Marko Kushta

Tirana, Albania | markokushta39@gmail.com | LinkedIn | +355 69 982 0388 | Driving License: Category B

EDUCATION

B.Sc. Computer Science (Dual Degree)
SUNY Empire State College & University of New York Tirana | 2020 – 2024
GPA: 3.51/4.0 (100% Scholarship), while working full-time from June 2023 – Relevant Courses: Data Structures, Algorithms, Database Systems, Software Engineering, Operating Systems, System Architecture, OOP

PROFESSIONAL EXPERIENCE

Software Engineer | /n software *January 2025 – Present (Remote, Tirana)*
IPWorks cross-platform networking and communications libraries spanning C#, TypeScript/Node.js, C++, Java and Rust. Small, autonomous team with direct access to senior engineers and product management.

IPWorks Networking Libraries (C# / TypeScript / C++ / Java)

- Ship fixes across the IPWorks libraries, including IPWorks IoT (MQTT, MQTTBroker), TCP, UDP, TLS, SMTP, FTP and SNMP components, testing on Windows, Linux and macOS; contribute to code reviews and technical discussions.
- Resolved TLS and connection-level failures (handshake hangs, mid-transfer disconnects, error propagation) using Wireshark, event logging and RFC research; designed a fail-fast fix for detecting bad socket state early and wrote a full technical writeup for the team lead.
- Provide frontline technical support to customers: reproduce and root-cause reported bugs, deliver fixes or workarounds, and turn protocol-level findings into actionable guidance.

Rust Editions of IPWorks IoT and IPWorks SNMP

- Sole engineer: ported both products from their Java editions by translating the library logic into Rust, and shipped them to customers.

DAV Drive (WebDAV product)

- Sole engineer on a product connecting Windows/Linux file systems to a WebDAV server: owned it end-to-end, from architectural design through production deployment, QA issue resolution, production support and all customer escalations.

Cross-Platform S/MIME Cryptography (C# / TypeScript / Java)

- Found and fixed a critical data-loss bug where the final encrypted block of a cryptographic stream was silently dropped, tracing buffering logic, encryption boundaries and the stream lifecycle across layers.
- Ported large security test suites across three platforms and documented cross-platform failures in technical writeups, separating platform limitations from code bugs to inform test-coverage decisions.

Node.js / V8 Native Interop and Code Generation (C++ / TypeScript / Python)

- Worked extensively in the custom C++/V8 bridge between native Windows code and the Node.js runtime (buffer handling, object lifetime, method dispatch); implemented type management in V8 and tested SSL, cryptography and compression components through NUnit tests driving the generated TypeScript.
- Optimized the internal Python code generation tooling that transpiles Java into TypeScript and C++ (method overloading, async patterns, abstract methods); the go-to person for developers facing issues with it, and fixed bugs in its C#-to-C++ wrappers.
- On own initiative, benchmarked Java virtual threads: JDK 21 gave a 49% thread / 35% memory reduction vs. 98% / 52% on JDK 25; presented the findings to product management, who limited the feature to JDK 24+.

Software Engineer | Greenleaf BSC *June 2023 – January 2025 (Tirana, Albania)*
Financial services – FX trading, custodian banking and multi-tenant SaaS platforms, built by small, autonomous teams in a flat structure working directly with bank stakeholders. Stack: .NET Framework / .NET Core, ASP.NET Web API, Entity Framework, PostgreSQL, RabbitMQ, Apache Pulsar, SignalR, Angular.

New Module for FX Trading System

- Built the latest service of a new FX trading module for a large bank (~8 microservices, hundreds of thousands of daily clients, each service with its own PostgreSQL database, communicating via RabbitMQ), with SignalR real-time notifications and .NET background services such as account limit alerts.
- Designed authentication across three identity providers (Keycloak/OIDC and the bank's proprietary protocols), via .NET HTTP message handlers and authorization filters to infer token origin and map bank roles.
- Owned version control, package upgrades and bank-controlled releases to test and production (Docker configuration via YAML); added health checks and logging per service.

Custodian Banking System (Treasury Bills and Bonds)

- Delivered major backend features end-to-end, from architectural design through production deployment, using event-driven architecture, state machines and dependency injection; extended the Stateless library to support multiple state machines per object.

- Built REST API endpoints (ASP.NET Web API) for the bank's internal systems, plus the email notification system and internal and external investment transfers.
- Used materialized views for investment balances and optimized transaction queries in PostgreSQL.

Multi-Tenant SaaS Platform

- Informally led architecture and backend development with three junior developers to upgrade an existing platform to multi-tenancy, coordinating with the Angular frontend to deliver complete features.

Additionally: screened candidates, ran technical interviews, mentored three junior developers through onboarding roadmaps, contributed to code reviews and documented project workflows.

PERSONAL PROJECTS

XMLUI – Open-Source UI Framework (TypeScript, SCSS, React patterns: RadixUI, TanStack)

- Outside daily tasks, acted as an internal reviewer: tested internal projects against the framework and sent its developers detailed issue reports with root cause and reproduction steps.
- Became the go-to person for developers wiring components to REST APIs or debugging custom components while documentation was still thin.

Crypto Trading Desk (.NET 9, React, PostgreSQL, Redis, SignalR, Docker) – github.com/absolut1on/crypto-trading-desk

- Built a real-time cryptocurrency trading desk as two .NET 9 minimal-API microservices: a Rates service streams live Binance WebSocket prices, caches them in Redis and publishes ticks to a Redis Stream; a Gateway service consumes the stream, pushes prices to a React/TypeScript client via SignalR and owns all business logic.
- Implemented market and limit orders with an order state machine (Pending, Validated, Executing, Filled), an event-sourced wallet ledger with a full audit trail, and portfolio positions with average entry price and live unrealized PnL.
- Secured the API with Supabase JWT authentication and ran the whole stack (PostgreSQL 16, Redis 7, Grafana monitoring) as six containers with Docker Compose.

Streaming XML Parser (C# / .NET 8 and C++)

- Wrote a forward-only XML parser from scratch as a character-level state machine (elements, attributes, text, comments), with element search by name, text value and attribute, plus optional element-path tracking; implemented it in both C# and C++ and timed it on a 116 MB XML file.

Exoplanet Hunting in Deep Space using Machine Learning – B.Sc. Thesis (Python)

- Built, trained and evaluated a machine learning model to detect exoplanets with scikit-learn, Pandas and SciPy.

SKILLS

Languages: C# / .NET (ASP.NET Web API, Entity Framework, LINQ, NUnit), TypeScript / Node.js, C++, Java, Rust, Python, SQL

Backend & Architecture: REST API design, microservices, event-driven architecture, multi-tenant SaaS, state machines, dependency injection, background services

Messaging & Real-time: RabbitMQ, Apache Pulsar, Redis Streams, SignalR

Data & Frontend: PostgreSQL (materialized views, query optimization), Entity Framework; Angular, CSS/SCSS

Security & Networking: OpenID Connect / OAuth 2.0, Keycloak, multi-provider authentication, TLS/SSL, certificates, S/MIME; TCP/IP, UDP, MQTT, SMTP, FTP, SNMP, WebDAV

Engineering Practice: unit and integration testing (NUnit), code review, debugging with Wireshark, logs and browser dev tools, Postman, technical documentation

Tools: Git, Azure DevOps, Jira, CI/CD, Docker / Docker Compose, Grafana, Visual Studio, VS Code; Windows, Linux, macOS

Spoken Languages: English (Fluent) | Albanian (Native)