

Spencer Scorcelletti

I'm a Software Engineer/Architect, working in fintech, who specializes in cloud-native, distributed systems

San Francisco, CA | s: <https://spenc.es> | e: contact@spenc.es

Work Experience

Credit Karma, Inc., Jan '18 - present

Financial Tools and Services - Senior Software Engineer

- Lead engineer/architect, Credit Karma Savings (backend)
 - Owned architecture for system that makes FDIC insured savings accounts available to 100 million users [1]
 - Achieved a fault-tolerant, cloud-native system design that safely handles financial transactions, and business-critical notifications for Fraud, et al.
 - Designed a microservice that drives banking features and exposes an API, which other verticals within the company can integrate with
 - Built an event-driven pipeline that reliably delivers over 2 million critical messages per day with 99.9999% uptime
 - Prioritized observability, logging, and metrics, which led to mission-critical bugs being discovered and fixed ahead of launch
 - Led team of engineers to go from nothing to delivering feature-complete code by end of Q1 '19, and launching to all users by end of Q3 '19
- Created `thrift-scrubber`, a customizable serialization protocol that redacts fields based on thrift IDL annotations

Enterprise Security Team - Senior Software Engineer

- Owned the rewrite of our `session-service` in Scala, which handles all of Credit Karma's SSO traffic
- Helped build our `key-service`, a key encapsulation mechanism as a service, achieving substantially better security and scale

Traveling/Independent Software Development, Mar '16 - Sept '17

- Bought a VW bus and traveled while prototyping various ideas of interest to me
 - `uselesscss`: a build tool for removing unused CSS by static analysis
 - `readout-markdown`: a client-side tool for inserting markdown content in your page as rendered HTML
 - `craft.js`: an implementation of the raft consensus algorithm, over WebRTC

Salesforce, Inc., Jan '14 - Jan '16

Compute Systems - Software Engineer MTS

- Worked on an in-memory, distributed, graph-based database to model state of next-generation data centers
 - Designed method for continuing development ahead of having customer requirements, using an intermediary query language and client-specified schemas
 - Helped design/code MVP using go, protobuf, and embedded consul.io consensus library
 - MVP handled multiple-thousand requests per second, guaranteeing strong consistency, while being fault-tolerant
- Owned a platform for integrating internal, legacy development/bug-tracking tools with GitHub enterprise
 - Handled authentication across platforms
 - Elegantly supported how to grandfather-in bugs tracked prior to this system, requiring no user intervention
 - Devised a way of extending new functionality through meta-data, for future-proofing further development
- Evangelized Docker development
 - Consulted with cloud architects on canonical Docker development practices
 - Reviewed Docker related code outside of team
 - Developed curriculum for in-house docker training material

Continuous Integration - Software Engineer MTS

- Worked on a next-generation, internal CI system
 - Product scope was build/test/reporting across three platforms: web (Linux), iOS, and Android
 - Featured auto-scaling elastic compute resources, heavily parallelized builds/test-runs, self-service support and customized compute environments
 - Owned design/development of a templating system that allowed teams to customize and maintain their test environments as code in GitHub
- Owned an OS X elastic compute service
 - Services run on top of highly available, gossip protocol layer, which provides consensus primitives
 - Clean API served authenticated, parameterized CRUD operations on compute nodes
- Served as team scrum master, mentored intern, moved team from Vagrant to Docker, and wrote team site
 - Advocated creating a team site as a top priority, which reduced time team spent on support by ~25%

Service Cloud - Software Engineer AMTS

- Named as an inventor on a patent with provisional status
- Full stack development of new features for Salesforce "Knowledge" (product)
- Fixed bugs, refactored old code, and wrote test automation

Intel Corp., May '12 - Jan '14

PVT Post-Si Validation - Software Engineering Intern (Co-op)

- Owned development of automatic robotic part handlers for post silicon validation

- Developed reliable pathfinding, self-calibration, and part handling capabilities for two \$25K+ 6-axis robots
- Abstracted work out to a robotic SDK that allowed different classes of robots (*6-axis, cartesian, etc...*) to be used interchangeably
- Integrated with interlocks, emergency switches, part-reading camera, vacuum suction cups, and part actuators to run and record results of automated test in a database
- Wrote GUI app for analogue test engineers to interface, via Modbus protocol, with thermal controllers

Salesforce, Inc., May '13 - Aug '13

Service Cloud - Software Engineering Intern

- Wrote majority of the front end development for Salesforce Knowledge's redesign which shipped into production
- Extracted separate device views into a single, responsive view across devices, which saved time spent on development and maintenance in the future

Education

University of Texas at Austin, Dec '13

College of Natural Sciences, Bachelor of Arts in Computer Science

- Thesis: *Using DSP to Determine Volume of Containers*
- Selected Projects:
 - Multi-Paxos variant of the Paxos algorithm
 - The Bellman-Ford (algorithm) is faster than Dijkstra, because it can better exploit machine architecture
 - A file-system and virtual memory from scratch, for a basic operating system

Skills

Programming Languages:

Scala, Swift, Java, go, C#, Node/JS, C, Python, Perl, x86

Tools/Technologies:

Microservice Architecture, SOA, SQL, RPC, GCP, AWS, Thrift, Docker, k8s, Grafana, Splunk,

Areas of work/academic interest:

Distributed systems, database design, software design, operating systems, formal specifications

Personal Interests

Making things, VW buses (1969-1979), motorcycles, sailing, music, art, writing and philosophy, and nature