(818) 522-1526
sasha@helveticalake.com

Sasha Khadivian

sasha.helveticalake.com
linkedIn.com/sashakhadivian
github.com/sashakhad

Skills

Languages: Bash, HTML/CSS, Javascript, Ruby, Typescript
Frameworks: Nestjs, Nextjs, Nodejs, React, Ruby on Rails,
Databases: PostgreSQL, MongoDB
Platforms: Asana, BitBucket, Confluence, Contentful, GitHub, GitLab, Jira, MailChimp, SendGrid, Slack, Stripe, Teams, Trello, Vercell
Tools: Bootstrap, Cypress, Docker, ESLint, Express, Git, GraphQL, Jest, Less, Material UI, Mobx, OAuth, Puppeteer, React Router, Redux, Sass, Storybook, TailwindCSS, TSLint, Webpack
Devops: Amazon S3, AWS Lambda, Cloudflare, Cloudfront, Google Cloud Run
Spoken Languages: English, French, Persian, Portuguese, Spanish

Professional Experience

Lead Software Engineer - Helvetica Lake Consulting   Remote | 2020 - Present
Guardian Life
Tech: React, Nextjs, Nestjs, Prisma

- Developed a comprehensive payment solution by architecting controller, service, repository, DTO, and entity responsibilities in Nestjs, enabling effective payment processing using a singular endpoint using Payment Intent and Customer Balance Transaction APIs
- Reengineered the Stripe checkout pipeline, shifting from client-side Payment Intent creation and webhook integration to a server-side approach, improving payment and donation functionalities via a proprietary credit system
- Led initiative to rebuild Admin portal CMS using React, Nextjs, and Prisma for application administrators to modify user permissions, budgets, and settings
- Transitioned broken authentication flows from page-by-page serverSideProps auth to NextJS Middleware JWTs and Okta
- Established scalable component and design systems with accessibility features using React, Radix, and Storybook
- Analyzed state of core product's front-end architecture and crafted multi-phase Jira epics to (1) resolve systemic responsiveness issues; (2) standardize application error handling; (3) centralize design system values into single source of truth; (4) replace Material UI with custom component library using Radix and TailwindCSS
- Integrated and assigned detailed tickets to team members in larger efforts to implement authentication fixes, deprecate MUI, and rebuild admin panel
- Optimized API endpoints to manage client requests and handle database operations
- Configured content models using Contentful, ensuring a structured and scalable approach to content management workflows for non-technical team membersInitiated and led program to transition from Material UI to Tailwind to enhance code maintainability and responsiveness
- Mentored a team of software engineers, promoting best coding practices and methodologies and driving the delivery of high-quality code, ultimately improving the team's productivity and software reliability

Reforge
Tech: React, Typescript, GraphQL, Ruby on Rails

- Translated design system into an organized a suite of reusable, type-safe React components using Storybook
- Created end-to-end unit and integration tests for features and flows using Jest and Cypress
- Updated Rails data models using Active Record and model validation utilities
- Defined GraphQL types using Apollo to enable CRUD operations for Rails model
- Routed application flows with React Router and url parameters to manage various user states
- Advised mid and junior engineers, assisting the team to increase its overall velocity sprint over sprint

FlyFlat
Tech: React, Nextjs, Prisma, Vercel

- Designed and executed CMS dashboard for adding, updating, and archiving flights for admins to manage their products
- Engineered Stripe checkout flow using Stripe Elements to collect customer payment details, establish authorization and capture flows, and process payments
- Defined relational database schemas using Prisma ORM and hosted PostgresQL database on Heroku
- Coded authentication management patterns and login flows using NextAuth and Next Middleware functions

- Developed email templates and send schedule pipelines using SendGrid
- Programmed mailing list subscription module by integrating the Mailchimp API
- Established Agile SOPs for feature development, bug reporting, post-mortems, and retrospectives

Software Engineer - *Airbnb*   San Francisco, CA | 2016 - 2020
Airbnb-friendly Apartments
Tech: React, Typescript, Ruby on Rails
- Leveraged React hooks and RESTful API requests to organize multi-step verification features in two-sided marketplace for professional hosts and building owners
- Enabled progressive enhancement features using ERB templates, thereby increasing rendering speed in app by 10x
- Tested end-to-end features using automated CI, Jest, Cypress, and Happo
- Converted 100+ files to be type-safe using Typescript, working with cross-functional teams to establish Typescript best practices throughout the organization

Urbandoor
Tech: React, Typescript, Redux, Mobx
- Created state management patterning system for Redux and Mobx page states built for React components
- Iterated on scalable CSS systems, implementing CSS modules, inline typed CSS, and global style sheets
- Exercised end-to-end integration tests to mock user behaviors and UX flows using Capybara
- Conformed codebase to Redux best practices to optimize team conventions and collaboration
- Refactored Redux codebase into Mobx to employ a pattern of external state ecosystem
- Created a modularized, scalable system for mocking data for unit tests using Typescript namespaces for data models
- Leveraged with Google Maps API to display custom pin UIs and hover content on Map
- Integrated Storybook to write isolated unit tests for collaboration between engineers and designers

Front-End Engineer - *Apple*   Sunnyvale, CA | 2015 - 2016
Tech: React, Javascript, Redux, Webpack
- Architected React and Redux application that communicates with Tomcat and Apache servers issuing SOLR queries
- Built a scalable styling system using Sass and CSS Modules, enabling modular and consistent styling
- Implemented Mocha, Chai, Enzyme test suite to unit-test UI and state behavior
- Built modular and dynamic components serving as the foundation of the application's UI
- Designed NPM QA and IT deployment scripts that reduced deployment time by over 10X

Projects
Real-Time Awards Flight Pricing Engine - *Easypoint*
Tech: React, Typescript, Nextjs, Prisma, Puppeteer, AWS
- Developed real-time pricing engine converting award flight points to dollar values using NextJS, Express, and Puppeteer
- Scripted Puppeteer to scrape up-to-date flight information from airline websites like Google Flights and United Airlines
- Maintained multiple deployment environments via Docker on AWS, Google Cloud Run, and Vercel
- Deployed serverless Express instances to AWS Lambda to handle application requests to prompt scrape requests

NSF Codebase Migration: Ember to React - *National Science Foundation*
Tech: React, Ember, Typescript
- Formulated comprehensive plan to to migrate Ember codebase to React, Typescript, TailwindCSS, and NextJS
- Led cross-functional technical design discussions on application architecture, UI components, UX, back-end and third-party integration, and testing.
- Debugged user dashboards for managing NSF grant data for researchers and adjudicators
- Handled legacy feature development using Ember Octane and Handlebars

CryptoCurrency Payment Calculator - *Easypoint*
Tech: React, Typescript, Nextjs
- Launched interface for converting flight prices to 10+ cryptocurrencies, enabling users to quickly transfer funds from their wallets to make payments
- Integrated Coin Market Cap API to convert ticket prices from fiat currencies to up-to-date cryptocurrency values
- Connected payment data from client website using url query parameters to automatically populate desired calculation