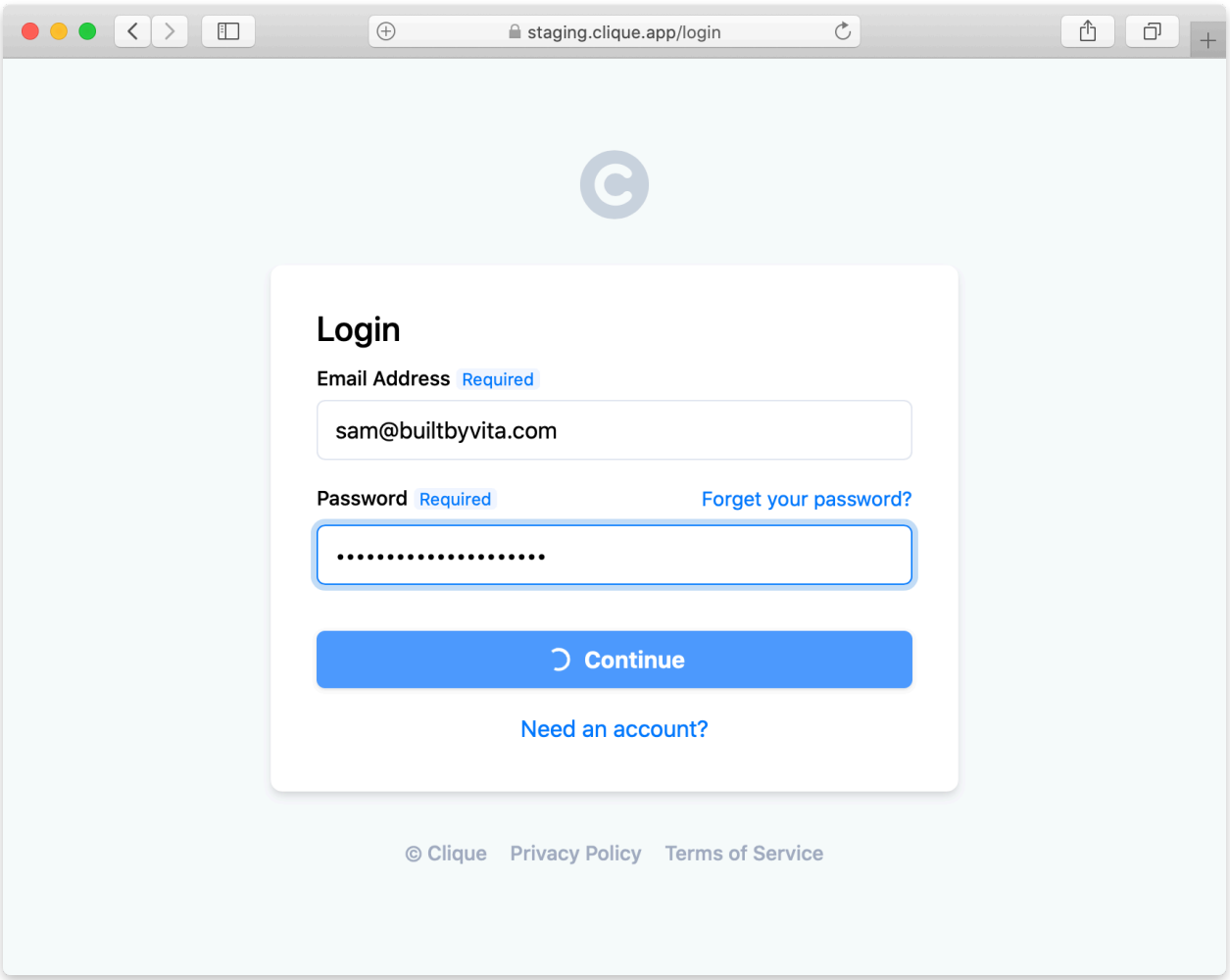
SAMUEL ROTH sam@builtbyvita.com



