# Israel López Maíz

Senior Software Engineer

Email: israellopezdeveloper@gmail.com
Phone: +995 591 311 854
Portfolio: https://israellopezdeveloper.github.io
LinkedIn: https://www.linkedin.com/in/israellopezmaiz/

## Summary

Senior Software Engineer with over 15 years of experience designing and building high-performance backend systems, distributed architectures, and production-grade infrastructure. Strong background in low-level systems, concurrency, and performance-critical code, combined with extensive experience in cloud-native platforms, Kubernetes, and GPU-accelerated workloads.

I have worked on large-scale distributed systems for AI model deployment, monitoring, and orchestration across cloud, on-prem, and edge environments. I value clean architecture, correctness, and measurable impact, and I am particularly interested in projects where performance, reliability, and scalability are first-class concerns.

## Senior Software Developer

Datura AI
March 2025 -

Context
Celium is a distributed platform for deploying and running AI models across cloud, on-prem, and edge environments. The system operates over heterogeneous GPU-enabled nodes with strict latency, availability, and cost constraints.
Action
• Designed and optimized the end-to-end model deployment lifecycle.
• Implemented orchestration, monitoring, and recovery modules in Python and Rust.
• Integrated C-based execution paths for latency-critical operations.
• Automated build, deployment, and rollback pipelines using Docker, Helm, and CI/CD.
• Tuned CPU/GPU scheduling, affinity, and resource allocation policies.
Metrics
• Reduced model time-to-production from hours to minutes.
• Improved sustained GPU utilization by ~20–30%.
• Lowered P95 inference latency by ~30–40% on critical paths.
• Reduced deployment-related incidents by ~40%.
Result
Enabled fast, reliable, and cost-efficient AI model deployment at scale. Improved platform stability and predictability while supporting decentralized execution and continued network growth.

### Project 1: Performance-based reward algorithm improvement

I led the optimization of the platform’s reward system, designed to incentivize nodes that consistently deliver the highest performance across the network. This involved a deep redesign of the scoring algorithm, introducing more representative performance parameters and recalibrating metric weights to better reward efficiency and stability.
The work included analyzing how nodes handled critical factors such as sustained uptime, inference latency, and the operational impact of hot model swaps. Based on this analysis, I introduced a new

scoring model that prioritizes continuous availability, progressively penalizes downtime or excessive model-switching times, and favors nodes providing predictable and stable service.
To improve fairness and robustness, I added safeguards against metric manipulation and opportunistic behavior, ensuring rewards are driven by execution quality rather than raw task volume. The algorithm was designed to remain flexible, allowing dynamic adjustment of metric weights as the network and strategic goals evolve.
As a result, the reward system now aligns much more closely with Celium's objectives: nodes that deliver high availability, low latency, and minimal transition times between models are rewarded proportionally, fostering collaboration, stability, and overall efficiency across the distributed infrastructure.
Introduced anti-manipulation mechanisms and dynamic weighting to maintain balance and fairness.

### Project 2: Model deployment and dynamic resource management

I implemented a platform for instant deployment of Machine Learning models in containers, running across on-premise, public cloud, and edge environments. The design prioritized portability, low latency, and scalability, enabling secure model publishing and updates with minimal service disruption.
The solution features dynamic resource management through real-time pod editing and hot reallocation of CPU, GPU, and memory to handle demand spikes efficiently. This significantly reduced time-to-production while maintaining system stability and an optimal cost-performance balance.
RESPONSIBILITIES
• Designed and implemented container-based infrastructure for multi-environment model deployment (local, cloud, and edge).
• Enabled real-time pod editing and hot reallocation of CPU, GPU, and memory without noticeable downtime.
• Automated CI/CD pipelines for model packaging, validation, and publishing with safe rollback mechanisms.
• Defined scaling, affinity, and anti-affinity policies for GPU-accelerated workloads.
• Implemented monitoring and alerting for latency, throughput, and resource usage (CPU/GPU/memory).
• Collaborated with Data Science and DevOps teams to shorten the training-to-production cycle.
• Ensured cross-environment model portability by standardizing images, dependencies, and inference contracts.

### Project 3: Automation and scalability

I developed automated model update and distribution pipelines across hundreds of remote nodes, ensuring version consistency while minimizing manual intervention.
This approach dramatically reduced propagation times for new releases and strengthened overall system resilience, ensuring continuity in large-scale distributed environments.
RESPONSIBILITIES
• Designed automation workflows for model updates and deployments in distributed systems.
• Implemented reliable distribution mechanisms with strict version control across hundreds of remote nodes.
• Reduced rollout times through optimized CI/CD pipelines.
• Monitored integrity and consistency of deployed models across heterogeneous clusters.
• Defined rollback and recovery strategies to ensure resilience during distribution failures.
• Worked with infrastructure teams to improve scalability and system availability.

