# João Lucas Andrade Madureira

Software Developer | Go · TypeScript · PostgreSQL

Belém, Pará, Brazil | jlmadureira90@gmail.com | joao-madureira.vercel.app | linkedin.com/in/lucasmadureira0
github.com/lucasonline0

## Professional summary

Software developer and Computer Science undergraduate with hands-on experience in public-sector production systems. Works across APIs, relational databases, web applications, and administrative workflows using Go, TypeScript, PostgreSQL, and Linux, with an interest in backend engineering and application security.

## Professional experience

**Software Developer Intern**, State Department of Education of Pará (SEDUC-PA) – Belém, Pará, Brazil 2025 - Present

- Develop and maintain institutional systems, REST APIs, business rules, authentication/authorization, SQL queries, reports, and dashboards using Go, Next.js, NestJS, PostgreSQL, and Prisma.
- Contribute across frontend, backend, databases, testing, documentation, and code review; use Git/GitHub, CI/CD, Docker, Railway, and Vercel for incremental delivery and production fixes.
- Contribute to architecture and security decisions involving RBAC, role-based data isolation, authorization-bypass prevention, HTTP hardening, and secure session handling.

## Professional projects

**Operational and Structural School Census - SEDUC-PA** 2026 - Present

Go · Chi · PostgreSQL · Next.js · TypeScript · Docker

- Developed APIs and dashboards for school data collection and analytics; built infrastructure, school-meal, and outsourced-service endpoints validated with real payloads covering 822 schools.
- Implemented cascading global filters, school/INEP-code filters, and server-side pagination while preserving parameterized SQL and existing contracts.
- Enforced session-derived Regional Education Directorate (DRE) scope before calculations, rankings, pagination, and reports; prevented query-parameter overrides and applied fail-closed behavior.
- Implemented server-side closure of submissions, blocking API writes while preserving reads, reports, and history; the change was released as an isolated production hotfix.

**SADEP - Probationary Service Performance Evaluation** 2026 - Present

Next.js · NestJS · TypeScript · Prisma · PostgreSQL

- Built authenticated flows, role-aware navigation, RBAC, supervisor workspaces, and employee dashboards supporting drafts, submission, and rectification; kept state and available actions controlled by the backend.
- Fixed 14 vulnerabilities after a security review, covering rate limiting, timing oracles, HTTP headers, CSRF, errors, payloads, JWT, personal data in logs, and cookie handling.
- Developed CESAD commission creation/administration, stage assignment, and rollover when a commission loses validity, preserving previous assignments as auditable history.
- Modeled opinion supersession with one active opinion per stage and preserved prior acts; tested critical cases against real PostgreSQL and contributed documentation of temporal authority, document, and signature invariants.

**PSS Candidate Evaluation - SEDUC-PA** 2026

Next.js · NestJS · TypeScript · Google Sheets · DOCX

- Contributed validation, API hardening, Google Sheets integration, and server-side DOCX generation in a monorepo with shared contracts.

## Technical and security projects

**WAF-Proxy** 2026

Rust · Lua · Prometheus

- Built a defensive, educational reverse proxy with a restricted Lua rule API, bounded inspection, normalization, anomaly scoring, and allow/log/block decisions.
- Implemented bounded body reads, hop-by-hop header removal, safe transfer-metadata handling, fail-closed behavior, rate limiting, tracing, and Prometheus metrics.
- Documented architecture, a security model, and a threat model; established checks using cargo fmt, Clippy, tests, and cargo audit.

**GoSpector** 2026

Go · DNS · TCP · TLS · HTTP · GitHub Actions

- Created a modular defensive reconnaissance framework for authorized assets, accepting domains, hosts, IPs, files, and stdin without exploitation, persistence, or evasion.
- Implemented passive enumeration, concurrent DNS, bounded TCP scanning, service/TLS fingerprinting, HTTP probing, rate limiting, timeouts, and text/JSON/JSONL output.
- Configured CI with tests, the race detector, vet, linting, and builds, plus Linux, macOS, and Windows releases through GoReleaser.

**Desktop Password Vault** 2026

Rust · Tauri 2 · React · TypeScript

- Built an offline vault with envelope encryption: Argon2id derives the KEK and AES-256-GCM protects a random vault key that encrypts user data.
- Kept critical operations and keys in the Rust backend, outside the web layer, with memory zeroization, strict CSP, auto-lock, clipboard clearing, a CSPRNG generator, and optional key-file support.

**Cloud-Native Hardened IaC** 2026

AWS · Terraform · Kubernetes/EKS · DevSecOps

- Designed an AWS infrastructure lab with a segmented VPC, public/private subnets, EKS, S3 remote state, and DynamoDB locking.
- Applied least-privilege IAM, default-deny network policies, mTLS/cert-manager, resource limits, and CI-based IaC checks using tfsec and Checkov.

## Technical skills

**Programming languages:** Go, TypeScript/JavaScript, Rust, Python, Bash, SQL.

**Backend:** REST APIs, Chi, NestJS, Node.js, PostgreSQL, Prisma, JWT, RBAC, validation, concurrency, integrations.

**Frontend:** Next.js, React, Angular, typed forms, API integration, role/permission-aware interfaces.

**Security:** Secure coding, AppSec, threat modeling, HTTP hardening, rate limiting, CSRF, sessions, WAFs, cryptography, least privilege, network policies.

**Infrastructure/DevOps:** Linux, Docker, GitHub Actions, Railway, Vercel, AWS, Terraform, Kubernetes/EKS, CI/CD.

**Engineering:** Git/GitHub, pull requests, code review, technical documentation, ADRs, unit/integration tests, production maintenance.

**Spoken languages:** Portuguese (native) · English (B2).

## Education

**Estácio**, B.Sc. in Computer Science – Belém, Pará, Brazil 2024 - 2027 (expected)

## Additional training

- Cisco Networking Academy - Introduction to Cybersecurity (2026).
- Fortinet Training Institute - Introduction to the Threat Landscape 3.0 (2026).