

# Jake Hendley

Passionate Software Developer with 5+ years experience building offline-first mobile apps, cutting edge APIs, and scalable infrastructure. Dramatically improved developer processes by creating a knowledgebase and internal frameworks at Digital Scientists. Ensured client satisfaction by providing insight and training.

## EXPERIENCE

### Digital Scientists — Senior Developer

October 2018 - Present

As a Senior Developer, I continued to build scalable APIs and mobile applications, built Solution Architecture documentation, and provided leadership and advice on many different projects. I implemented serverless architecture, low-level bluetooth communication, and robust testing environments.

- Planned and built platform architecture for clients.
- Built offline-first Android and iOS applications using React Native.
- Implemented Bluetooth communication for IOT products.
- Orchestrated User Analytics implementations.
- Built real-time infrastructure using web sockets and Redis.
- Built heavily-tested APIs for consumption by Web and Mobile applications.
- Built granular estimates for projects based on wireframes and designs.
- Automated deployment of native applications.
- Implemented Elasticsearch as a caching and search layer, and built an automation layer for data syncing.
- Managed infrastructure and deployment of products.

### Digital Scientists — Developer

July 2017 - October 2018

As a Developer, I built cutting edge APIs in Node and Ruby, robust admin implementations in Rails, and offline-first mobile applications in React Native and Ionic. I implemented Elasticsearch for multiple clients, handled deployment to AWS and GCP, and managed the App Store submission processes.

### Arkiver — CTO/Lead Developer

July 2014 - July 2017

As the first hire, I built Arkiver from the initial idea to a cross-platform application consisting of a robust API, responsive Web frontends, and iOS and Android apps.

### The Iron Yard — Teaching Assistant

January 2014 - May 2014

### Freelance — Consultant

August 2013 - Present

101 Reedy View Drive  
Greenville, SC 29601

(864) 640-9515

[jakehendley.com](http://jakehendley.com)

[jhendley25@gmail.com](mailto:jhendley25@gmail.com)

<https://github.com/jhendley25>

## TECHNOLOGY EXPERTISE

**Languages** Javascript, Ruby

**Frameworks** React, React Native, Express, Rails, Socket.io, Sequelize, Angular, Ionic

**Infrastructure** AWS, GCP, Elastic Beanstalk, App Engine, RDS, EC2, Elastic Cloud

**Data** Elasticsearch, Postgres, Redis, Flux/Redux, SQL

**Tools** Git, Cocoapods, Gradle, Vim, Fastlane, Xcode, Android Studio

**Processes** Image processing, Video transcoding, data federation, Geodata analysis, Financial automation

## PROJECTS & ORGANIZATIONS

### Openworks - Greenville, South Carolina

December 2014 - Present

Grew the organization through marketing campaigns, designing the upfit of the space, and organizing community events.

### CATE advisory board - Greenville, South Carolina

January 2015 - Present

Provided industry-specific insights and curriculum advice to the Career and Technology Education board for South Carolina schools.

## Case Studies

Over my career, I have built or contributed to more than a dozen native apps, APIs, and web front ends. The following are a few examples of these applications and the problems they solve.

### E-Commerce Platform

**Business Problem** The client needed the ability to conduct transactions, make real-time reservations, administrate sellers, and reconcile financial data.

#### Associated Hurdles

- Millions of items available for sale with complex associations between users, meaning administration was needed on a granular level.
- Scale at thousands simultaneous requests, meaning a robust caching layer was needed.
- Organizing and validating payouts between sellers and vendors with various levels of financial differences, meaning

**Solution** A large distributed platform consisting of an Ionic consumer application & web frontend, NodeJS API, Rails Admin, Elasticsearch caching layer, and websocket real-time layer.

### Media-Heavy Offline Application

**Business Problem** The client needed an application to interface with their existing image analysis software.

#### Associated Hurdles

- Used in rural areas, meaning offline-first design was needed.
- Media-heavy, meaning image processing and resilient uploading.

**Solution** A React Native App using native image and file system libraries to achieve offline-first media management in concert with a Node API.

### Offline QR Code Ticket Scanner

**Business Problem** The client needed the ability to scan QR codes on mobile devices and printed tickets via a Sled-based scanner with connected iPod.

#### Associated Hurdles

- High attendance at events leads to clogged networks, meaning the scanner cannot reliably connect to an external API.
- Tens of thousands of tickets that may or may not be valid for a given event, meaning too much data to store in device runtime memory.
- Reliance on third-party hardware (the Sled) to allow for easier scanning and device charging, meaning the app must implement functionality from a custom SDK.

**Solution** An offline-first React Native App which interfaces with the Sled via a custom SDK, and implements a SQL-like local database. The result was an application that could download gigabytes of data ahead of time from an elasticsearch layer, intuitively interface with the connected hardware, and intelligently sync ticket scans based on network connectivity.