### Project 4: Monitoring and performance optimization

I integrated real-time performance and resource usage metrics, significantly increasing system observability and improving early detection and resolution of incidents across the platform.
I applied advanced load-balancing and process affinity techniques to optimize energy efficiency and fully leverage available hardware in distributed environments.
RESPONSIBILITIES
• Designed and implemented real-time metrics collection for CPU, GPU, memory, and inference latency.
• Built dashboards and proactive alerts to provide full platform observability.

• Applied dynamic load-balancing strategies to maintain stability under high demand.
• Configured process affinity and anti-affinity policies to optimize hardware utilization.
• Analyzed usage patterns to continuously improve energy efficiency.
• Collaborated with operations teams to define and enforce performance KPIs.

### Project 5: Interoperability with decentralized ecosystems

I contributed to the integration of Celium with decentralized networks such as Bittensor, expanding the platform's capabilities and improving interoperability across collaborative distributed ecosystems.
This integration enabled the use of decentralized infrastructure for resource sharing and model deployment, strengthening scalability and ecosystem resilience.
RESPONSIBILITIES
• Designed and implemented connectors between Celium and decentralized networks such as Bittensor.
• Ensured protocol and data format compatibility across heterogeneous systems.
• Developed secure authentication and validation mechanisms for cross-network interactions.
• Optimized task and workload distribution using decentralized resources.
• Collaborated with research teams to explore new interoperability use cases in decentralized AI.
• Contributed to documentation and standardization of multi-ecosystem integration processes.

### Technologies used:

- Prometheus
- CUDA
- Git
- Docker
- Python
- PostgreSQL
- TensorRT
- Terraform
- Bash
- Logstash
- cAdvisor
- Substrate API
- PyTorch
- Bittensor
- GPU Computing
- Helm
- C
- FastAPI
- Celery
- gRPC
- Kafka
- Elastic Stack
- Kibana
- Nginx
- Web3
- Tech
- GitHub Actions
- Kubernetes
- Rust
- Redis
- Grafana
- Ansible
- MinIO
- Elastic Search
- Node Exporter
- Polkadot.js
- ONNX Runtime

---

## Senior Software Developer

Indra
April 2024 - February 2025

Context
Mission-critical defense platform developed under the European Defence Fund. The system targets next-generation military vehicles and must operate under strict availability, security, and interoperability constraints across NATO environments.
Action
• Designed and implemented backend services in Java and Node.js using Clean Architecture.
• Integrated a RAG system for low-latency contextual retrieval of operational intelligence.
• Developed GIS components for real-time geospatial analysis and tactical visualization.
• Defined secure APIs and interfaces to interoperate with NATO-compliant systems.
• Automated CI/CD pipelines and Kubernetes deployments with full observability.
Metrics
• Reduced operational data retrieval time from seconds to sub-second latency.
• Improved service availability to >99.9% in simulated hostile environments.
• Decreased deployment and integration errors by ~35% via automated CI/CD.

• Validated platform stability under sustained load and fault-injection scenarios.
Result
Delivered a scalable and interoperable mission platform aligned with NATO standards. Improved situational awareness, system resilience, and decision-making speed in complex operational scenarios.

### Project 1: Next-generation mission system for military vehicles

A strategic project led by Indra under the European Defence Fund to build the first next-generation mission system for military vehicles in Europe and NATO countries. The system integrates Artificial Intelligence, GIS, and sensor management to provide armed forces with a tactical advantage in complex operational environments.
The architecture is open, modular, and scalable, enabling interoperability between allied forces and rapid integration of new technological components. It includes a RAG system for contextual information retrieval, real-time geospatial analysis, and secure communication modules for critical data exchange.
RESPONSIBILITIES
• Designed and implemented backend microservices in Java and Node.js following Clean Architecture principles.
• Developed and integrated a RAG system for semantic search and contextualization of operational data.
• Implemented GIS features for geospatial analysis and tactical visualization on interactive maps.
• Defined secure APIs and protocols to interoperate with other NATO systems.
• Automated CI/CD pipelines in GitLab with deployments to Kubernetes.
• Optimized performance through load testing and fault-tolerance validation.

### Project 2: Retrieval-Augmented Generation (RAG) systems integration

I actively contributed to the integration of Retrieval-Augmented Generation (RAG) systems to enhance the mission platform's ability to access critical information in real time. My work focused on designing and implementing pipelines to ingest, process, and query data from multiple heterogeneous sources, ensuring that retrieved information remained relevant, accurate, and low-latency.
In addition, I adapted the system to scale efficiently in distributed and resource-constrained environments by optimizing semantic queries, integrating embedding vectors, and applying hybrid search techniques (full-text and semantic). This work was instrumental in providing mission personnel with timely and actionable information in dynamic, high-complexity operational scenarios.

