Roberto López

Innovation Engineer · Mexico
rolope@rolope.com · github.com/rolope

SUMMARY

Software engineer with 10 years of experience, the last six building the WhatsApp conversational-commerce platform behind bots for
Banorte, Banco Azteca, Invex, Chedraui, Soriana, Coppel, Estafeta and Essity. I worked across the whole stack: .NET messaging services on RabbitMQ and Redis, the JavaScript library ~219 production bots are built on, NestJS client APIs, and AI features (LLM agents, receipt OCR) inside banking journeys. I led the launch of the platform's bot runtime and set engineering standards (BDD/TDD, feature flags, Conventional Commits, service generators).

SKILLS

- **Languages:** C#, JavaScript/TypeScript, Java, Swift, Python, SQL, PHP
- **Backend:** .NET Core/5/6/8, NestJS, Java EE, Node.js, Laravel, Clean Architecture/CQRS (MediatR), Prisma, Liquibase
- **Messaging & data:** RabbitMQ, Redis/ElastiCache, MongoDB/DocumentDB, MySQL
- **Mobile:** iOS (Swift)
- **WhatsApp:** Meta Cloud/Graph API, WhatsApp Flows, catalogs & cart, templates, lists, CTA, carousels
- **AI:** LLM agents, function calling, confidence-based disambiguation, OpenAI vision OCR, speech-to-text, Claude Code
- **Cloud & delivery:** AWS (Fargate, S3, SES, ECR), Kubernetes/Helm, Docker, GitHub Actions, GitLab CI, SonarQube, New Relic
- **Quality:** BDD/TDD (SpecFlow, Cucumber/Gherkin, Jest), load testing (BlazeMeter), feature flags & A/B testing

EXPERIENCE

Auronix — Innovation Engineer · Mexico · Mar 2020 – Jun 2026

Conversational-commerce platform serving Mexican banks, retailers and brands over WhatsApp.

Fintech & AI collections (2025 – 2026)

- Created the preventive-collections AI bot (promise to pay, LLM collections agent, function-calling FAQ, low-confidence disambiguation menus), later extended by the team.
- Built AI receipt OCR inside WhatsApp Flows for Invex and a multi-client payment-report flow: detects the issuing bank, rejects handwritten receipts, enforces hard timeouts.
- Built core pieces of Banorte's WhatsApp collections: the collections bot (CSAT, deeplinks, AI-agent handling), the CSAT survey's persistence layer and the collections API.
- Moved Banco Azteca's service-rating survey cache from in-memory to Redis/ElastiCache; added collections endpoints to the gateway serving 100+ client integrations.
- Built the Medallia webhook connector and Julio delivery notifications.

Platform standards (2025 – 2026)

- #2 contributor to the company NestJS archetype (Nx): introduced Conventional Commits; wrote the create-app CLI and the generators for observability, health checks, Prisma + Liquibase persistence and deployment type.
- Validated a generated service end to end on the shared GitLab CI/CD pipeline and fed the fixes back into the archetype; set up AI-assisted development (Claude Code, openspec).
- Top contributor to the company architecture docs (Chedraui, OXXO Spin Premia, Spin by OXXO).

Conversational commerce & WhatsApp features (2023 – 2024)

- Shipped WhatsApp Flows (native in-chat forms) across the bot library, platform translator and Meta Cloud API bridge; created the `WhatsAppFlows` module in the shared bot library.
- Created the `FeatureFlag` module (GitLab flags, percentage rollouts) and bot-to-bot handoff with preserved metrics, enabling Estafeta's bot A/B tests.
- Built WhatsApp cart and catalog shopping on VTEX and Meta Catalog for Essity TENA, retail/CPG demos and Chedraui; built a voice-note transcription demo and RFP FAQ bot for Coppel.
- Upgraded the Meta Graph API integration from v12 to v19; added CTA buttons, sectioned lists, multi-product and carousel messages (#2 contributor to the bridge).
- Introduced BDD/TDD (SpecFlow, Coverlet) in the bot runtime and BDD conventions with automated deploys in the bot CLI.

Leading the FlowsJS bot runtime (2022)

- Kicked off and led the .NET service that runs JavaScript bots on live WhatsApp traffic; ran its develop → rc → master release train. Production instances now autoscale from 5 to 30 replicas on Kubernetes.
- Split ~10k lines of bot libraries out into a versioned npm package, decoupling bot development from service deploys.
- Set up SonarCloud, migrated production to a new AWS Fargate cluster and added BlazeMeter load tests for Soriana to CI.

Messaging platform foundations (2020 – 2021)

- Built Redis primitives (distributed locks, atomic list operations, fault-tolerant timeouts) in the shared .NET library every messaging service depends on.
- Added bot protocols (bots 2.0, Flow, FlowJS) and session billing events; documented the rating engine rules and conversation data model.
- Maintained the mass-mailing app (AWS SES/SMTP): resilient DB pooling, open/click tracking, and the Log4Shell patch shipped on 2022-01-03.

KEA — App Developer · Mexico · Jun 2016 – Feb 2020

- Developed the backend (Java EE) and the iOS app for InovaFIT, Inova's fitness app launched in 2019: personalized workout plans from the user's profile and goal, guided routines and nutrition plans, progress tracking with smart-band data, membership with a free trial, and weekly live classes.
- Helped build the coupon and membership-discount system for Sixtynite, a motel-booking app in Mexico (iOS, Android, Huawei): coupon creation and rules, redemption and validation, tiered member discounts that grow with repeat visits (10% up to 30%), and the backend API and admin panel to manage them.

SELECTED PROJECTS

- **@auronix/build-core** — shared bot library behind ~219 WhatsApp bots; created the WhatsApp Flows and Feature Flag modules, main contributor to WhatsApp Cart.
- **Larii (freelance, 2026)** — offline-first household-organizer PWA on Firebase; passwordless auth, role-based Firestore/Storage security rules covered by automated emulator tests.

LANGUAGES

Spanish (native) · English