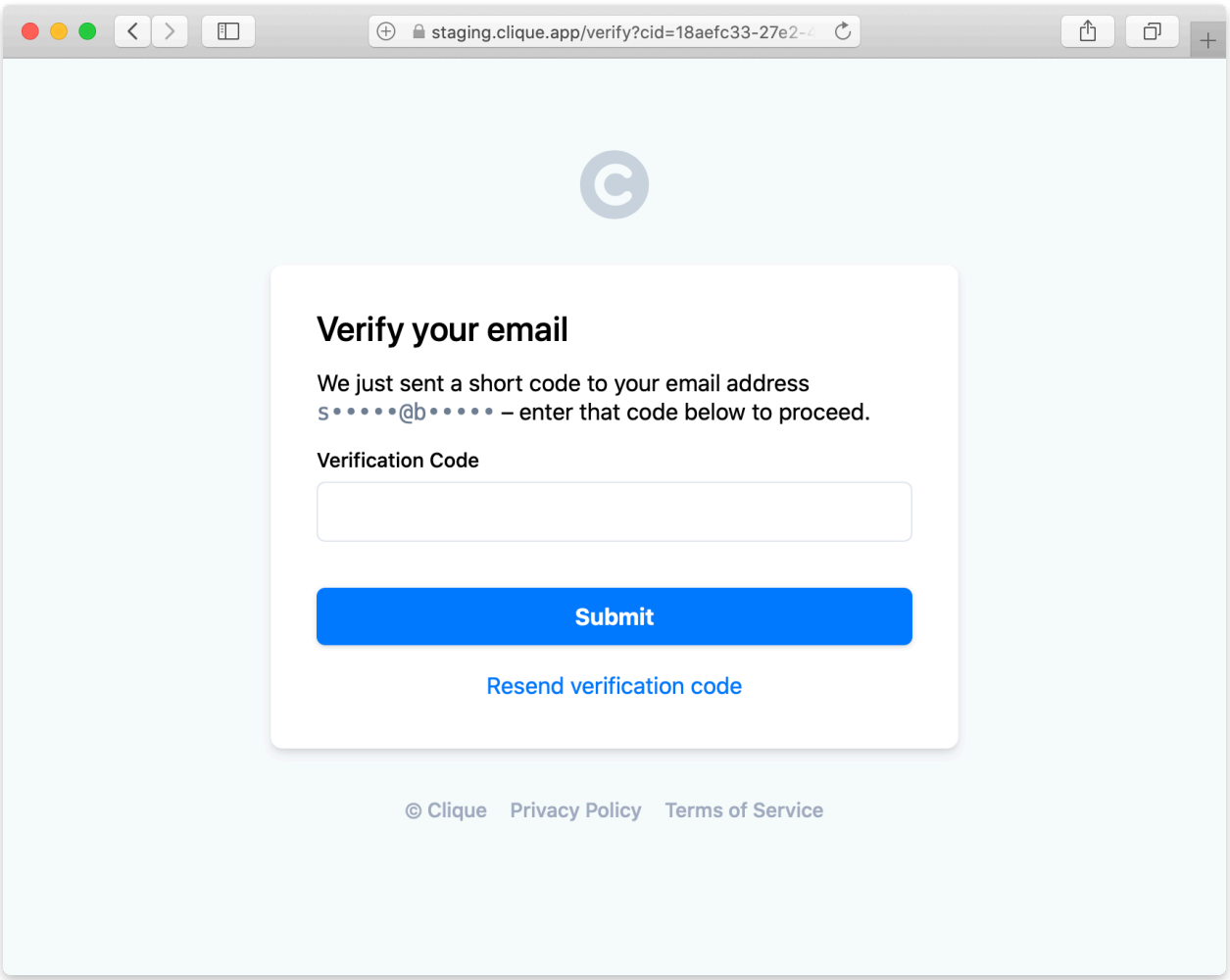
Login screen for Clique, a private social network.