### Project 3: Artificial Intelligence modules collaboration

I was part of the team responsible for designing and integrating Artificial Intelligence modules into the platform, aimed at improving analysis and decision-making in mission-critical scenarios. I collaborated on services capable of real-time data processing, including satellite imagery analysis, video pattern recognition, and natural language processing for automated extraction of relevant information.
My contributions included implementing training and deployment pipelines, converting models to GPU-optimized formats for resource-constrained environments, and integrating APIs that connected AI modules with the rest of the platform ecosystem. This work took place in a highly multidisciplinary environment, aligning advanced AI capabilities with NATO interoperability requirements.

### Project 4: Integration testing and deployment in distributed environments

I led and participated in complex integration testing efforts, ensuring that system modules communicated reliably in distributed, mission-critical environments. This included defining test scenarios, building continuous integration pipelines, and validating interoperability across services deployed on both on-premise and cloud infrastructures.
I also contributed to the design and execution of deployments using container orchestration and automation tools to ensure platform scalability, resilience, and security. Real-time monitoring, event tracing, and dashboards were implemented to enable rapid incident detection and resolution. This effort was key to delivering a robust system capable of operating under demanding conditions with high availability requirements.

**Technologies used:**

- Java
- Spring Boot
- Spring Data
- Spring Security
- Node.js
- Express
- Python
- FastAPI
- RAG
- Vector DB
- Embeddings
- PostgreSQL
- MongoDB
- GIS
- PostGIS
- OpenLayers
- Docker
- Kubernetes
- Helm
- GitHub Actions
- Jira
- Prometheus
- Grafana
- Node Exporter
- Elastic Search
- Qdrant
- LangChain
- PyTorch
- Bittensor
- Hugging Face Transformers
- OpenCV
- CUDA
- ONNX Runtime
- Jenkins
- Git
- Ansible
- Terraform

---

## Senior Software Developer

Devo
March 2020 - March 2024

Context
Cloud-native security analytics platform ingesting massive volumes of sensitive logs. The ingestion layer must handle bursty traffic, strict reliability targets, and compliance constraints (FedRAMP/FIPS/NIST).
Action
• Owned and optimized the ingestion load balancer for sustained high-throughput production traffic.
• Improved fault tolerance, recovery paths, and observability across ingestion services.
• Implemented reliability features (including RELP support) to reduce event loss under failure scenarios.
• Built multi-region / multi-cloud redundancy mechanisms to improve disaster recovery posture.
• Integrated quality and security controls into CI/CD to support FedRAMP readiness.
Metrics
• Sustained ~60 TB/day ingestion under production load.
• Reduced event loss and improved delivery guarantees during network/service failures.
• Lowered incident frequency and improved detection time via stronger telemetry and alerting.
• Improved recovery time for ingestion components through hardened failover and runbooks.
Result
Delivered a faster and more reliable ingestion layer for mission-critical security data. Increased operational resilience and strengthened the compliance path while keeping performance stable at scale.

### Project 1: Improve the existing load balancer

I significantly improved Devo's event load balancer performance by introducing advanced strategies to increase fault tolerance and strengthen observability across ingestion services. I analyzed system requirements, identified bottlenecks, and ensured the balancer handled highly variable traffic more efficiently. I improved failure-recovery mechanisms to maintain high availability under unexpected faults. I also implemented RELP (Reliable Event Logging Protocol) support from scratch to ensure reliable, low-loss event delivery.
In addition, I designed client-aware load balancing to keep customer data locality where possible, enabling more compact queries and faster retrieval. By avoiding unnecessary data mixing across nodes, query paths became shorter and the system's operational overhead was reduced. These changes delivered a more robust, responsive load balancing layer and improved the overall ingestion and query experience.

## Project 2: Build a multi-cloud backup system

I led the design and development of a next-generation multi-cloud backup system from scratch, a critical upgrade to Devo's redundancy and disaster-recovery capabilities. The system stored customer logs across multiple regions and cloud environments to maximize data protection and availability. A key feature was real-time configuration of storage tiers, allowing seamless changes to cloud storage settings based on live operational needs.

I implemented automated data transition rules to move data across storage classes based on access patterns and retention policies, ensuring cost-efficiency without compromising accessibility or performance.

The system also included robust node-recovery mechanisms to restore service quickly after node failures, reducing downtime and protecting data integrity. To improve efficiency, I implemented advanced compaction strategies for data approaching end-of-life, optimizing storage utilization and reducing overhead.

Finally, I integrated precise cleanup schedules so obsolete data was removed on time, keeping storage health and performance stable. This project reinforced Devo's data resilience and supported strict security and retention requirements.

