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 8+ diverse stakeholders, demonstrating strong communication and analytical skills.
- Contributed in the design of a pipeline based on the ELK stack to build 4 dynamic data visualization dashboards.
- Designed and implemented 70+ RESTful endpoints on a Node.js server to perform CRUD operations against a MySQL database.
- 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 for an NFT marketplace with a royalty-based compensation scheme.
- Built a web server and related RESTful APIs (11 endpoints) that performs CRUD operations against a properly configured Redis database, integrated and secured by a specifically adapted 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 4 public APIs, including TomTom, OpenStreetMap, and others.
- Operated a workaround leveraging AWS Lambda for Docker issues on a core service decreasing deploy failures from 66% to ∼0%.

Convex hull parallel solver

C, MPI, OpenMP — [**Code** | **Report**]

- Converted a C++ implementation of a serial algorithm solving convex hull problem into C source code.
- Conceptualized, coded 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 several 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 against ∼40 students.
- Proposed the project workflow in order to meet the requirements of the produced code within strict 2-weeks time windows.
- Devised algorithms to achieve a robust and invisible watermark embedding that resulted the 2nd best in both metrics used.
- 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.