[Kevin Mok] # _

647-685-2500
me@kevin-mok.com
Kev-Mok
kevin-mok.com

Work Experience

Red Hat

May 2020 – Aug 2021

Cloud/Software Engineer Intern

- Reduced deployment time by **66%** by implementing ability to deploy locally-compiled binaries onto **Kubernetes/OpenShift** using only command-line (**GoLang** used for this and below).
- Implemented ability for Kubernetes operator to fetch data from a deployed service and update config with data.
- Added startup probes to handle starting legacy application containers that require additional startup time.
- Refactored probes to have default values assigned based on deployed YAML while also fixing reconciliation issues.
- Rewrote the **Jenkins** nightly pipeline to run in a GitHub PR using a trigger keyword.
- Wrote documentation on how to get started with the project to onboard new developers and mentored the incoming intern.

Projects

Rarity Surf

Oct 2021

- Web app to give rarity rankings to NFT's within minutes of their metadata being revealed and check which are listed (based on rarity and price filters) on the OpenSea marketplace using their API.
- Reverse engineered the ranking algorithm to match the leading rarity ranking site's rankings (scraped using Selenium) with a **discrepancy of <0.25%**.
- Used app to frontrun purchases of **top 5%** rarity NFT's against competing buyers.
- Wrote **Django (Python)** backend to fetch metadata from IPFS, store rarity rankings in PostgreSQL and serve rarity data using GraphQL.
- Wrote **React** frontend with hooks to dynamically load rarity data. Styled with Tailwind.

Astronofty

Jan 2023

- Created for a 36 hour hackathon (UofTHacks X) where it **came 2nd overall**.
- Created and deployed a smart contract with **Solidity** on the Ethereum blockchain to create/buy/sell NFT's using MetaMask.
- Wrote a **React** hook to fetch and show listed NFT's, NFT details and owned NFT's.
- Used API to synchronously upload images and metadata to IPFS.

AWS/Kubernetes/Terraform

May 2024

- Deployed various web apps using **Docker** (Compose) on an **AWS EC2** Debian/**Linux** server.
- Created **Kubernetes** manifest files to quickly recreate my server setup with persistent storage/restarts and open ports.
- Created **Terraform** files to deploy an AWS EC2 instance and Docker containers.
- Used Amazon Route 53's DNS and **NGINX** to route subdomains to each web application.
- Used AWS security groups to allow inbound HTTPS traffic.
- Used Amazon EBS snapshots to regularly back up server.
- Wrote a **JavaScript** server script and systemd service/timer to display the uptime of my pages every hour.

Skills

AWS, Kubernetes, Terraform, Docker (Compose), **JavaScript**, **React**, **Python**, Go(Lang), Bash, Solidity, C, **Django**, Node.js, Jenkins, PostgreSQL, MongoDB, **Linux**, **Git**, **Command Line**

Education

University of Toronto

2018 – 2023

Computer Science Specialist – 3.84 GPA (CS). Graduated with High Distinction.

References

See my LinkedIn for references from my Red Hat managers/mentee, a startup client and a graduate student mentor.