

Experience

Trait Sniper

<https://app.traitsniper.com/>

Co-founder — Sep 2021 - Nov 2022

Trait Sniper is the leading NFT rarity ranking platform. It has **2M+** monthly unique users, **100K+** NFTs bought with **100K+** ETH in volume. It is backed by a strong community with **100K+** Chrome Extension users and **30k+** users in our Discord server. I built Trait Sniper from the ground up with the following notables:

- **NFT metadata revealing system:** A reliable pub-sub task system that can handle large throughput combines with a creative proxy strategy is our recipe for fast NFT data crawling.
- **TS API server:** We built a robust and cost effective API server on AWS with a high demand for freshness and latency for our 2M+ monthly users. A simple setup of Golang API server with PostgreSQL and Redis served us well.
- **Finding cheapest NFT deals:** Collaboration with various Web3 teams/marketplaces (Looksrare, X2Y2, Gem.xyz, Blur.io, etc) enables us to find the best prices for our user.
- **NFTs analytics:** We store and process NFT data in InfluxDB to generate insightful metrics for users.
- **TS NFT collection:** Launching a NFT collection is huge collaboration effort between engineers, artists and community managers. I am responsible for the whitelist process by making sure the Merkle proof was generated correctly. I also took care of the metadata update process and metadata reveal process.
- **NFT API as a service:** By talking with various Web3 teams and potential customers, I refined and shortlisted a list of features. I then assembled a small team of 3 engineers and built the first version of TS Sass. We had 5 monthly subscribers in the first week, which is enough to cover some infrastructure cost.

AXON Enterprise

<https://global.axon.com/company/offices/ho-chi-minh-city>

Research Engineer — Nov 2018 - Sep 2021

I spent my years at AXON building an innovative license plate reading system on Fleet3 in-car camera for US police agencies. Fleet3 was officially launched in the US on 30/06/2021 for more than 15 police agencies. I am one of the earliest research engineers, which means I had to wear multiple hats from the beginning:

- **Inference pipeline on camera:** Raw image frames travel from camera sensor into the pipeline, which utilizing DSP and GPU for AI inference and image processing tasks, finally produce a plate read if it satisfies a system of thresholds. I had to bring research ideas (Python code) onto camera device (C++ code) while also be mindful about computing constraints on limited power SoC to keep the frame-rate near real-time.
- **AI models conversion:** I worked closely with researcher on delivering of AI models onto device, which means I had to take care of model conversion, model tuning and model quantization to deliver expected and good performance on SoC device.
- **Evaluation system:** I designed and implemented the evaluation system based on Azure distributed computing clusters. We splitted the dataset to run a small amount on camera, the rest is ran on x86 machines.
- **Jenkins CI job to detect performance degradation:** To better predict the performance on production, keeping the performance gap between research and production as close as possible is one of the top priorities. I built a CI job to compare outputs of research Python pipeline and production C++ pipeline per commit. So that we are notified of disparity as early as possible.

Lozi

<https://lozi.vn/>

Backend Engineer — Sep 2017 - Oct 2018

I was in a team of 3 engineers that built the back-end for Loship - a food delivery service based in Ho Chi Minh city. I was responsible for:

- **Loship automatic shipper assigning:** Built a pub-sub model based on RabbitMQ that finds the nearest shipper based on their GPS location which is stored in Redis. Once a shipper is found, the order(s) are assigned to him.
- **Server heartbeat:** Loship was using a physical node with a local ISP, so we sent CPU, RAM and IO usage to Grafana to monitor and set up alert.

Education

- **Ho Chi Minh University of Sciences**
Bachelor of Science in Computer Science

Ho Chi Minh city, Vietnam
2012 - 2016

Technologies used

- Golang, Python, C++
- PostgreSQL, Redis, RabbitMQ, InfluxDB
- Docker, Vim & Tmux, UNIX, AWS
- gRPC, Grafana, GraphQL