## Project 3: Integrate quality and security checks into CI/CD

I played a key role in embedding comprehensive quality and security checks into Devo's CI/CD pipeline, helping the engineering process meet strict industry standards including NIST, FIPS, and FedRAMP.

Jenkins pipeline integration

I introduced SonarQube into the Jenkins pipeline for continuous code-quality analysis, enabling early detection of defects, code smells, and potential vulnerabilities and ensuring only high-quality changes reached production.

Migration to GitLab CI/CD

After stabilizing the approach in Jenkins, I migrated the pipeline to GitLab CI/CD to streamline workflows and improve automation and collaboration.

Security vulnerability scanning

I integrated Snyk and OWASP tooling into CI/CD. Snyk provided real-time dependency vulnerability scanning and remediation guidance, while OWASP tooling supported broader security assessments focused on web application risks.

Compliance outcomes

I implemented automated gates to ensure every change passed rigorous quality and security checks before deployment. This strengthened the platform's security posture and reduced the friction of compliance verification, supporting Devo's FedRAMP readiness.

## Project 4: Build an alerting system with Prometheus

I built a comprehensive alerting system to monitor performance and critical situations in production using Prometheus. The goal was real-time visibility and rapid response for issues such as attack indicators, fraud patterns, and operational anomalies.

I integrated Prometheus into Devo's infrastructure, defined core KPIs and thresholds, and implemented alerting rules to detect abnormal patterns in ingestion throughput, processing latency, and resource usage.

To reduce incident response time, I created a detailed troubleshooting playbook with step-by-step diagnostics, recommended actions, and escalation paths. I also implemented early automated responses (e.g., service restarts, load redistribution, and dynamic scaling) to mitigate incidents before they escalated, reducing MTTR and minimizing user impact.

Finally, I helped define incident-handling protocols (roles, communication, coordination) for high-severity scenarios, improving response consistency and platform reliability.

## Technologies used:

**Technologies used:**

- BDD
- TDD
- Jira
- GIT
- Kubernetes
- Gitlab CI/CD
- Jenkins
- Owasp
- Snyk
- SonarQube
- FedRAMP
- FIPS
- C++
- C
- Node.js
- GCP
- AWS
- Java
- NIST
- Ansible
- Bash
- Python
- Kibana
- Elastic Stack
- Docker
- Grafana
- Prometheus

---

## Senior Software Developer

European Commission
June 2017 - January 2020

Context
Public-sector, mission-critical systems within DG COMP, supporting antitrust procedures and legal workflows across the EU. Systems operate under strict security, auditability, and availability requirements.
Action
• Led architecture definition and end-to-end delivery across multiple projects.
• Captured and enforced non-functional requirements (security, performance, auditability).
• Designed RESTful APIs (Richardson Level 3) for secure document exchange.
• Defined CI/CD, QA gates, and testing strategies across heterogeneous stacks.
• Drove technology selection and cross-team integration.
Metrics
• Delivered 3 large-scale systems on schedule under EU regulatory constraints.
• Reduced integration issues through early dependency and interface definition.
• Improved delivery predictability via automated CI/CD and test gating.
• Maintained zero critical security incidents in audited environments.
Result
Delivered secure, scalable platforms supporting EU competition policy enforcement. Improved reliability, auditability, and long-term maintainability of DG COMP systems.

### Project 1: Definition of two RESTful APIs

The first project involved defining two RESTful APIs following the Richardson Maturity Model Level 3, including HATEOAS. These APIs served as secure platforms for exchanging legal documents in antitrust procedures.
• Internal API: Designed for European Commission staff.
• External API: Designed for external parties.
Both APIs enforced access control via a Single Sign-On system (CAS). Data persistence used Oracle DB and MongoDB. The APIs also integrated with an internal secure document storage system via a SOAP interface. Each procedure workflow was modeled using the State pattern, and system administrators were notified of state transitions via email.

### Project 2: E-Confidentiality

In the second project, I served as Software Architect overseeing the delivery of a system to negotiate confidential versions of legal documents in antitrust procedures. I defined the approach for creating and assigning dynamic review groups responsible for managing online legal workflows. These groups (internal staff) were authenticated via the Central Authentication Service (CAS), a system previously implemented under my direction.

I designed the document state architecture using the Status/State pattern with support for change reversal, enabling safe rollbacks and preserving integrity throughout the negotiation process.
I selected and integrated Oracle DB and MongoDB to cover the project's persistence needs, and ensured the API design aligned with Richardson Level 3 principles including HATEOAS. The result was a secure, scalable, and user-friendly platform for handling sensitive legal documentation, strengthening DG COMP's operational effectiveness.

