

# Francesco De Rosa

Backend software engineer

(+39) 340 077 17 33

francescodero@outlook.it

[github.com/fDero](https://github.com/fDero)

## EDUCATION

---

- 🎓 **University:** Bachelor's degree in *Computer Science* at the University of Naples Federico II obtained in March 2025, with a grade of 110+/110 (graduated with honors).
- 📅 **High school:** Got my high school diploma in 2021 at Liceo Scientifico Statale "Elio Vittorini" in Naples.

## PROJECTS

---

- 🔧 **Basalt:** A compiler for my own low-level type-safe programming language with full support for generics, type-inference and pseudo-polymorphism. The compiler uses LLVM as a backend to generate machine code for most modern architectures, generating object files compatible with most modern linkers.
  - 🌀 From this project I learned how to manage complexity in a large codebase and how to design for maintainability in the long run. I also learned how to manage tasks and deadlines in a large project and how to respond to unforeseen difficulties.
- 🔥 **DietiDeals24:** A dummy e-commerce webapp with Java Spring Boot backend, with JWT-based authentication, Redis integration for caching and session management, and a PostgreSQL database for storing products and orders. Developed locally with Docker-compose and released on AWS and integrated with Google OAuth 2.0. It also supports OpenAPI (Swagger) for self documentation.
  - 🌀 From this project I learned how to work with the cloud, and most importantly how REST APIs work, how the codebase lifecycle of a web-based backend works and how to use state of the art tools and technologies to build modern and scalable backends.
- 📄 **IssuuDownloader:** A command-line utility to download PDFs from Issuu.com in bulk. It uses multiple threads to download files in parallel and it can be used to download entire collections of documents in one go. It is also able to resume downloads from where they left off using a cache file.
  - 🌀 From this project I learned how to understand and reverse-engineer network protocols and non-documented APIs. I also learned how to use threads in python and how to take advantage of parallelism to speed up I/O-bound tasks.
- 🧠 **BackproPy:** A simple and beginner-friendly explanatory implementation of the back-propagation algorithm for neural networks in Python. It takes inspiration from "minigrad" by Andrej Karpathy and it is designed to improve it by making it less error-prone and more readable.
  - 🌀 From this project I learned how the back-propagation algorithm works and how to use it to train simple neural networks.

## SKILLS

---

- 🔗 **Languages:** C, C++, Java, Python, Go (golang)
- 🔗 **Tools:** Git, Docker, Docker-compose, Redis, PostgreSQL