

Abraham José Kazerian Blengio

Senior backend developer. I build and speed up serverless systems on AWS with Node.js and TypeScript. 8+ years shipping software, and since 2023 for a U.S. financial client.

Montevideo, Uruguay · hello@abrahamkazerian.dev · abrahamkazerian.dev · linkedin.com/in/abraham-kazerian-3ba470ba

ABOUT

I'm a backend developer from Montevideo, Uruguay. Since 2017 I've shipped products for fintech, crypto, retail and healthcare. I started full stack with Angular, React, React Native and Node.js.

Since 2023 I've worked through DEPT for a U.S. financial client, building serverless back ends on AWS with Node.js and TypeScript. I care about systems that are fast and easy to maintain: I built the shared middleware our Lambda functions run on, and cut execution time by 70% on the heaviest ones.

I work in English every day, and I'm open to remote roles with U.S. and European teams as well as freelance projects.

EXPERIENCE

DEPT

Senior Software Developer · Jan 2023 - Present

- Contractor for a U.S. financial client: serverless back ends on AWS Lambda with Node.js and TypeScript, plus React on the front end.
- Built a shared Lambda middleware (validation, logging and error handling) reused across functions, making the code easier to maintain and read.
- Cut execution time by 70% (2.5 s → 800 ms) on the heaviest Lambda functions over millions of records: query, join and index tuning plus a full rewrite of the backend logic (see the case study).
- Plan and deliver features end to end, and review code across the team.

Globant

Senior Software Developer · Dec 2021 - Dec 2022

- NestWealth (National Bank of Canada): new features for a portal to open bank accounts online, with KYC, personal data and document signing. Angular, NestJS and PostgreSQL.
- Researched and introduced NgRx as a single source of truth for the state shared across components, ending the data inconsistencies between them, and left it as a pattern any teammate could follow to add new state.

Switch Software Solutions

Mid-level Software Developer · Feb 2021 - Dec 2021

- Tata (Uruguayan supermarket chain): React Native app to order groceries and pick them up in store, with a Node.js and MySQL backend. Improved a backend whose requests were taking far too long.

SpaceDev

Software Developer · Sep 2018 - Feb 2021

- Ndax: crypto exchange (buy, sell, price alerts). Angular, Node.js, MongoDB and AlphaPoint.
- Flexio: payment-link platform that lets merchants sell in installments, integrated with SR.Pago. Angular, Node.js and MongoDB.
- CP&A: offline-first Ionic app for sales reps to place supermarket orders.
- Somos.UY: platform to hire nurses to care for family members. AngularJS, Node.js and MongoDB.

Full-time study

Sep 2017 - Sep 2018

Altimetrik

Software Developer · May 2017 - Sep 2017

- Migrated a U.S. client's legacy Salesforce components to Lightning components.

Universidad ORT Uruguay

2014 - 2016

- Information Technology Analyst, 3-year degree (Analista en Tecnologías de la Información).

CASE STUDY

-70% execution time on a critical Lambda

Current role at DEPT · for a U.S. financial client

PROBLEM	The heaviest Lambda ran a complex query that pulled a large amount of data over millions of records, and spent much of its time sorting them in the database.
WHAT I DID	Attacked it on several fronts: simplified the queries, reworked the joins, improved the indexes and rewrote the backend logic behind it from scratch. The rewrite also took the sorting out of the database: results are grouped in an object keyed by a small numeric ID (at most ~2,000 values), which JavaScript iterates in ascending order. Instead of an ORDER BY over millions of rows, only a couple of thousand groups end up sorted.
RESULT	From ~2.5 s to ~800 ms (600 ms in the best case): about 70% less execution time, plus lower CPU and memory usage, measured with CloudWatch and Splunk.

Node.js · TypeScript · AWS Lambda · SQL · CloudWatch · Splunk

SIDE PROJECT

tialuk

NOT LAUNCHED

THE IDEA	Small businesses answer sales DMs from their phones, across several accounts. tialuk brings them into one inbox in real time, flags intent and urgency, summarizes each conversation and helps draft replies, masking personal data before it ever reaches the AI model.
WHAT I BUILT	— Meta webhooks verified with HMAC and queued in a transactional outbox on PostgreSQL (FOR UPDATE SKIP LOCKED): fast acknowledgement, idempotent processing and real-time updates over SSE. — AI on Vertex AI (Gemini) for intent, urgency, summaries and draft rewrites, with PII redaction, a per-tenant usage ledger, daily caps and prepaid credits to keep costs under control. — Multi-tenant security: Meta tokens encrypted with AES-256-GCM, JWT in HttpOnly cookies with rotating refresh, Argon2id and OAuth state bound to the session. — Infrastructure as code with Terraform on GCP (Cloud Run, private Cloud SQL, Secret Manager, keyless CI/CD with Workload Identity Federation) and Cloudflare Workers. Deploys run migrations first, smoke-test and roll back automatically.
WHERE IT STANDS	It reached a working end-to-end dev environment, but it hasn't launched: Meta's App Review and the production setup are still pending.
CODE	Private repository. I'm happy to walk through it in an interview, or to grant temporary read-only access on request.
HOW	Built solo between May and July 2026 with an AI-assisted workflow (Claude Code), backed by design docs and recorded architecture decisions for every key choice.

~22k lines of backend TypeScript ~965 automated tests 50+ API endpoints

Bun · ElysiaJS · TypeScript · PostgreSQL · MikroORM · SvelteKit · Vertex AI · Terraform · GCP · Cloudflare Workers · GitHub Actions

SKILLS

SERVERLESS ON AWS	Lambda · SQS · SNS · API Gateway · DynamoDB · ECS · CloudWatch
BACKEND	Node.js · TypeScript · JavaScript · NestJS · Jest · Bun · Vertex AI · Gemini
INFRASTRUCTURE & DELIVERY	Terraform · Google Cloud · AWS CDK · Serverless Framework · Docker · GitHub Actions · Datadog
DATABASES	PostgreSQL · MySQL · MongoDB
FRONTEND & MOBILE	React · React Native · Angular · Ionic · SvelteKit