assisted development** across the greenfield build — code generation and review over the Kotlin Multiplatform and multi-module layers.

`Kotlin Multiplatform (KMP)` `Kotlin` `Bitmovin Player` `Ktor` `Koin` `Coroutines / Flow` `Jetpack` `Firebase + GA4` `webOS · Tizen`

Uber Technologies

May 2021 — May 2022 · 1 yr

Android Engineer · Maps SDK team · via Matchpoint Solutions · PST hours · full remote CONTRACT

- Worked on the **Maps team SDK** module — JNI/C++ bindings between Uber's native Maps engine and the Android client.
- Worked in Uber's **RIBs** architecture inside the internal monorepo (not the public repository); the SDK module shipped into Uber Rider, Uber Eats and Uber Driver.

- Operated inside a 10K+ engineer codebase using **Buck, Flipper**, custom analytics / logger / crash reporter and a feature-by-module architecture.

`Java · Kotlin` `C++ · NDK · JNI` `Groovy` `Python · SQL` `RIBs` `Buck` `CMake` `Flipper` `Firebase`

Super Revendedores

Oct 2018 — Feb 2021 · 2 yrs 5 mo

Android Engineer · full remote

- Migrated the full codebase from Java to Kotlin; implemented **online payment gateway**, virtual showcases and an SSO link to a partner web ordering portal.
- Built scalable asynchronous order processing on **AWS SQS / SNS** for reliable queuing and notification delivery.

`Kotlin` `AWS (SQS · SNS · Storage)` `Firebase (Firestore · Auth · Remote Config · IAM · FCM)` `Facebook OAuth` `DbFlow` `Dagger 2` `MVP` `Retrofit2`

Concrete · an Accenture Company

Dec 2016 — Oct 2018 · 1 yr 10 mo

Jr. Android Engineer

- Built the **Minha CVC** travel app from its initial implementation as part of the agency team — **1M+ downloads**; shop geolocation directing customers to physical stores; itinerary feature reducing customer-service call volume.
- Worked on **Getnet**; migrated MVP to MVVM; introduced RxJava2 and LiveData.

`Java 8` `Kotlin` `RxJava2 · LiveData` `MVP → MVVM` `Dagger 2` `Google Maps API` `Retrofit2 · Picasso` `Firebase` `Jenkins CI/CD`

PraPagar (fintech)

Jun 2015 — Dec 2016 · 1 yr 6 mo

Jr. Android Engineer · collaborating with US team (MaxxPotential)

- Built mobile payments via **QR code and geolocation**; first Android + Laravel REST experience.

`Android` `PHP · Laravel 5` `REST` `OOP · Design Patterns`

IBM Brasil

Jun 2014 — Dec 2014 · 7 mo

Software Engineering Intern · Rio de Janeiro · while pursuing B.S. at Instituto Infnet

- Wrote **SQL queries against IBM DB2** in support of contract and sales teams; produced reporting spreadsheets and Hyperion-based BI artefacts.
- Documented internal processes via SharePoint; first professional exposure to enterprise databases, REST APIs, and Agile workflows.

`IBM DB2` `SQL` `IBM Hyperion` `SharePoint` `Python` `REST APIs` `Git`

FASOLTI · Fábrica de Soluções em TI

Jan 2014 — May 2014 · 5 mo

Test Developer · `INTERNSHIP`

- Documented and implemented automated test cases in **Selenium with TestNG**.

§ 03 Published Applications

APP STORE · GOOGLE PLAY · WEB · API



Archive — Home Video Player

ANDROID TV · APPLE TV (TVOS)

TV client for my self-hosted media library, built natively for each platform: HLS playback, server-side transcoding, library browsing and resume across devices.

Kotlin, Media3, Compose for TV, Room · Swift, SwiftUI, AVFoundation, HLS

Google Play App Store



IPTV Player

APPLE TV (TVOS) · IPHONE & IPAD (IOS)

Player for the user's own Xtream accounts and M3U playlists: live TV with an EPG guide, movies and series with resume, HLS, MP4 and MPEG-TS playback, an on-device library that loads in small batches so large lists stay responsive, and private iCloud sync.

Swift, SwiftUI, AVKit, GRDB (SQLite), CloudKit, StoreKit 2

App Store · tvOS & iOS



Beer Price Calculator

ANDROID · IPHONE & IPAD (IOS)

Compares beer prices by volume to find the best value per millilitre, built natively for each platform. Offline-first with local persistence and in-app billing on Android.

