# Jeremiah Boyle

**Software Engineer**

jeremiah.boyle@gmail.com
(415) 205-9443


## SKILLS

Backend swiss-army knife fluent in Python and C#, competent in C++ and Java, and an enthusiastic navigator of uncharted waters. Accustomed to agile methods, domain-driven design, and test-driven development. Recenly busy building microservice-based web backends supported by Amazon Web Services, PostgreSQL, Docker, AMQP.

## EXPERIENCE

**Principal Software Engineer** June 2018 - Present
iBeat San Francisco, CA

Designed and implemented over-the-air firmware update infrastructure for consumer wearable. Devised version system, messaging protocol, and algorithm capable of coping with unreliable communication channel.

Donned DevOps hat to professionalize pre-existing Lambda-based microservices architecture with continuous testing and deployment, infrastructure-as-code, monitoring, and error reporting.

**Lead Software Engineer** 2016 - 2018
Kalo San Francisco, CA

Led team which built “Kalo Pay” freelancer payment system which produces the greater portion of Kalo’s revenue. Worked extensively with third-party payment processors to assemble seamless international support.

Deleted 12% of codebase in a week by extricating email notification code from core business logic and into an event-driven microservice, reducing developer pain and increasing productivity.

Created take-home assignment for engineering applicants and conducted pair-programming interviews with candidates.

**Senior Software Engineer** 2008 - 2016
Roon Labs (f.k.a Meridian Audio, Sooloos) San Francisco, CA

Created world’s most extensive music metadata database by leveraging parallel processing to retrieve, normalize, analyze, and cross-reference all leading music data sources. Designed identification service capable of turning a mislabeled Napster mp3 into a portfolio of metadata and relationships (images, artist biography, local tour dates, etc).

Simultaneously built and supported Mac, Windows, iOS, and Android native application ports by quickly prototyping a web-based UI (later replaced with more performant GL) to maximally leverage code sharing.

Resolved circular dependency between iOS and field hardware with a protocol negotiation service and discreetly loaded interpreted code on iOS. Exposed Obj-C messaging to embedded script engine and built UIKit bindings by hand.

**Software Engineer** 2006 - 2008
Bloomberg L.P. New York, NY

Freed 1,200 developers from dependence on legacy (FORTRAN, circa 1982) date/currency string utility by reverse-engineering original FORTRAN, designing a format grammar, and implementing an extensible plug-in architecture in C++.

## EDUCATION

**B.S. Computer Engineering with Highest Honor** 2004
Georgia Institute of Technology Atlanta, GA