



» Enthusiast about Functional Programming, software engineering/craftsmanship, clean code, architectural/design patterns and programming languages theory.

» Good at identifying inefficiencies, finding solutions and improving processes. I enjoy collaborating with diverse teams to solve challenging problems.

SUMMARY

[ 📄👁️ Please, read through my best practices and skills | 📖 Can also read the extended CV version ]

With a keen interest in Scala and other modern functional languages, my background is mostly about designing/implementing backend systems and APIs. Skilled in analyzing requirements and translating them into code. Focused on code readability, expressiveness, precision, conciseness, and correctness. Committed to continuous improvement & continuous refactor and code reuse as a mantra. Accidental-complexity fighter. Curious by nature, proactive and self-taught. I enjoy contributing to developer experience and building products. Like learning 😊, love crafting ❤️.

»»» EXPERIENCE

Backend Software Engineer | Quarkus, Reactive

Tymit, 2024–2025

- 🎯 Authored an RFC with actionable proposals to align the current architecture with Clean Architecture™ & SOLID principles, with a special focus on the Interface Segregation (ISP) and Single Responsibility Principle (SRP), plus other enhancements.
- » Engaged with the integrations team to support the migration of hundreds of thousands of customers into our system, while also developing new public-facing APIs to enable a new business model requiring first-party and third-party B2B backend integrations.
- » Improved testing reliability by introducing Property-Based Testing and showcasing its principles to the backend team, ensuring more robust testing including edge-cases, generating thousands of test values.
- » Spearheaded API contract segregation into dedicated modules, enabling the publication of versioned API artifacts for seamless and efficient cross-service integration using Gradle, contributing to DX.
- » Shared coding best practices and actively contributed to code reviews and BE discussions.
- Stack: Quarkus, Microservices, Reactive Programming, AWS (ECS, DynamoDB, SQS, SNS, RDS), Liquibase,...

Software Engineer | C++, MQL

Self employed, 2022–2023

- 🎯 Established my own company to provide software engineering services as a contractor.
- » Algorithmic trading, research, design & implementation of a risk management system for loss control, as well as key metrics collection/reporting using DDD & Hexagonal arch.
- » High level design of a multi-level affiliation system with referral fee distribution.
- » Took a break to learn about finances, markets, investments and expand Fintech expertise.

Software Engineer | Spring Boot, Kotlin

N26 Bank, 2021–2022

- 🎯 Collaborated to streamlining the processing of banking fees with the goal of having a more centralized and clear domain boundaries for charging fees.
- » Joined the Memberships team, which was in charge of designing and operating a new fee platform intended to be the new-standardized way of charging company-wide banking fees.
- » This involved creating a dedicated microservice, repurposing and refactoring of existent ones (by DDD/CleanArch), along with APIs, as well as DB migrations.
- » Being a bank, testing and security were some of the most important and cared matters (pyramid testing, contract testing, regression, TDD,...).
- Stack: Kotlin, Spring Boot, Microservices, EDA, Kafka, REST, Postgres, Datadog, ELK, AWS,...

Senior Software Engineer | Spring Boot, Java

GFT IT Consulting, 2016–2021

- 🎯 Successfully delivered a large and complex regulatory banking project to provide external, permission-based access to banking APIs, ensuring on-time completion. Then, was called to contribute to building a new bank architecture for high-concurrency from the acquired know-how.
- » Banc Sabadell: **Middleware Architecture** (2020–2021) – Redefine from scratch a whole new framework and development environment based on MsA, Spring Boot, API-First, Event Driven, DDD, Hexagonal, and using OKR project framework – **Role:** Software Engineer.
- » Banc Sabadell: **PSD2 & APIfication** (2018–2020) – Design & implement the APIs and requirements for the European **PSD2 regulation**; pioneering the API exposition for the Bank. Define and implement architectural components for the MsA. Mentoring and onboarding new members. Got established as the technical lead for the dev team – **Role:** Tech Lead.
- » Bankinter: **Microservice Architecture** (2017) – Development of an architecture based on MsA and Spring, providing the pre-configured components: from security, tracing, audit, error handling to cryptography; enabling developers to easily get started.

Software Engineer | Mobile development

ICG Software, 2015–2016

- » Developed a new Android-native warehouse management app, featuring barcode scanning, offline operation, an object database with Realm, and integration with the in-house developed ERP.
- » Successfully migrated a whole app for a franchise clothing store to native Windows Phone. Our client needed that we developed our Android app for Windows Phone, to deploy it globally on all their stores.

Researcher/developer | Python

AI Research Institute IIIA-CSIC, 2012–2013

- NEWMATICA: Intelligent and Energy Efficient Advanced System for Vacuum Waste Collection.
- » Investigated Machine Learning algorithms based on Approximate Dynamic Programming (RL).
- » Co-developed a waste collecting simulator to optimize operational plans, run simulations, collect data and process results.

Presentations

- 2020 Functional Programming super-short intro
- 2018 A technical introduction to Blockchain technology

»»» EDUCATION (to see more, view the cv version)

M.Sc. Open Source Software Eng.

University of Lleida (ES) / VIA UC (DK), 2011

- » Master Thesis: Setting up and deployment of Sauron system for DNS system management at University of Lleida. Mobility internship program at VIA UC, Denmark.

CONTACT

🌐 US/EU Worldwide remote

✉ gerard.bosch@gmail.com

🌐 linkedin.com/in/gerard-bosch

🐱 github.com/gerardbosch

💻 gerardbosch.xyz

Fork me on GitHub

PROGRAMMING LANG

Scala 3	★★★★★
Kotlin	★★★★★
Java 23	★★★★★
Haskell	★★★★★
Bash	★★★★★
Python	★★★★★
Perl	★★★★★

TECHNOLOGIES

- 🔧 Spring Boot 🔧 Vavr 🔑 oAuth/JWT
- ☁ Cloud-Native / AWS 🐳 Docker / Podman
- 🗄 SQL 🗄 JOOQ 🏠 API First & OpenAPI
- 🔗 OpenShift ☁ Cloudflare
- 📦 Devcontainers 📦 Testcontainers
- 🔧 Maven/Gradle/sbt 🌐 JGitVer

TOOLS

- 🔧 IntelliJ / VSCode / vim :)
- 🐱 GitHub Actions 🔧 GitLab CI 👤 Jenkins
- 🔗 Git ➤ ZSH 💬 Codestream
- 🔧 CI/CD 🐧 Linux 🍏 Nix

METHODOLOGIES

- Agile, OKR, SCRUM, Kanban
- TDD, Property-Based Testing, Mutation Testing
- Domain Driven Design, Hexagonal, Clean Architecture
- API-First, OpenAPI, API Management, API Governance
- Microservices, Event-Driven Architecture
- SOLID, Design Patterns, Refactoring, eXtreme Programming

IN GRAPHICS...

