# Steven Brunwasser

http://brun.ws steve@brun.ws (845) 282-0626 210 Harrison St Apt 324, San Francisco, CA 94105

## Experience

### Nextdoor — San Francisco, CA

iOS Software Engineer — May 2018 – June 2019

› Created a custom, reusable, unit-tested UI to allow a user to select image and video assets to attach to a post.
› Refactoring the infrastructure used within the app to display posts in the news feed to migrate away from neglected Objective-C code in favor of Swift, and to use more appropriate native tools and patterns—such as using UITableViewCell reuse to improve scrolling performance, and using the data source and delegate patterns for large reusable components.
› Mentoring developers new to the iOS platform to get them familiar with various Cocoa patterns and APIs, as well as developer tools including Xcode, Interface Builder, Instruments, and the Accessibility Inspector.
› Helping developers manage interoperability between the Swift and Objective-C runtimes—including debugging edge-cases when bridging between the two languages, understanding the limitations when using both languages within a framework or unit test target, and identifying the different semantics and APIs between the two.
› Educating developers and project managers about the importance of accessibility, and the various APIs available in Cocoa.

### DoorDash — San Francisco, CA

iOS Software Engineer — March 2017 – October 2017

› Developed techniques to incorporate unidirectional dataflow architectures into the traditional iOS app architecture to improve testing, maintenance, and code reuse.
› Educated iOS developers about software design patterns, antipatterns, and functional programming to help them proactively spot code smells and preemptively develop solutions.
› Extracted duplicate functionality into shared, unit tested frameworks to maximize reliability while minimizing the code's overall surface area.

### Google — Mountain View, CA

iOS Hands Free Software Engineer — January 2016 – March 2017

› Implemented frontend and backend features for the Hands Free app on iOS.
› Refactored out singletons in favor of dependency injection to better facilitate testing.
› Introduced reactive programming tools, such as event streams, into the code base to more directly indicate how components react to each other. This also allowed for more straight-forward unit testing, as it reduced the need for mock objects and notifications.

### Apple — Cupertino, CA

iOS Contacts Software Engineer — January 2015 – December 2015

› Implemented features and bug fixes in the Contacts app for iOS 9, as well as the new Contacts framework.

iOS Spotlight Search and CloudKit Intern — June 2014 – December 2014

› Contributed to bug fixes in Spotlight Search, Springboard, and UIKit for iOS 9.
› Created performance testing infrastructure for the new CloudKit in iOS 8.

UIKit Intern — June 2013 – August 2013

› Developed a proof-of-concept debugging tool to visualize and inspect an iOS app's view hierarchy in real time.
› Leveraged Objective-C metaprogramming to implement boilerplate code at runtime, which more easily facilitated the view debugging tool.

UIKit Intern — June 2012 – August 2012

› Created a proof-of-concept testing infrastructure to detect animation regressions with device rotation and other view transitions.

# Education

## Rochester Institute of Technology — Rochester, NY

Bachelor of Science in Computer Science — May 2014
Minor in Deaf Culture Studies

# Projects

## Bibliotek — https://github.com/stevebrun/Bibliotek

› An Objective-C library for communicating with library databases using the Z39.50 protocol, powered by the YAZ toolkit.

## Reggie — https://github.com/stevebrun/Reggie

› A Swift library for creating Finite State Machines, such as Non-Deterministic Finite Automata and a Push-Down Automata, which can be used to create functions that pattern match strings like regular expressions.

## SwiftOFX — https://github.com/stevebrun/SwiftOFX

› A Swift library for reading financial information from OFX files exported through banking institutions' web portals.

## Cobbler — https://github.com/stevebrun/Cobbler

› A C library of macros that can be used to form a primitive dynamic object system inspired by the Objective-C runtime.

# Knowledge

## Advanced

› Objective-C ◦ Swift ◦ C ◦ C++
› Functional Programming ◦ Reactive Programming ◦ Unidirectional Dataflow
› Object-Oriented Design Patterns ◦ Software Antipatterns

## Proficient

› C# ◦ F# ◦ Haskell ◦ SML ◦ Go ◦ Ruby ◦ Python ◦ JavaScript
› American Sign Language (upper intermediate)

## Familiar

› Programming Language Theory
› Human-Computer Interaction ◦ Graphic Design
› Library Science ◦ Linguistics ◦ German (elementary)