Steffen Reindl

Naupliastr. 28
81547, Munich
Germany

Phone: On request.
Email: str@malignat.us
Portfolio: github.com/MordecaiMalignatus

SKILLS AND EXPERIENCE

Ryte, Backend Software Engineer, **2017 – present.**

Maintained a Scala web crawler central to the company, distributed over YARN and Flink, the output of which is analyzed by a Scala service written in Spark. This system is running on AWS EMR, and was subsequently improved to support Javascript rendering and crawling single-page applications.

EDUCATION

Technical University of Munich, Incomplete B.Sc. Computer Science, (2016 – 2018)

PROGRAMMING LANGUAGES

Scala	Go-to language for implementing services, specifically the Akka ecosystem; also possessing experience of the Typelevel family of libraries.
Python	Proficiency in building robust services and rapid prototyping, basic experience in data science.
Elixir/OTP	Experience in building highly-available services to be used as building blocks in an infrastructure; also some experience using Phoenix as web-development backend.
Rust	Experience in creating CLI tools that do specific, small things well, as well as agents/daemons for metrics. Also created a desktop application using Rust and `web-view`

TECHNOLOGIES

Infrastructure	I use tools like Terraform and CloudFormation to construct all my infrastructure from day one.
AWS	I have used Amazon Web Services to power large-scale crawlers and analysis pipelines.
Databases	Primarily worked with PostgreSQL and variations of SQL databases. In addition I also made heavy use of PrestoDB and DynamoDB.
Hadoop	Created and maintained applications on top of Apache Flink, Apache Spark and YARN.
CI/CD	Experience in setting up the software, process and tooling needed to continuously test and deploy services and products.
Docker	Experience packaging services to be used with Docker and integrating Docker into the build process.

INTERESTS & FOCUSES

Metrics	What isn't measureable isn't actionable. I add metrics and logs to any service I write, as well as supporting infrastructure like dashboards and log aggregation.
Systems Design	I design systems for failure, which results in simple, but reliable architectures, easy to maintain and expand.
Tooling	I am very fond of having good tools to work with, and will create and maintain tools that enable and simplify future changes.