SENIOR SOFTWARE ENGINEER · BACKEND · C++ · PYTHON · AI · BUILDING PRODUCTS END-TO-END

Jesús Izquierdo



Senior Software Engineer with 11+ years building **scalable backends** and **distributed systems**. Deep expertise in **Python** backend services, AI integration, high-performance systems and C++. **Track record of owning end-to-end product stacks. 4+ years 100% remotely** for international teams across Europe and the US.

hello@jesusizquierdo.dev · jesusizquierdo.dev · linkedin.com/in/jesus-izquierdo-cubas · github.com/jesusizq

Madrid, Spain · EU citizen

EDUCATION	MSc & BSc in Industrial and Electronics Engineering, Universidad de Alcalá (2011–2017)
LANGUAGES	Spanish (native) · English (professional)

Core expertise

BACKEND	Python (Flask) · TypeScript (Node.js) · REST APIs · PostgreSQL · MongoDB · WebSockets · Async processing (Celery, RabbitMQ)
AI	Claude Code & AI agents · LLMs · RAG
SYSTEMS & PERFORMANCE	C++17/20 · Multithreading · HPC · Distributed Systems
INFRASTRUCTURE	Docker · CI/CD · Nginx · Google Cloud · Microservices

Experience

BSH Home Appliances (via Scalian) — Tech Lead

Oct 2026 – Jan 2027

Next-generation software platform for BSH induction hobs.

- **Team lead** — four engineers; planning, the Jira backlog and delivery of features
- **Requirements to tasks** — client cards and acceptance tests → tasks with acceptance criteria
- **Production C++17** — features across containerised Linux services

C++17 · CMake + Conan · Linux containers (Pantavisor/LXC) · Google Test/Mock · Jenkins CI

Omi — Senior Software Engineer

Aug 2025 – Oct 2026

PARIS, FRANCE · REMOTE

SaaS platform for photorealistic 3D model generation and product visualization.

- **C++ asset processing pipelines** — mesh optimization, texture conversion, format validation
- **Python integration layer** — custom AI models (Hugging Face) bridged to C++; Blender API scripting
- **AI outputs into C++** — custom bindings close the loop between inference and production assets
- **Full-stack performance** — Python pipelines, C++/Qt tools, rendering backend, OpenGL webviews
- **Artist tooling** — workflows automated in Python; C++/Qt tools improved; CMake/QMake modernized

SEDDI — Senior Software Engineer

May 2022 – Aug 2025

NEW YORK, US · REMOTE

Fashion-tech SaaS for real-time 3D fabric simulation in the browser; half Python backend, half C++ HPC.

- **Backend microservices and async jobs** — Python/Flask, PostgreSQL, RabbitMQ queue workflows
- **API layer** — consumed directly by the frontend; distributes work across microservices
- **C++ HPC services** — real-time 3D simulation with sub-second response times
- **Distributed pipeline bottlenecks** — found and fixed; early CUDA evaluation for parallel tasks
- **Docker CI/CD pipelines** — shared C++ libraries and internal tooling that standardized workflows

- **Code quality** — refactoring, Google Test and Pytest coverage, SOLID across C++ and Python

Grupo Epelsa — Software Engineer → Senior Software Engineer

Oct 2015 – May 2022

ALCALÁ DE HENARES, MADRID

Industrial weighing systems for retail. **Built the entire software product** from scratch with one other engineer, through to production.

- **Complete software stack** — embedded Linux, C++, Python: weighing, thermal/label printers, GUI
- **Custom TCP/IP protocol** — real-time sync between embedded devices on each store network
- **MariaDB replication** — 6–8 devices per store, consistent under intermittent connectivity
- **Linux services and drivers** — plus deployment automation, backups and multi-language support
- **Full product lifecycle** — from requirements and OS selection to field deployment and maintenance

Selected projects

PRODUCTS I DESIGNED, BUILT AND RUN

MiniaturZ

IN PRODUCTION · LIVE DEMO

Engineered end-to-end SaaS from scratch: a real-time, multi-user 3D platform enabling therapists to replicate a specific clinical technique with their patients online.

- **Solved critical 3D rendering performance bottlenecks** with custom asset preloading system and KTX2/Draco compression for efficient loading.
- **Architected full infrastructure**: containerized microservices, WebRTC for video calls, Stripe for subscriptions, and CI/CD pipeline.
- **Built real-time multiplayer 3D synchronization** using custom protocol over WebSockets (Socket.IO).

Next.js · TypeScript · Three.js · WebRTC · Socket.IO · Docker · Stripe · miniaturz.com/playroom?demo

Derivio

2026 · NO LIVE LINK

B2B SaaS connecting psychology centers with verified external psychologists matched by therapeutic style. Drove 100% of product strategy, architecture, and implementation autonomously.

- **Designed a 3-layer matching algorithm** combining deterministic scoring with LLM inference and semantic embeddings — validated across 44 clinical scenarios.
- **Built E2E encryption for patient data** (dual-key, PIN-based recovery), 30-day automatic data retention, GDPR-compliant by design.
- **Architected modular Next.js application** deployed across multiple Docker containers on VPS with custom CI/CD.
- Built end-to-end: Stripe payments, NextAuth (OAuth + magic link), REST API, custom observability, transactional emails, cron jobs with failure alerts.

TypeScript · Next.js · MySQL · Claude API · Google Embeddings · Docker · Nginx · Stripe

Agent Desk

OPEN SOURCE

Run many AI coding agents in parallel without babysitting terminals. Agent Desk shows every question your sessions ask and lets you answer from your desk or your phone, follow their sub-agents live and review what they deliver. Local, no cloud, zero runtime dependencies.

TypeScript · Node.js (standard library only) · Claude Code hooks · VS Code extension · Tailscale · github.com/jesusizq/agent-desk

Triangulator

LIVE DEMO

High-performance C++ geometry processing pipeline with WebAssembly compilation and containerized microservice deployment.

- **Hybrid Compute Architecture**: Seamlessly switches between server-side (native C++) and client-side (WASM) execution
- **WebAssembly Optimization**: Shared C++ core library compiled with Emscripten/embind for zero-copy JS interop
- **Scalable Microservice**: Stateless API design using cpp-httplib, containerized with multi-stage Docker builds
- **Production-Ready Gateway**: Cloud-native deployment with Caddy (auto-SSL) and Nginx for optimized delivery

C++17 · WebAssembly · Emscripten · CMake · Docker · GoogleTest · triangulator.jesusizquierdo.dev · github.com/jesusizq/2DPolygonTriangulator