

Stephen Szelpal

www.stephenthe.dev.com

+1(216) 744-5802 - stephenszelpal@icloud.com

Technical

Python - JavaScript - React - SQL - REST API - HTML - Redux - Flask - Node JS - CSS - Express
- GraphQL - AWS - React-Native - Unit Testing - End-to-End Testing - GoogleMaps API

Education

Full Stack Software Engineer - 2020

App Academy - San Francisco, CA

Coursework: Software Development & Design, Computer Science, Front/Back-End Engineering

Bachelor of Fine Arts - 2009

Cleveland State University (degree unfinished) - Cleveland, OH

Coursework: Theatre Arts, Theatre Design, Acting for the Camera

Experience

Software Engineer - Mar 2021 | current

Transition Studios (contract) - Cleveland, OH

- Conceived and executed development for a web/mobile video streaming application
- Developed in Typescript, for the frontend, used an Expo build and for the backend, AWS Amplify, utilizing React-Native and GraphQL
- Integrated 10,000 subscribers

Actor - Nov 2018 | Feb 2020

Television & Film - New York City, New York

- Committed 12+ hrs each day for production, most of which ran much longer
- Composed and maintained strong relationships with casting directors and producers without agency representation

Owner / Operator - Apr 2013 | Feb 2016

Ohio City Dog Haven - Cleveland, OH

- Instantiated a brick and mortar LLC as sole-proprietor and produced an average annual revenue of \$90000.
- Negotiated contracts and pricing with distributors and manufacturers as well as consignments for local artists.
- Utilized QuickBooks for business accounting.

Lead Bartender - Mar 2015 | Feb 2020

Barrio - Cleveland, OH

- Remodeled the customer rewards program, resulting in an eventual \$1000/wk increase in bar sales.
- Cultivated a large group of regular customers, a few turned into longtime friends.

Projects

[Seen Scenes](#) - Javascript - Express - Pug - CSS - PostgreSQL

Online film database application inspired by Goodreads - [GitHub](#)

- Developed with a team of three engineers.
- Overhauled the first iteration of CSS by eye to replicate Goodreads design.
- Implemented a persisting review feature that allows users the ability to write one review per film.

[Journal Easy](#) - Python3 - Flask - React - Redux - PostgreSQL - CSS

One page journaling application - [GitHub](#)

- Developed full CRUD on the users, entries, and ratings features.
- Created front end constraints utilizing the Redux store, limiting the amount of daily ratings or journal entries a user can submit.
- Formulated a GET request on the users dashboard that accesses the redux store iterating and correlating the users last 30 daily ratings with each ratings respective dates on a line graph. Instead of making a backend API request for the last thirty dates, I created a data set pulling the most recent 30 ratings, iterating backwards, and applying a new date object as each key, decrementing each day with that obj built-ins.

[LinkedIn](#)

[GitHub](#)