### Project 3: Semantic index for legal documents

The third project, proposed by me, focused on building a semantic indexing system for legal documents. I acted as the project architect and stayed hands-on in implementation to ensure delivery. The solution applied clustering techniques (K-Means) to DG COMP's antitrust legal document dataset. Indexing ran in a dedicated microservice built with gRPC and implemented in C++14 for maximum efficiency and performance.
The microservice architecture followed CQRS and was deployed as a service pool to ensure resilience and fast response times. Semantic search was powered by a neural network trained on the legal document corpus using TensorFlow. I also built a complementary Java service that consumed the semantic search capability to enhance downstream functionality.
This dual-service approach enabled high-speed semantic search and robust, scalable indexing of complex legal documentation. The system emphasized resilience, efficiency, and precision, making it a key capability for legal document management and retrieval.

### Project 4: Renewable energy distribution system development

The fourth project involved building a renewable energy distribution system aimed at normalizing and stabilizing electricity consumption over time to maximize renewable energy utilization. I contributed significantly across architecture and implementation.
The solution integrated a wide range of technologies. At the core, a Kafka-based messaging cluster handled device communication using Avro schemas. An intermediate gateway, implemented on Raspberry Pi devices, sent telemetry and received commands. The gateway software was written in Go and supported ModBus (serial and TCP), with TCP secured via VPN.
Device software lifecycle management was handled with BalenaIO, enabling reliable deployment and operations across distributed gateways. CI/CD was managed with CircleCI to ensure efficient delivery.
I also built a Groovy-based statistics analyzer for telemetry ingestion, generating operational insights on energy consumption patterns and system performance. I helped drive architecture, implementation, and integration across key components to ensure a reliable end-to-end system.

### Technologies used:

**Technologies used:**

- BDD
- Bamboo
- CAS
- Selenium
- MongoDB
- Eclipselink
- Spring Documentation
- Spring Security
- Enterprise Architect
- gRPC
- OpenCL
- Google Test
- C++
- Avro Schemas
- Azure
- Make
- RaspberryPI
- Go
- TDD
- JUnit
- GIT
- AngularJS
- Oracle DB
- JPA
- Spring Data
- Java
- Docker
- Valgrind
- Microservices
- Cassandra
- Kibana
- Kafka
- GCP
- Python
- BalenaIO
- SonarQube
- Jira
- Maven
- Swagger
- WebLogic 12
- Spring IoC
- Spring MVC
- IntelliJ IDEA
- Ansible
- TensorFlow
- Groovy
- CUDA
- Elastic Search
- IoTHub
- Kubernetes
- Modbus
- Unit Test

---

## Senior Software Developer

Panel Sistemas
September 2015 - June 2017

Context
Multi-project delivery in enterprise environments across transport, energy, and security sectors. Systems operated under high-availability, data integrity, and long-term maintainability constraints.
Action
• Started as Functional Analyst defining NFRs, scope, and delivery plans.
• Assumed Senior Developer role during execution, owning architecture and implementation.
• Defined project structure, CI/CD pipelines, and testing standards.
• Applied TDD and BDD to reduce regressions and stabilize delivery.
• Delivered internal technical training to raise team-level engineering quality.
Metrics
• Delivered 3 complex, multi-year projects across different industries.
• Reduced post-release defects through enforced unit and integration testing.
• Improved delivery predictability via task decomposition and estimation discipline.
• Increased team autonomy and code quality through shared engineering practices.
Result
Successfully transitioned projects from analysis to stable production systems. Improved code quality, delivery reliability, and team maturity across heterogeneous, high-impact enterprise platforms.

### Project 1: Modernizing maritime and air transport messaging: replacing IATA with XML/JSON

The first project delivered a modern messaging exchange platform for maritime and air transport. It successfully replaced an outdated IATA standard based on plain text and positional fields (highly error-prone) with a more robust, readable XML/JSON-based format.
I defined a parser that translated legacy messages into the new standards through a RESTful API aligned with Richardson Maturity Model Level 3. Beyond translation, the API enabled interoperability with external platforms worldwide and persisted all exchanges to ensure end-to-end traceability of global parcel transit. Given high availability requirements, we implemented a replicated, sharded

MongoDB architecture across a server cluster.

### Project 2: Iberdrola smart terminal management with Siemens

The second project was a joint initiative with Siemens to build Iberdrola's smart terminal management infrastructure. Terminals reported customer electricity consumption daily via SNMP, with optional high-resolution monitoring down to per-minute usage for selected devices. Data was stored in a distributed MongoDB setup (multiple clusters, load balancing, and replicated sharding). Data placement followed geographic locality when capacity allowed, reducing query scope and improving performance. We retained one year of per-minute data, then compacted it into daily min/max/avg for five years, followed by additional compaction thereafter. The system also supported real-time control and monitoring of devices via SNMP and JNLP. This platform was built using C++ microservices with gRPC and deployed across a fleet of Red Hat Enterprise Linux servers.

