Michael Leonhard’s Resume

michael206@gmail.com   415.361.1808   San Francisco, California

Cozy Date App, Founder, 2019-2022

- Studied UX design
- Interviewed hundreds of dating app users and identified key pain points
- Wrote user personas
- Wrote strategies to reduce user pain and enhance positive user experiences
- Created user flow diagrams
- Designed the app UI
- Implemented iOS & Android apps in Flutter + Dart
- Implemented API server with Rust, rouille web server, diesel SQL library, Heroku, PostgreSQL, JSON
- Coded some integration tests for tricky codepaths, including some end-to-end tests
- Hired a designer to design the website, logo, color scheme, and app icon
- Implemented the website in HTML, CSS, JavaScript, Bootstrap, Netlify: www.cozydate.com
- Tested the app with users and improved the UI
- Researched monetization strategies, interviewed cafe owners as potential ad buyers

Google, Technical Infrastructure, Software Engineer, 2013-2018

= Assimilator Team, 2015-2018 =
Assimilator is an internal system that manages system software on Google's server machines. It installs/updates/rolls-back the Linux kernel, firmware, OS software, machine certificates, and Google system software. Assimilator daemon runs on every Google machine and router. Every datacenter has its own controller instance. Its scheduling functions are extremely complicated because they implement canaries and pipelined releases and comply with machine downtime restrictions for Google Cloud users.

The project had accumulated a lot of technical debt and its maintainers had all left. I was the first member of the new dev team. What I did:

- Refactored the C++ codebase to make it testable. Wrote unit tests. Wrote integration tests for all features in use.
- Added features to unblock other teams
- Removed obsolete features
- Learned Golang and rewrote portions of the system in Golang. This was a multi-year effort with my excellent teammates.
- Mentored a junior teammate
- Got promoted

= Machine Database Team, 2013-2015 =

Machine Database was the database of record for every piece of hardware costing more than $5 in every Google data center. What I did:

- Helped teammates re-write API server in Java & gRPC
- Ported 100 C++ client programs from other teams to use the new API, and numerous Java & Python clients
- Attention to detail: While porting one client, I identified a bug in how it used our API. The bug would have caused a major outage affecting public apps. I submitted a PR. The other team enthusiastically accepted it and told our Director who gave me a bonus.
- Set up continuous integration tests for schema migrations with API deployments and rollbacks. This saved us a lot of trouble and increased our velocity in changing the database.
- Added features to support new kinds of hardware and new datacenter management software
- Removed obsolete columns, tables, and API features

Rest Backup, Founder, 2010-2012

- Created a REST API for storing application data backup files.
- Built a web app for downloading files and for managing backup accounts
- Server was Java on Linux, deployed on EC2, S3, DynamoDB, ELB, SQS, and SES, and CloudFront.
- Wrote functional tests in Python.
- Created backup software for Windows using Python. Backup script language supports MS SQL Server, MS Access and other file-based databases. Wrote Windows service to upload in the background. Inno Setup installer scripted in Pascal. Automated tests on 12 flavors of Windows.
- Sales via cold calling and direct mail. Tech support.

Amazon Web Services, Software Development Engineer, 2007-2010

SimpleDB is a hosted NoSQL database service. It is the predecessor of Amazon DynamoDB. What I did:

- Owned the subsystem that gathers and reports customer usage for billing purposes. Wrote, tested, and deployed new versions to keep up with SimpleDB architecture changes. Set up the subsystem for new SimpleDB instances in Northern California, Ireland, and Singapore data centers.
- Wrote and maintained operations documentation
- Wrote functional tests for new API features, increased test coverage, added regression tests
- Wrote and maintained the network partition test that stresses the distributed architecture of SimpleDB. With this test, we made SimpleDB tolerate all kinds of network and server problems.
- Developed and maintained the team's operations tools
- Wrote and maintained command-line tools, web-based tools, monitors, alarms, automated processes, documentation, and procedures

- Oncall

Opportunity International, MIS Assistant (part-time), 2001-2007

Managed servers and networks, maintained the website, and supported onsite and offsite users.

FOSS contributions, Rust libraries, GitHub, Personal projects

- cloudping.info
- crates.io/crates/safe-regex - A Rust regular expression library. It compiles regular expressions to Rust code, which is then compiled and optimized as part of your program.
- crates.io/crates/safina - A safe Rust async runtime.
- crates.io/crates/beatrice - A modular HTTP server library in Rust. Uses async code internally for excellent performance under load. Unique feature: You can write handlers in straight-forward non-async code.
- github.com/mleonhard/maggie-rs - An app server library in Rust. Implements the server side of a new JSON-based remote UI protocol. Code responsive apps in Rust, without macros or complicated generic types.
- crates.io/crates/dns-server - A threaded DNS server library in Rust.
- crates.io/crates/prob-rate-limiter - A probabilistic rate limiter library in Rust.
- See others on my website: www.tamale.net

Education

Bachelor of Science in Computer Science, University of Illinois at Chicago, 2006
Published Research Paper: A Comparative Study of Three Random Password Generators. IEEE EIT 2007.
National Taiwan Normal University, Mandarin Training Center, Taipei, 2010 – 2011
Studied Mandarin Chinese full-time.
Chiba University, Japan, Exchange Student, 2005
Studied Japanese language and culture. Awarded Freeman Asia Scholarship and J-PAC stipend.