Building Compose, a design system for Vita services

This and other screens from Clique are built using an internal framework called Compose. Being a solo founder and developer, I found myself spending a lot of time duplicating various interface elements across the many iterations of Clique's MVP. I've saved myself a substantial amount of time and effort by slowly extracting and refining components into a separate NPM package. Every component is strongly-typed, extensively tested, and built with accessibility in mind.

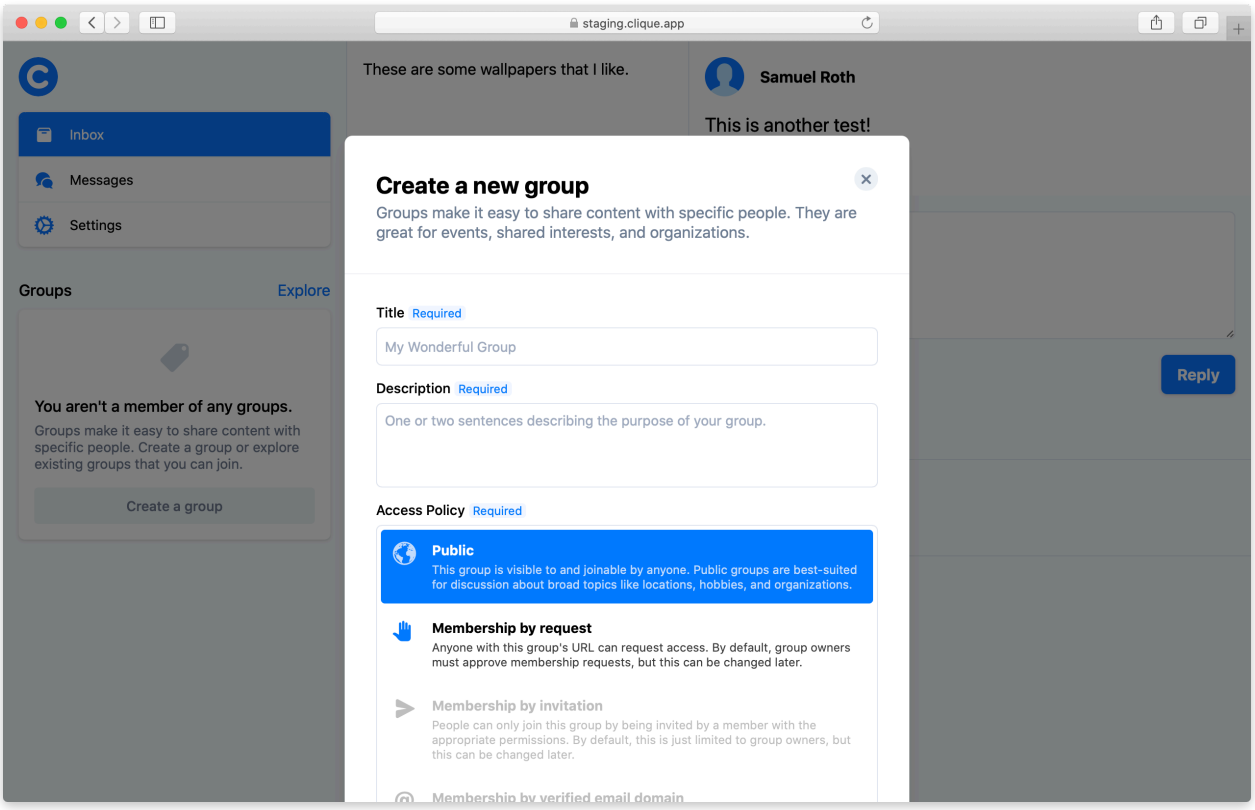
The web implementation of Compose is built with TypeScript and React, and I'm actively working on a SwiftUI port for Clique's iOS application.

