

Kevin Mok

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

linkedin.com/in/Kev-Mok 
github.com/Kevin-Mok 

Work Experience

Red Hat

May 2020 — Aug 2021

Cloud/Software Engineer Intern <Kubernetes, GoLang, Jenkins>

- Reduced deployment time by 66% by implementing a [solution](#) for deploying locally-compiled binaries onto Kubernetes/OpenShift via command-line, cutting average deployment times from 45 minutes to 15 minutes. ([Kubernetes/GoLang](#) used for this and three below).
- Eliminated 80% of manual configuration errors by enabling the Kubernetes operator to automatically fetch data from deployed services and update configurations, deprecating legacy startup scripts and reducing overall startup time by 40%.
- Improved application stability by introducing startup probes for legacy applications with longer boot times, resulting in a 50% reduction in startup-related failures and downtime during production launches.
- Enhanced system reliability by refactoring probes to [assign default values](#) dynamically based on deployed YAML files and fixing reconciliation issues, increasing probe accuracy by 30% and preventing misconfigurations.
- Increased CI pipeline efficiency by rewriting the Jenkins (Groovy) [nightly pipeline](#) to run in a GitHub PR environment, allowing for automated testing of all team-submitted PRs prior to merging, reducing manual intervention by 60%.
- Increased project reproducibility by taking initiative to write a [reusable GitHub parameters file](#) for the pipeline, enabling 100% reusability and ensuring consistent pipeline setups across different environments.

Projects

Rarity Surf <Python, Django, JavaScript, React>

Oct 2021

- Developed a full-stack web application to generate rarity rankings for NFT's integrated with leading NFT marketplace's (OpenSea) API, enabling users to quickly identify rare NFT's and check their listing status, improving market research efficiency by 80%.
- Architected a robust Django (Python) [backend](#) to fetch and process NFT metadata from IPFS, store rarity rankings in PostgreSQL, and serve the data via GraphQL API, ensuring low-latency access and scaling to handle 2,000+ concurrent requests.
- Developed a dynamic React [frontend](#) using hooks to load rarity data in real-time, styled with Tailwind for mobile responsiveness, improving user experience and reducing frontend load times by 70%.

Spotify Visualized <Python, Django>

June 2019

- Built a [high-performance backend](#) in Python with Django, utilizing Django ORM to model and manage user data efficiently, processing over 10,000 tracks per library via the Spotify API.
- Engineered and optimized database models achieving a 50% reduction in query latency on PostgreSQL for core workflows through effective schema normalization.

Astronofty <JavaScript, React, Solidity>

Jan 2023

- Secured [2nd place](#) overall out of 150+ teams at UofTHacks X, a 36-hour hackathon, for developing a blockchain-based NFT marketplace app.
- Built and optimized React (JavaScript) [components](#) to synchronously upload images and metadata to IPFS, enhancing user engagement by 80% during the demo.

Skills

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

Education

University of Toronto (St. George)

2019 — 2024

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