Max Azatian

max.azatian@gmail.com | +49 178 123 45 67 | GitHub | LinkedIn

Summary

Backend developer with experience in crafting efficient systems. Combine technical precision with teamwork to solve problems through a clean, maintainable design.

Experience

Software Engineer (Part-time) Jun 2021 – Present
Self-employed *Remote*

- Reached 90% pilot user adoption for pet salon booking system by developing Django/PostgreSQL MVP with auto scheduling, accompanied by Doxygen documentation featuring UML diagrams (class, activity, deployment).
- Increased monthly active user growth by 40% for board game community platform via creating Django REST backend implementation supporting profile creation, blog posts, and filtered game searches with Bootstrap-based UI.

Projects

Integr8sCode | *Python, FastAPI, Pydantic, Kubernetes, Svelte, MongoDB, Prometheus, Grafana*

- Reduced memory usage by 25%, tracked in Prometheus, by enforcing per-pod CPU/memory limits (CPU: 100m, memory: 128Mi), adding auto-scaling policies (Horizontal Pod Autoscaler), and optimizing Docker layers (e.g., moving nonessential packages and using `python:{version}-slim`).
- Achieved 30K+ daily script executions with an error rate below 0.3%, monitored via Grafana, by orchestrating ephemeral K8s pods and adding request validation with Pydantic.
- Raised backend test coverage to 92%, as measured by Codecov, by introducing unit/integration tests for critical modules and logging/reporting coverage in HTML reports.

Flet-Chat | *Python, FastAPI, Flet, PostgreSQL, Redis, Docker, Pydantic*

- Created real-time chat with an average latency of less than 100ms, confirmed by stress tests, by integrating FastAPI WebSockets and Flet UI components.
- Acquired "A" code quality (measured by Codacy) and 85% test coverage, validated via Codecov, by standardizing linting rules and expanding unit/integration tests.
- Reduced database load by 31.2%, tracked in Docker, by adopting Redis for caching ephemeral data.

ResuMariner | *Python, FastAPI, Pydantic, Neo4j, Qdrant, Docker, Traefik, Redis, LLM*

- Increased data retrieval accuracy by 78% compared to keyword-based SQL queries via implementing hybrid search (Neo4j graph traversals, Qdrant vector similarity), validated through precision/recall on technical skill matching.
- Lowered mean processing time from 9s to 2.5s by creating parallel processing pipelines and LLM prompting strategies (context windowing, structured output schemas).
- Added zero public exposure of internal services, confirmed via OWASP-based penetration testing, by implementing network isolation with Traefik's internal entrypoints and dedicated bridge networks.

Education

Technical University of Munich Oct 2024 – Present
M.Sc. Computer Science *Munich, Germany*

Technical University of Munich Oct 2019 – Aug 2024
B.Sc. Computer Science *Munich, Germany*

Technical Skills

Languages: Python, JavaScript, HTML/CSS
Backend: FastAPI, Flask, Django, ORM (SQLAlchemy), Linters (Flake8, Ruff), Locust, RESTful APIs
Databases: PostgreSQL, MongoDB, SQLite, Redis, Neo4j
DevOps: Docker, Docker Compose, Kubernetes, Git, Linux, Cloud platforms (GCP, Azure), Traefik
Monitoring: Prometheus, Grafana

Additional Information

Languages: English (C1), German (C1), Japanese (B1).