### Project 3: Monitoring and assisted-care system for elderly and disabled users (Securitas Direct)

The third project, delivered for Securitas Direct, involved building an end-to-end monitoring and assisted-care system for elderly users and people with disabilities. The system inferred user status using in-home sensors (presence, door opening, and personal fall detectors) and produced a detailed activity timeline. Beyond storage and querying, it generated alerts by learning typical behavioral patterns and detecting sudden deviations, with configurable alarm policies.
The solution also included a hybrid mobile app that leveraged device sensors to detect falls, track GPS location, and support additional safety-related signals.

### Technologies used:

- Ansible
- Redmine
- Maven
- Phonegap
- Primefaces
- RabbitMQ
- Hibernate
- Spring Data
- Java
- JNLP
- Microservices
- cLion
- C++
- Typescript
- Spring Batch
- Docker
- Jenkins
- Selenium
- AWS
- JSF
- MongoDB
- JPA
- Spring MVC
- Spring IOT
- Jira
- gRPC
- Valgrind
- Confluence
- Ionic
- Spring IoC
- IntelliJ IDEA
- GIT
- Hadoop
- GWT
- JUnit
- Wildfly/JBoss
- Spring Documentation
- Spring Security
- SNMP
- Apache Tomcat
- Boost
- Google Test
- Jasmine
- Oracle DB

---

## Senior Software Developer

Grupo PRISA
March 2012 - May 2015

Context
Large-scale digital transformation initiatives within a global media and education group. Projects were delivered fully remotely, involving distributed teams across multiple continents and time zones.
Action
• Led technical execution across three parallel projects with distributed teams.
• Worked directly with Product Owners to translate business goals into deliverable technical plans.
• Coordinated teams across South America, India, and China, resolving communication and

dependency issues.
• Acted as technical liaison, translating and clarifying requirements for Chinese-speaking teams.
• Improved delivery workflows through automation and process optimization.
Metrics
• Delivered 3 production systems with globally distributed teams.
• Maintained predictable sprint delivery across multiple time zones.
• Reduced delivery friction by minimizing rework caused by requirement misalignment.
• Improved team throughput via tooling and workflow optimizations.
Result
Successfully delivered complex digital platforms in a fully remote, multicultural environment. Improved cross-team alignment, delivery predictability, and execution efficiency across global teams.

### Project 1: Educational content viewer mobile application

This project involved building a cross-platform mobile application (Android and iOS) for consuming educational content packages. The app was optimized to run smoothly on low-end devices, ensuring broad accessibility without sacrificing user experience.
It supported SCORM-compliant learning packages, tracked user progress, and reported learning metrics back to the server. Backward compatibility with legacy content ensured long-term usability across existing educational materials.

### Project 2: Educational package standardization and Flash-to-HTML5 migration

This project focused on standardizing educational content packages by building a Flash-to-HTML5 translation tool compatible with the SCORM standard. The goal was to avoid costly manual migrations while preserving existing content.
A centralized shared library ensured that updates propagated automatically to all previously generated packages. New versions were thoroughly validated in virtualized environments (QEmu, KVM, Xen) before release, ensuring stability, consistency, and long-term maintainability.

### Project 3: Integrated education management platform

The third project was built on the .NET platform and delivered a comprehensive digital environment for teachers and education professionals. It enabled student management, educational content administration, and result analysis.
The platform also integrated a high-resolution IP-based audio and video communication system, supporting real-time interaction and media streaming. This unified solution streamlined educational workflows and improved collaboration between educators and learners.

### Technologies used:

- Confluence
- GIT
- Jira
- jQuery
- Bootstrap
- JavaScript
- HTML
- SCORM
- Dagger
- Objective-C
- XCode
- Android Studio
- Java
- Valgrind
- NginX
- Bash
- Xen
- KVM
- QEmu
- AngularJS
- Flash
- Linux
- C++
- Microsoft SQL Server
- Windows Server
- IIS
- .NET
- C#

# Senior Software Developer

Intelygenz
January 2010 - January 2012

Context
Client-facing and platform-level systems delivered under strict performance, reliability, and integration constraints across media and banking domains.
Action
• Owned technical design and implementation across three concurrent projects.
• Delivered low-level, performance-critical components alongside end-user applications.
• Collaborated directly with clients and product teams to align requirements and execution.
• Balanced performance, usability, and maintainability in heterogeneous stacks.
Metrics
• Delivered 3 production systems for large enterprise clients.
• Met strict latency and resource constraints on mobile and backend systems.
• Reduced integration friction through clear ownership and direct client coordination.
• Stabilized delivery across diverse technologies and platforms.
Result
Successfully delivered performant and reliable systems for high-visibility clients. Improved execution speed, system robustness, and end-user experience across multiple production platforms.