SAMUEL ROTH sam@builtbyvita.com

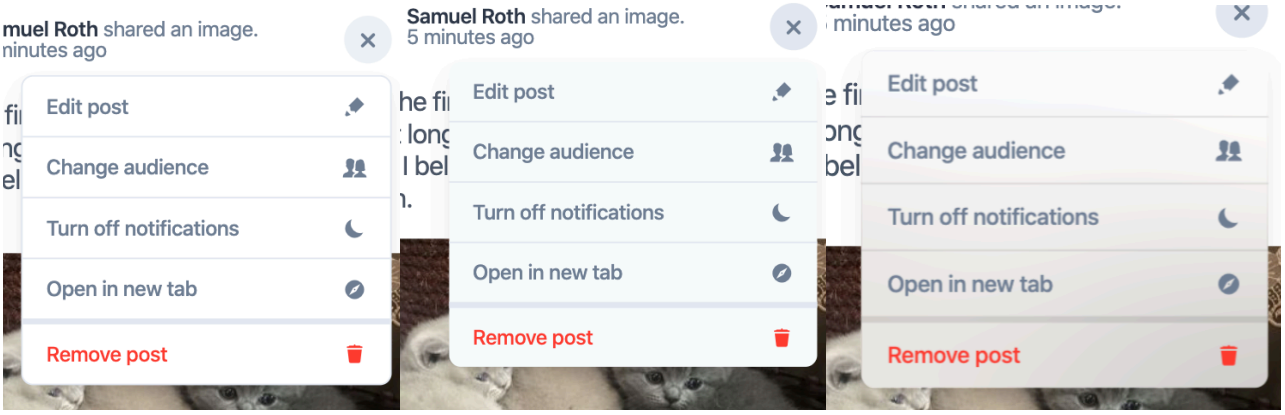


Multi-factor authentication screen for Clique; supports email and SMS verification. Web Authn in progress.

