LEONARDO **VICENTINI**

SOFTWARE ENGINEER

✉ vicentini.leonardo99@gmail.com | 🏠 leonardovicentini.com | 💻 vicentinileonardo | in leonardovicentini

"Learning never exhausts the Mind." — Leonardo da Vinci

Skills

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

Education

University of Trento   *Trento, Italy*
MASTER'S DEGREE IN COMPUTER SCIENCE (ENGLISH) — CURRENT GRADES: 29.4/30   *Sep. 2021 — Present*
- Relevant courses: Distributed Systems, Cloud Computing, Service Design, Security Testing, Blockchain, Data Mining, HPC

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

ESA – European Space Agency (European Space Operations Centre)   *Darmstadt, Germany*
SOFTWARE ENGINEER INTERN - CLOUD & INFRASTRUCTURE   *Apr. 2024 — Jul. 2024*
- Deployed a Service Status Board with a GitLab CI/CD pipeline to validate critical metadata related to incidents and interventions on critical operational systems and services (servers, VMs, networks) used across 3 Space Mission Operations teams within ESOC.
- Designed a Performance Dashboard based on Prometheus, Grafana and several data exporters, resulting in an end-to-end solution that includes data extraction from heterogeneous data sources (internal APIs, VMware APIs, etc.)
- Implemented data analysis and visualization with 3 dashboards to monitor the health and performance of infrastructure services.
- Leveraged Ansible for deployment automation of some system components, such as Prometheus Node Exporters on 10s of VMs spread across the Operations Local Area Networks.
- Deployed the Performance Dashboard following a cloud-native approach leveraging Kubernetes.

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 to the design of a data 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.

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%.

Multi-level distributed cache
JAVA, AKKA — [**CODE** | **REPORT**]
- Designed 4 operation protocols (Read, Write, Critical Read, Critical Write) along with the other team member.
- Proposed and implemented the request and response message mechanisms to correctly route messages through the system.
- Implemented Critical Write, Crash Detection and Recovery algorithms essential for the goal of the simulated distributed system.