### Project 1: Antena 3 Television interactive Android application

This project involved building a native Android application for Antena 3 Television that acted as a real-time interactive companion to live TV content. The app synchronized dynamically with broadcasts using inaudible audio markers embedded in the signal.
I implemented a custom high-performance solution using the Android NDK to capture and process audio with minimal battery impact. The system encoded binary data through precisely timed audio echoes, enabling seamless interaction between the broadcast and the mobile application.

### Project 2: BBVA document management system

The second project was a document management system for BBVA, implemented in C# on the .NET platform. The system integrated with external services built on Google App Engine and was tailored for secure handling of financial documents.
It supported editing, review, and accreditation workflows, embedding BBVA's business logic and approval processes. The solution improved cross-team collaboration and ensured alignment with banking standards and regulatory requirements.

### Project 3: Collaborative frontend platform for BBVA

The third project focused on building a collaborative frontend for BBVA, developed in Java and JavaScript and integrated with the NACAR platform. This environment enabled advanced interaction patterns within banking applications.
A key outcome was the creation of IRIS, a reusable gesture and on-screen interaction library that significantly improved usability and responsiveness. This library became a core component for enhancing user experience across the platform.

### Technologies used:

**Technologies used:**

- Confluence
- Android NDK
- jQuery
- HTML
- Windows Server
- C#
- JSF
- Websphere
- Spring Documentation
- Spring Security
- GIT
- Android Studio
- Bootstrap
- Cassandra
- IIS
- Eclipse
- JUnit
- Hibernate
- Spring Data
- Jira
- Java
- JavaScript
- Microsoft SQL Server
- .NET
- Maven
- Oracle DB
- JPA
- Spring MVC

---

## Senior Software Developer

University of Alcalá (UAH)
January 2006 - January 2009

Context
Applied research within a university environment, focused on AI, accessibility, computer vision, and biometric systems. Work combined research, prototyping, teaching, and cross-university collaboration.
Action
• Conducted applied research on AI algorithms, computer vision, and accessibility technologies.
• Designed and prototyped experimental systems and assistive devices.
• Collaborated with external research groups on interdisciplinary projects.
• Authored technical reports and contributed to public R&D grant proposals.
• Supported teaching activities in AI and Computer Vision courses.
Metrics
• Delivered multiple functional research prototypes across AI and accessibility domains.
• Contributed to funded and submitted public research proposals.
• Supported university-level courses as teaching assistant.
• Participated in competitive robotics and applied research initiatives.
Result
Produced practical research outcomes and transferable knowledge in AI and accessibility. Strengthened collaboration between academia and applied technology while contributing to education and research capacity.

### Project 1: Accessible interaction devices research

This project focused on researching and prototyping assistive devices to improve interaction for users with visual, auditory, or motor disabilities. The goal was to design inclusive input mechanisms and adaptive interfaces for desktop, mobile, and embedded systems.
User-centered testing and direct collaboration with people with disabilities ensured practical, usable, and impactful solutions that significantly improved accessibility and user experience.

### Project 2: Standardized communication system for biometric devices

I contributed to the design of a standardized communication system for biometric devices from multiple vendors. The solution embedded an RPC layer directly into devices, enabling transparent, language-agnostic, and OS-independent communication.
The system addressed interoperability, security, and robustness requirements, enabling encrypted and authenticated data exchange for sensitive biometric information.

### Project 3: Research group website platform

I designed, built, and maintained the research group's public website using Joomla CMS. The site served as a central hub for publications, project updates, and events.
My work focused on performance optimization, accessibility compliance, responsive design, and ongoing maintenance, ensuring scalability, security, and ease of content management for non-technical contributors.

### Project 4: Research, teaching, and robotics activities

I supported multiple research activities, organized technical seminars on Robotics, and served as a teaching assistant for AI and Computer Vision courses.
I also participated in the Hispabot robotics competition (maze-solving category), applying AI and robotics techniques in a competitive, real-world setting.

#### Technologies used:

- Mono
- .NET
- C#
- SVN
- Arduino
- PIC
- Qt
- Anjuta
- C++
- Linux
- Android
- Java
- Joomla
- PHP
- HTML
- CSS
- JavaScript
- Python
- Octave
- MATLAB
- C

---

## Senior Software Developer

Knowcentury
January 2005 - July 2005

Context
Small web agency delivering CMS-based websites for local clients, with tight timelines and direct customer interaction.
Action
• Built and customized client websites using PHP-based CMS platforms.
• Implemented layouts, content structures, and basic dynamic features.
• Worked directly with client requirements and feedback.
Metrics
• Delivered multiple client websites within short delivery cycles.
• Ensured functional correctness and basic performance for production use.
• Gained early production experience with real customer constraints.
Result
Delivered functional websites for real clients while building a strong practical foundation in web development and professional software delivery.

