Frederik Hoffmann

707-816-0124 | hoffmann3733@gmail.com | linkedin.com/in/frederikhoffmann | https://fredferd1995.github.io

Education

Georgia Institute of Technology
M.S. Computer Science *2021 - 2024 (expected)*

University of California, Santa Barbara
B.S. Physics *2017*

Technical Skills

Cloud: AWS
Languages: Java, Python, C/C++, MATLAB
Tools and Frameworks: JUnit, Guice, PyTest, JUnit, Postman, SQL, NoSQL
Developer Tools: Git, Docker, Kubernetes
Libraries: pandas, NumPy, Matplotlib
Dev Environment/DevOps: Linux (as build/dev env), Unix, bash. CI/CD

Experience

Software Development Engineer 2022 - 2023
Amazon Web Services (AWS)

- Designed/coded attribute-based access control service.
  - Utilized Java, JUnit, DynamoDB, KMS, AWS Identity.
  - Allows AWS OpsCenter customers (56K active WoW) to increase security posture by downscoping resource access for users based on resource metadata, AWS context keys.
- Automated security tests for 13 internal/external APIs, bringing coverage to 100%.
  - Before automation, tests were manual/slow to develop.
  - Wrote providers to populate API requests using AWS SDK for Python (boto3), integrated with internal AppSec PyTest framework.
  - Built CI/CD pipeline, deployed security tests in order to gatekeep promotions, generate tickets on failure.
  - Improved team development velocity from months to weeks since security requirements became automated in pipeline.
  - Hosted lunch-and-learn, wrote documentation for easy future support by team.
- Wrote Java code, JUnit/Guice tests on 4-dev team for design, implementation of new feature to support cross-account resource access, e.g. User A accesses User B's resource.
  - Used DynamoDB, SQS and SNS to periodically cache data from AWS Organizations to determine if caller has appropriate permissions.
  - Reduced latency by removing call to AWS Organizations API on each client request (found to be slow during initial research), saving up to 30s.
- Created CloudWatch alarms, metrics, logging, dashboards in Ruby.
  - Logging and runbook additions brought on-call MTTR down to minutes from hours.
- Authored design documents and drove team consensus on new feature design and implementation.

Systems Software Engineer, Infrared Imaging 2018 - 2022
Lockheed Martin

- Implemented software simulation of hardware-based image-processing suite, enabling validation/introduction of new system.
  - Obsoleted need for additional in-flight testing, saving months of planning and thousands of dollars.
- Developed, released production software to test camera systems, measure performance against requirements.
  - Resulted in complete reduction of infant mortality failure mode, and improvements in product yield by up to 50% in some runs.
- Led technical projects, 4 technicians, 2 test engineers. Communicated technical details, and defended proposals, analysis to site leadership.

Customer Success Manager 2017 - 2018
AppFolio, Inc

- Trained 100+ clients on usage of AppFolio SaaS Property Management software.

PROJECTS

Multi-threaded Distributed File System | *C++14 w/ STL, gRPC, Docker, Linux, WSL2, Protocol Buffers*

- Distributed multi-threaded file server/storage.
- Lists, uploads, downloads, deletes shared files across multiple clients (threads).
- Utilized gRPC/Protocol Buffers to communicate actions across clients/server.
- Implemented mutexes/other synchronization constructs for thread safety.

Classification-Based Learner for Stock Trading | *Python, Pandas, NumPy, Linux*

- Stock bot, advises buy/sell/short decisions.
- Trained via Classification-Based Decision Tree learner on input data set of historical price/volume data.
- Yielded 8 percent YoY positive returns after training, versus 7 percent against reference buy-and-hold strategy.