Kotlin, Jetpack Compose, Room, Play Billing · Swift, SwiftUI

Google Play App Store



ADAMA Studio

WEB APP FOR ARCHITECTS

Turns a floor plan into an engineering model in the browser: upload the plan, let the AI draft it in 3D, review it in conversation and in 2D, isometric and room views, then export to DXF, IFC and Revit. Each account connects its own AI provider.

Next.js, React, TypeScript, Three.js, Better Auth

vilanova.fastsoftware.uk



RugShield

PAY-PER-CALL API FOR AI AGENTS · PERSONAL AGENTIC PROJECT

A live API that AI agents call, and pay for per call, to check a Solana token or an unsigned transaction before signing, or to ask several independent risk advisors in parallel. Agents pay in USDC on Base or Solana through x402 (HTTP 402 payments, no account or API key), over plain HTTP or as MCP tools. A durable job store saves each paid result before the payment settles, so a lost response can be recovered without charging twice.

Python, FastAPI, x402, MCP, SQLite, Fly.io

getrugshield.com

§ 04 AI-Native Engineering Practice

BUILT AND RUN ON MY OWN MACHINES AND TIME · NOT IN ANY CLIENT CODEBASE

Multi-provider LLM orchestration

A dispatch daemon routing coding work across my own seven-machine fleet, exposed to the orchestrating agent as **15 MCP tools**.

- Cost-aware routing over **8 providers and 22 models in 6 tiers**, applying rate-limit backoff, per-provider cooldowns and capability matching in order.
- Flat-rate providers are marked **serial-only**, so a parallel fan-out cannot burn a subscription's concurrency limit.
- Encrypted job transport keyed off the existing SSH identity — **RSA-OAEP + AES-256-GCM**, fresh key and nonce per message — with five-source peer discovery under an explicit precedence ladder.

Instrumentation as a definition of done

I authored a language-neutral **trace-contract** standard: the compile-time reference graph is the specification, and the runtime trace is reconciled against it automatically.

- Ships a written spec, JSON Schemas, a cross-language conformance oracle, a golden-case corpus, and an agent plugin whose hook blocks completion while the graph is stale.
- Gate severity is scoped — agent-authored edits block at the definition of done; human commits degrade to a lint error, never a test failure.

Closed-loop evaluation on my own TV app

My personal tvOS and Android TV media player exposes an on-device HTTP diagnostic surface — trace, state, screenshot, scripted navigation, and **149 named test routines**.

- Each routine declares required parameters and returns a structured pass/fail or a machine-readable 422. An agent drives the app, captures what the screen actually shows, reads the emitted events and evaluates against the contract.
- Compiled out of release builds.

Fleet operations

Manifest-driven deployment across five personal machines under launchd, systemd and Task Scheduler.

- Each application declares its install, launch and health contract; deployment is a git push. Supervision and auto-update run on decoupled intervals so a crash restarts in seconds. Cross-machine control is RSA-signed per request.

§ 05 Education

FORMAL · CONTINUING

Instituto Infnet · Rio de Janeiro

2013 — 2018

B.S. Computer Engineering — software development, mathematical modelling, OS & hardware architecture.

§ 06 Technical Stack

CORE COMPETENCIES

CORE	Android · Applied AI / LLM · Kotlin
ANDROID · TV	Android SDK · Android TV · Fire OS · webOS · Tizen · Jetpack Compose · Leanback · Kotlin Multiplatform · NDK · JNI
APPLE · CAR	iOS · tvOS · Swift · SwiftUI · UIKit · Apple CarPlay · Android Auto
DATA	SQL · SQLite · Room · PostgreSQL · Supabase · RLS · IBM DB2
STREAMING	Media3 / ExoPlayer · HLS · DASH · Live · VOD · Widevine · PlayReady · SSAI · CSAI · Google IMA · GStreamer
ARCHITECTURE	MVVM · MVI · Clean Architecture · Use Case · Multi-module · Coroutines · Flow · Hilt · Dagger 2
AI SYSTEMS	Agent orchestration · MCP · RAG · vector search · Evaluation harnesses · Local inference · Python
DELIVERY	JUnit · Mockk · Robolectric · Espresso · Gradle · GitHub Actions · Jenkins · Jira