

Giulio Pimenoff Verdolin

SOFTWARE ENGINEER

☎ (+39) 3883274066 | ✉ giuliopime@gmail.com | 🏠 giuliopime.dev | 📱 Giuliopime | 🌐 giuliopime

Work Experience

BMAT

Barcelona, Spain

BACKEND - BIG DATA DEVELOPER

Sep. 2025 - Jan. 2026

- Designed and developed a system to retrieve statistics about artist's revenue, using **Django** for the REST API, **DuckDB** and **Spark** for data processing, **Apache Iceberg** and **S3** as base for the data Lakehouse and multiple **AWS** services (Glue, Batch, Athena and more...).
- Improve performance by implementing multiprocessing logic for resource intensive code sections.

Niksen App

Milan, Italy - Remote

BACKEND DEVELOPER

Feb. 2024 - Jun. 2024

- Refactored multiple Python backend microservices and integrated AWS Cognito for user authentication.
- Worked in a fast pace startup environment where communication and independence was key.

Corvallis S.r.l.

Padua, Italy

SOFTWARE ENGINEER INTERN

Jan. 2022 - Mar. 2022

- Designed and developed an analytics dashboard for one of the company SaaS using Angular with Typescript.
- Refactored the existing SpringBoot (Java) backend and implemented new REST endpoints required by the dashboard, while also improving some queries to a MariaDB database by more than 15%.

Education

University of Trento, DISI

Trento, Italy

B.S. IN COMPUTER SCIENCE

Sep. 2022 - Mar. 2026

- Relevant courses: Computer programming 1-2, Functional programming, Advanced programming, Software engineering, Calculus, Math foundation for cs, Algorithm and data structures, Databases, Networking, Operating systems
- Average grade 27/30 - GPA 3.7

Projects

Astro

[Website](#) - [GitHub](#)

DISCORD APPLICATION / BOT

Apr. 2020 - Present

- Created and scaled a Discord bot application from 0 to **15k+ distinct daily active users**.
- Running a **SpringBoot Kotlin backend** split into **4 microservices** and a shared core layer, with **MongoDB** as the database (11M+ users stored), **Redis** for performance and caching, **BigQuery** for data analysis, Sentry for performance and failure monitoring, Chargebee / Stripe for premium subscriptions managing, **Docker**, **git** and **Semaphore** for **CI/CD**.
- Developed a web dashboard using **NuxtJS**, used by 1k+ distinct users per month.
- Setup a **Kubernetes cluster** hosted on Hetzner servers using **Pulumi** for **infrastructure-as-code**. The cluster supports auto-scaling and has a Grafana dashboard for monitoring purposes. Thanks to CI/CD, all microservices are containerized and deployed to the k8s cluster, which has a staging environment for testing. Also self-hosting Redis on a dedicated server to reduce costs.
- Contributed to **JDA** (an **open-source library** for interacting with the Discord APIs) by adding support for Discord application premium entitlements management (view pull request)
- Did **Q/A testing** for Discord REST APIs (cannot disclose the specific feature that was tested)
- Managing an active community of 6k+ members on Discord, and provided chat support for over 4k customer tickets.
- Migrated the use of multiple services / frameworks (expressJS, Ktor, InfluxDB, and more...) while keeping the service active and online during migrations. Also migrated the project from a monolith to a microservice architecture to support horizontal scaling and sharding.

Index

NATIVE MOBILE APP

Ago. 2022 - Present

- Designed and developed a **native Android app** using **Jetpack Compose** and an **iOS app** with **SwiftUI**, with user authentication, **offline first support** (with Android Room, Datastore and SwiftData) and **real-time data via websockets**. Published on the Google Play Store and App Store.
- Built a **REST API** using Kotlin with **Ktor**, fully documented conforming to the **openAPI Specification**.
- Rolled my own authentication with support for email and password (email verification included) and OAuth services such as Google and Apple. Includes session management using cookies and authorization logic.
- Designed the database schema to efficiently store data in **PostgreSQL**, handled migrations, and used Redis for temporary or performance critical data such as user auth sessions.
- Implemented **real-time data and collaboration** between users with **websockets**. **RabbitMQ** is used as the main dispatcher of events, thus ensuring the service is horizontally scalable.
- Integrated **Prometheus** and **Grafana** for performance monitoring, and **BigQuery** for data analysis.
- Implemented **CI/CD** using **Docker** and **CircleCI**. Deploying the monolith backend on my **Kubernetes** cluster, setup with **Terraform** and **ArgoCD** to automate deployments.