

# LEONARDO VICENTINI

SOFTWARE ENGINEER

✉ vicentini.leonardo99@gmail.com | 🏠 leonardovicentini.com | 📷 vicentinileonardo | 🌐 leonardovicentini

“Learning never exhausts the Mind.” — Leonardo da Vinci

## Skills

**Programming** Python, Java, JavaScript, C, Solidity  
**Technologies** SQL (MySQL, PostgreSQL), MongoDB, Redis, Elasticsearch, Node.js, NGINX, Linux, Bash, Git, Docker, Kubernetes  
**Languages** English, Italian (native)

## Education

### University of Trento

Trento, Italy

MASTER'S DEGREE IN COMPUTER SCIENCE (ENGLISH) — CURRENT GRADES: 29.83/30

Sep. 2021 — Expected Mar. 2024

- Relevant courses: Distributed Systems, Cloud Computing, Service Design, Security Testing, Blockchain, Data Mining, HPC
- Ongoing theses work: Architectural analysis and design of a Kubernetes-based SaaS

### University of Trento

Trento, Italy

BACHELOR'S DEGREE IN COMPUTER SCIENCE — GRADE: 106/110

Sep. 2018 — Sep. 2021

- Relevant courses: Algorithms & Data Structures, Software Engineering, OOP, Operating Systems, Databases, Networks, HCI, ML

## Work Experience

### FIPIC – Italian Wheelchair Basketball Federation

Rome, Italy (Remote)

SOFTWARE ENGINEER INTERN - BACKEND

Feb. 2021 — Jun. 2021

- Co-led a 4-member team in developing a Federation's historical data and multimedia archive, reducing the estimated project completion time by 50% through customer-centered development strategies.
- Elicited comprehensive requirements from diverse stakeholders, demonstrating strong communication and analytical skills.
- Contributed in the design of a pipeline based on the ELK stack to achieve dynamic data visualization and in-depth analysis.
- Designed and implemented RESTful APIs to perform CRUD operations against a MySQL database using Node.js with Express.
- Created a multimedia collector component by leveraging Google Drive APIs and OAuth 2.0 authentication.
- Deployed and configured the entire system on a dedicated Ubuntu server using NGINX, UFW and PM2.

## Projects

### ProjectsChain — Ethereum-based CAD designs marketplace

SOLIDITY, WEB3.JS, NODE.JS, DOCKER, REDIS, IPFS — [ [CODE](#) | [REPORT](#) | [DEMO](#) ]

- Designed backend and blockchain-related architecture. Implemented a web server and related RESTful APIs that performs CRUD operations against a properly configured Redis database, secured by a specifically designed digital signature mechanism.
- Responsible for the Chainlink node and jobs configurations together with its specific smart contract development and integration.
- Contributed to smart contracts design together with optimization and testing. Reduced code size by ~15% and gas fees by ~20%.

### Daytrip — service-oriented web app for daytrips suggestions in Italy

DOCKER, PYTHON, FLASK, NODE.JS, NGINX, MONGODB — [ [CODE](#) | [REPORT](#) | [DEMO](#) ]

- Responsible for the design and implementation of 14 out of 19 services (managed with Docker Compose), spanning from data layer to business logic and process centric services that fetch, transform and elaborate data to suggest destinations to users.
- Developed a recommendation algorithm using data from multiple APIs, including TomTom, OpenStreetMap, and others.
- Proposed and implemented a workaround leveraging AWS Lambda for Docker issues of a service using matplotlib Python library.

### Convex hull parallel solver

C, MPI, OPENMP — [ [CODE](#) | [REPORT](#) ]

- Ported a C++ implementation of a serial algorithm solving convex hull problem into C source code.
- Designed, implemented and tested a parallel algorithm exploiting MPI and OpenMP libraries obtaining a maximum speedup of 13x w.r.t. serial implementation on an HPC cluster using PBS configurations on shell scripts.

### Digital watermarking tool

PYTHON, OPENCV, NUMPY — [ [CODE](#) ]

- Developed a digital watermarking suite as part of a university competition on Multimedia Data Security.
- Proposed the project workflow in order to meet the requirements of the produced code within the restricted time constraints.
- Devised algorithms to achieve a robust and invisible watermark embedding that resulted the 2nd best in both metrics.
- Exploited cloud services (IaaS) to parallelize the attack algorithm against other teams' watermarked images, reducing the computation time by 33% attacking 10 groups out of 10 with a success rate of 93.3% for the 30 images involved.