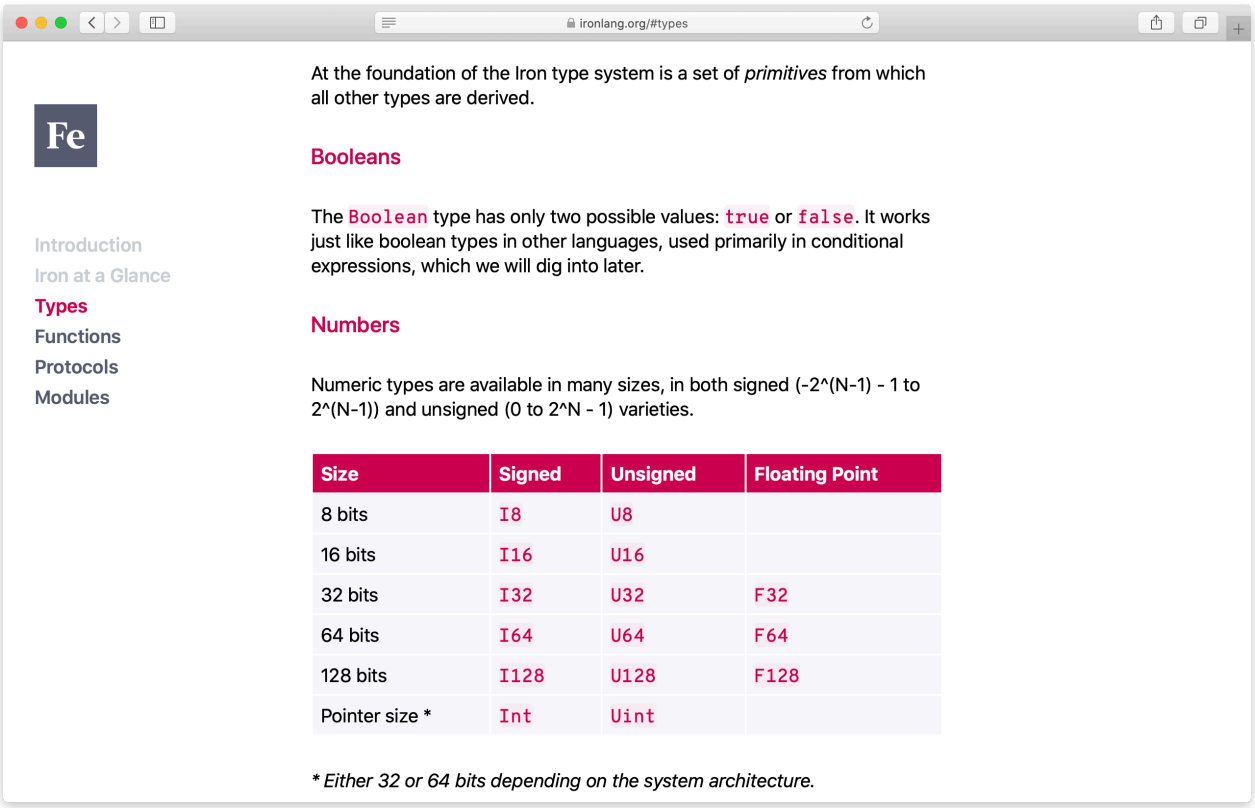
SAMUEL ROTH sam@builtbyvita.com



Example modal screen, using the same Compose input elements from the authentication flow.



Experiments with popover menus.



The website for the Iron programming language.

Building documentation for a new programming language

When someone visits the website for a programming language they want to learn, resources to do so are often scattered through many pages, potentially on separate websites. When it came to building the official website for Iron, a programming language I work on in my free time, there are plenty of frameworks consider. Most do a good job of converting a series of Markdown files into a browsable website, but I wanted something different. As a result, I'm building a new documentation platform from scratch, on top of NextJS.

From the collection of Markdown files I have, the Iron documentation is compiled into a single web page that will cover everything including installation, syntax, and even common recipes (e.g. web services). All the user will have to do is scroll.

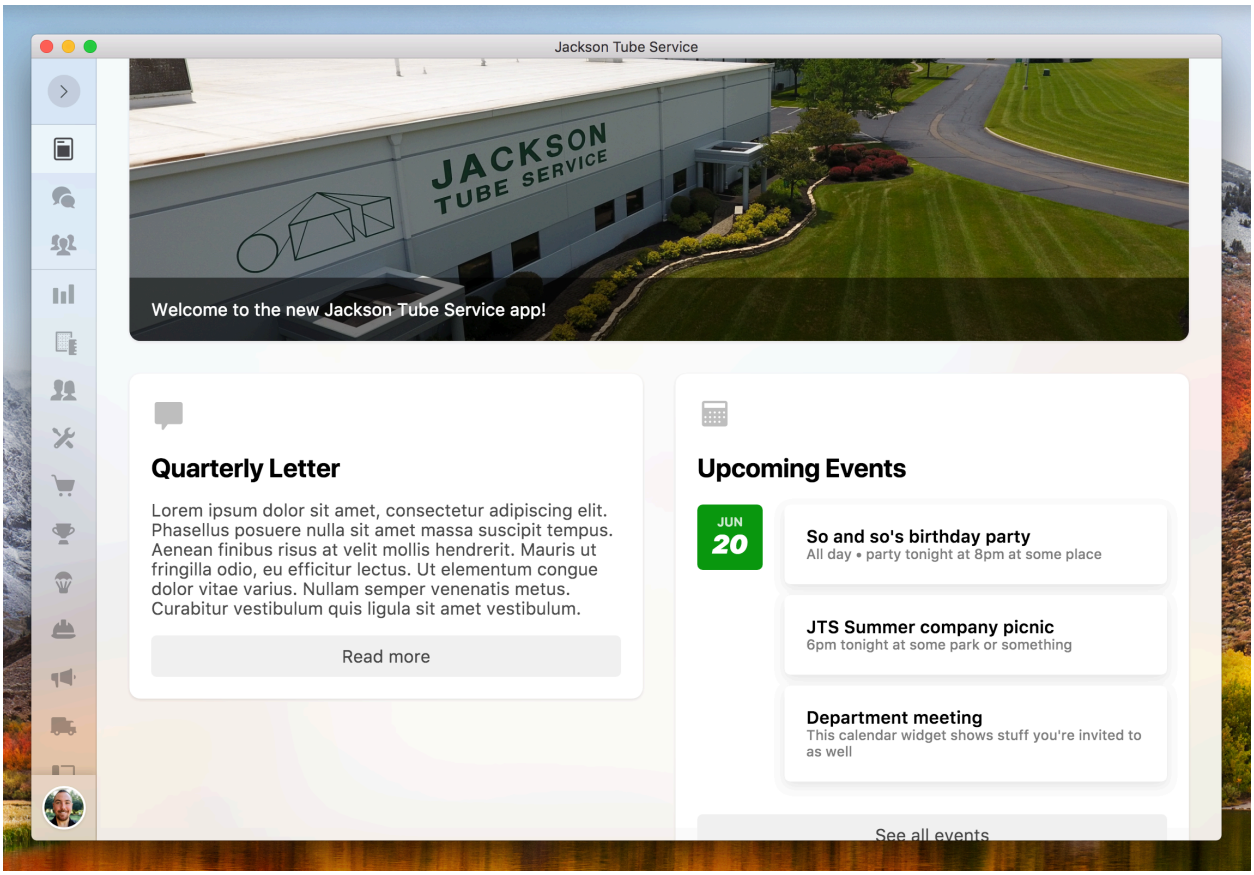
The goal is to make this website installable as a PWA, or printable, so that Iron developers can have the language documentation in its entirety in one place.