### Project 1: Website development

Built and customized websites using popular CMS platforms, focusing on clean structure, basic performance, and client-specific requirements.

#### Technologies used:

- HTML
- MySQL
- JavaScript
- PHP
- Joomla

---

## Senior Software Developer

IBM
September 2003 - December 2004

Context
Applied research on enterprise-grade virtualization using IBM System z/390 mainframes. The goal was to evaluate feasibility, performance, and security for remote and consolidated workloads.
Action
• Researched and validated QEMU-based virtualization on IBM System z/390.
• Designed and deployed virtual Linux environments on SUSE Linux Enterprise.
• Analyzed CPU, memory, and I/O allocation strategies under virtualized workloads.
• Evaluated security, isolation, and stability for enterprise usage scenarios.
Metrics
• Successfully ran multiple isolated Linux instances on mainframe hardware.
• Demonstrated stable long-running virtualized environments.
• Validated predictable performance under consolidated workloads.
• Confirmed suitability for secure remote system deployment.
Result
Validated mainframe-based virtualization as a viable, secure, and scalable approach for consolidating enterprise workloads and enabling remote systems.

### Project 1: Mainframe-based system virtualization research

Research project focused on virtualizing Linux environments on IBM System z/390 using QEMU. The goal was to enable secure, high-performance remote systems by consolidating workloads on a mainframe running SUSE Linux Enterprise.

### Technologies used:

- IBM System z
- Linux
- QEMU
- Virtualization
- SUSE Linux Enterprise
- Enterprise Systems

## Training

**University of Alcalá de Henares**
Master's Degree in Artificial Intelligence for Information and Communication Technologies
September 2007 - June 2009

Artificial intelligence and machine learning: supervised, unsupervised, and reinforcement learning.

- Symbolic learning.
- Classification and regression models.
- Model optimization.
- Deep learning: multilayer networks, backpropagation, loss functions, hyperparameters, and training strategies.
- Convolutional neural networks for image recognition. Sequential and recurrent networks (LSTM).
- Parallelization and GPU-based computing.
- Vectorization techniques.
- Programming with TensorFlow and Theano.
- Scalable automated learning.

- Cluster parallelization frameworks.
- Applications in medicine, finance, autonomous driving, and more.

**University of Alcalá de Henares**
Technical Degree in Computer Engineering
September 2001 - June 2007

Graduated with honors in: Artificial Intelligence, Fuzzy Logic, Computer Vision Systems, Physics II, Mathematical Analysis, Data Structures, and Networks.

## Complementary training

**Linux Foundation**
LPIC - 2
September 2015 - February 2016

Official Linux system administration certification covering:

- advanced Linux administration
- kernel-related tasks
- advanced storage and filesystems
- network authentication
- system security (firewalls, VPNs)
- installation
- configuration of core network services (DHCP, DNS, SSH, web, file, and mail servers)
- monitoring
- automation
- infrastructure advisory.

**Linux Foundation**
LPIC - 1
September 2013 - February 2014

Official Linux system administration certification covering:

- Linux installation (including X11)
- command-line usage
- file and permission management (ACLs)
- basic system security
- routine maintenance tasks.

**University of Alcalá**
Design and Evaluation of Digital Educational Content
September 2007 - October 2007

Design of digital educational content using the SCORM standard.

**University of Alcalá**
Computer Vision Course
September 2007 - October 2007

Computer vision systems, convolution matrices, and filtering techniques.

**IBM**
Introduction to Deep Learning & Neural Networks with Keras
December 2025 - December 2025

This course provides a comprehensive foundation in deep learning and artificial neural networks, teaching how to build and evaluate models for regression and classification using the Keras library.

You have also gained expertise in designing advanced architectures like CNNs, RNNs, and transformers to solve complex real-world challenges such as image analysis and natural language processing.

**IBM**
Machine Learning with Python
December 2025 - December 2025

This program covers the complete machine learning workflow, from applying supervised and unsupervised learning algorithms using Python and scikit-learn to building advanced deep learning models with Keras.

I have developed the expertise to design and evaluate complex neural architectures, including CNNs, RNNs, and Transformers, to solve real-world challenges in image classification, natural language processing, and predictive modeling.

## Languages

| | Spoken | Written | Read |
|---|---|---|---|
| **Spanish** | Native | Native | Native |
| **English** | Fluent | Fluent | Fluent |
| **Chinese** | Fluent | Fluent | Fluent |