Evan James Martin

CONTACT INFORMATION

phone: +1-604-816-6682
email: evan.martin@shaw.ca
website: evmarts.surge.sh

2966 West 36th Ave, Vancouver, BC, Canada

EDUCATION

B.Sc. Computer Science and Statistics, University of British Columbia, Vancouver campus. Expected graduation May 2019, 3.30 GPA. View my academic background here .

SKILLS

Highly skilled and proficient in: Python, JavaScript, SQL, MATLAB, R Statistical Software
Proficient in: C, C++, Java
Frameworks/libraries/tools: Node.js, Express, React, Knex.js, Heroku, urllib, NumPy, RegEx, Git, Adobe Illustrator, OpenCV, Python Imaging Library (PIL)

EXPERIENCE

Back-end Engineer Intern at Axiom Zen (CryptoKitties Team) (May 2018 - August 2018)

- Synopsis: Developed the back-end for the Kitty Collections feature, which included CK's first **CRUD API**. This feature allows users to create, update and delete groupings of their kitties into sharable collections. Developed several REST endpoints to supply lists of kitties to different pages of the website, including the Catalogue page.
- Challenges: The responsibilities of interns at Axiom Zen was quite high and development moved fast. From this, I learned that proactive communication with team members is key. Understanding how the CK back-end worked with the **Ethereum blockchain network** was also difficult at first. To improve my understanding, I would schedule weekly meetings with senior engineers to review the CK back-end.
- Outcomes: Learned how to manage myself in a fast-paced environment with high responsibility. Developed an understanding of the importance of team communication. Gained a deep understanding of how the Ethereum blockchain network works and the costs and benefits of building an application on top of it.

PROJECTS

Projects available at github.com/evmarts

Employment and Social Development Canada (ESDC) Occupation Search Tool (Full-stack Web Development), see web app

- Synopsis: Built a website that allows users to search for occupations based on their personal aptitudes and characteristics. The back-end includes a **PostGres** database and an API written with **Express**, both hosted on **Heroku**. The front-end is written with **React**.
- Challenges: This was the first time I had built a distributed application with a front and back-end from scratch so understanding what was needed at each next step was a challenge. I ended up doing a lot of research on how modern web-applications are constructed and was able to work my way to a final product that implements those modern technologies.
- Outcomes: Website is used by the client multiple times a week. Currently seeking to extend the service to multiple clients. Learned important elements of **front-end development** and **UX design**. Gained a macro-level understanding of how modern web-applications are built and how all the elements work together.

Instagram Account Growth and Marketing (Internet Marketing + Full-stack Web Development)

- Synopsis: Grew six niche Instagram accounts via **web-scraping**, data-analysis and **image generating techniques**. Became affiliated with businesses and advertised their products via the account. Developed a web app to facilitate user engagement contests by tracking user engagement and awarding points on a leaderboard system based on those engagements (built with **Express**, **React**; hosted on **Heroku**, Surge).

- Challenges: Spent a lot of time on finding content to post to my accounts, solved by automating content generation. Spent a lot of time searching for users to attract to my accounts, solved by scraping the usernames of users who interact with other popular accounts in my niche. Struggled with user engagement drop, solved by creating user engagement contests.
- Outcomes: Grew a **200k+** network of followers spread amongst multiple niches. Further strengthened full-stack web development skills. Learned basics of Internet Marketing.

Internet Meme Generator (Python programming), see blog post

- Synopsis: Applied **OpenCV** and **Tesseract OCR** with **PIL** to break an Internet meme into cropped image and string of text. The program is then able to compose a recycled version of the internet meme in a cleaner format than the original.
- Challenges: Needed to be able to dynamically crop an image to remove white borders, solved by using OpenCV to recognize the boundaries of the largest rectangular component in an image. Needed to be able to recognize images of text as strings of text, solved by integrating Tesseract's optical character recognition engine.
- Outcomes: This project gives one the ability to take a collection of memes and recycle them into a cleaner format. It may also allow one to analyze the textual and pictorial content of a set of memes, which has implications in exploratory data analysis.

MORE INFORMATION

More information about what I do can be found at evmarts.github.io/blog/