JARED ROTHENBERG

SOFTWARE ENGINEER

https://www.linkedin.com/in/jared-rothenberg www.Jaredar.com

Los Angeles (CA) +1 (310) 880 0109 Jaredar@gmail.com github.com/jaredar1232

EDUCATION

HACK REACTOR 2020

Advanced Software Engineering Immersive | 1000+ hours | Computer Science Fundamentals

UNIVERSITY OF CALIFORNIA, SANTA BARBARA 2019

B.A. Economics, Philosophy

3.5 GPA

NATIONAL UNIVERSITY OF SINGAPORE Aug. - Dec. 2019

Study Abroad

3.9 GPA

EXPERIENCE

LIVE NEWS DIRECTOR

Happs Sept. - Nov. 2019

(crowd-sourced news startup)

- Composed a single livestream from multiple live feeds
- Onboarded new contributors

SALES & SERVICE REP

SOL Boards Inc. 2016 - 2019

- Rebuilt and reprogrammed broken equipment
- Hosted product launch events

EXTRACURRICULAR ACTIVITIES

MAYBANK HACKFEST 2018

- Singapore finch hackathon
- Finalists before C-level judges

PHI THETA KAPPA 2015 - 2017

Commissioner

- Co managed local chapter of an internationally recognized honors society
- 300+ local members & 3,000,000+ global members

TECHNICAL SKILLS

Languages: JavaScript (ES5/6+)

Front End: HTML5 \ CSS3 \ React \ Redux \ Styled Components \ Chakra UI \ Joker.js

Back End: PostgreSQL \ MongoDB \ Mongoose \ Firebase \ MySQL \ Node \ Express

Tools / Deployment: Git \ npm \ Webpack \ Babel \ AWS \ Gatsby \ Netlify \ Heroku

Testing: Jest \ Mocha \ Chai \ Loader.io

APPLICATIONS

NIKE PRODUCT DISPLAY PAGE MOCK-UP

- Constructed a Nike product display page and followed a micro-service architecture to keep concerns separate and modular
- Generated 5000+ random reviews for 100 real nike products with data persistence in order to simulate a realistic user experience
- Queried a Mongo database with nested subdocuments by implementing CRUD operations and a custom API to give the user appropriate information
- Rendered a single page comprised of 3 micro-services that communicate via url change and proxy server

SYSTEM DESIGN

- Expanded data set from 100 unique records to 10,000,000 unique records
- Reduced database migration time by 80% by using automated CSV generation and batch import (~12 min down to ~2.5 min)
- Optimized query times between Mongoose ODM, MongoDB, and PostgreSQL and by indexing to reduce queries to an average of under 0.1ms per request
- Deployed independent database and servers; scaled from 1 EC2 instance with 2000 rps, 0% error rate, and 126ms latency up to 3 NGINX load balanced EC2 instance servers with 3000 rps, 0% error rate, and 63ms latency

STREAK BASED ACTIVITY TRACKER

- Implemented a monolithic architecture with git feature branch workflow to produce a more uniform codebase
- Utilized Firebase for authentication, adaptable storage, and session state persistence to maintain full features while under time constraints
- Practiced mobile first emphasis to ensure the final application didn't sacrifice key features when displayed on mobile

CUSTOM PORTFOLIO

- Increased site performance by converting to Gatsby server side rendering and reducing CSS animations when mobile media query breakpoints are reached
- Ensured stable media sources by hosting all content on private AWS S3 bucket
- Secured Sendgrid API by building a separate server that interfaced with my pre-rendered client