SAMUEL ROTH sam@builtbyvita.com

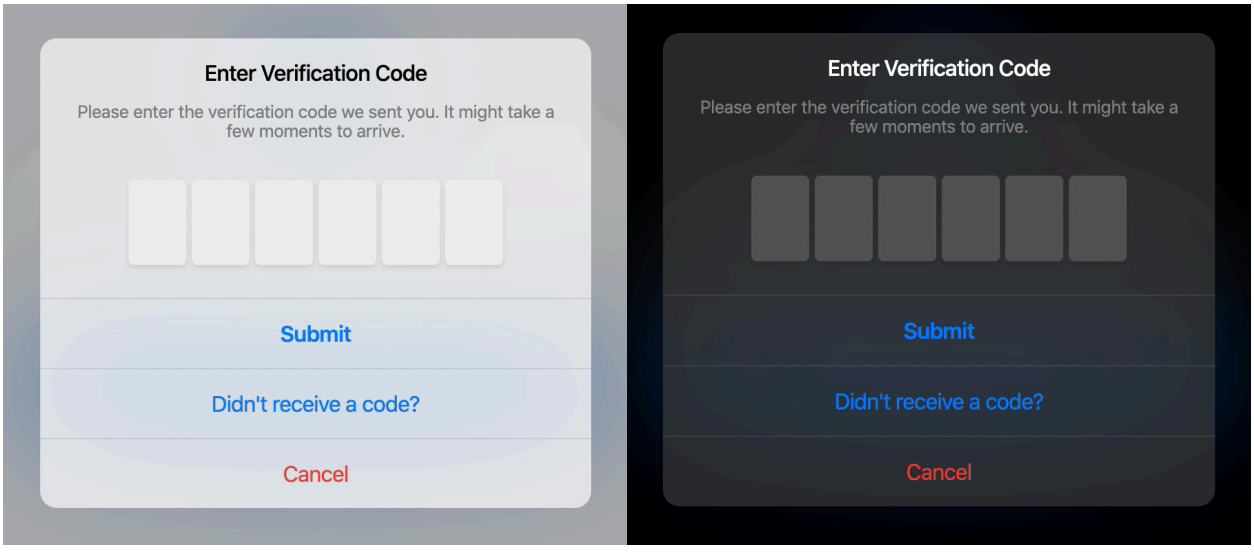
Screens from a separate project in the healthcare space.

Exploratory icons for that project's macOS application.

SAMUEL ROTH sam@builtbyvita.com



Proof of concept for an Electron desktop application for Jackson Tube Service, a steel tubing manufacturer.



Exploratory UI for multifactor code input (React + TypeScript).