Curriculum Vitae

Tuấn-Anh Nguyễn

Contact Info

- Full name: Nguyễn Tuấn-Anh
- Mobile: +84 868 640 710
- Email: me@ubolonton.org
- Web: `https://ubolonton.org`
- Other: github, twitter, linkedin

About

I am a well-rounded software engineer/architect, currently focusing on backend infrastructure and data processing pipelines. My previous education as a physicist gives me a highly analytical and critical mindset when solving problems, whereas a large portion of software professionals tend to "throw everything at the wall to see if it sticks", relying primarily on "experience" and skill set accumulation.

In many projects I participated in, I was responsible for working directly with stakeholders to analyze the business needs, coming up with overall system designs, building end-to-end solutions, deploying and maintaining systems through multiple iterations. I tend to favor solid designs with deep thinking over quick-and-dirty solutions. I pay a lot of attention to details, **learn extremely fast** while also **conducting in-depth research**, and have **discerning tastes** when it comes to evaluating technologies and ideas, even before they become well-known.

I am interested in real-time/distributed data systems, productivity tooling/practices for programmers and data scientists, systems programming, security, language theories (both human and computer).

Knowledge and Skills

- Designing system architecture.
- Analyzing, troubleshooting, optimizing systems.
- Business analysis, requirement analysis.
- Strong product mindset.
- Linux administration, maintenance, automation.
- Managing and coaching technical teams.
- Data science fundamentals, machine learning operationalization.
- Technical writing.
- Languages: English, Vietnamese, Russian.

Tools and Technologies (Sampled)

- Languages: Python, Rust, Scala, ELisp, JavaScript, Clojure, Erlang, PHP, Go, Mathematica...
- Databases and data stores: PostgreSQL, ClickHouse, DuckDB, Kafka...
- Web/network frameworks: Tornado, Twisted, Flask, Django...
- Big data and ML tools: Flink, Spark, TensorFlow, BigQuery, Dagster...
- Infrastructure: AWS/GCP, Kubernetes, Docker, Ansible, Terraform, Ceph, eBPF...
- Other: Emacs, git, Jupyter, Jenkins, Postfix, Bazel, OIDC...

Work Experience

Parcel Perform — Software Architect/Tech Lead/Staff Engineer, August 2019 -

The carrier-independent delivery experience platform that provides e-commerce merchants with standardized logistics data, analytics, and other value-added services.

My responsibilities:

- Leading the effort to transform backend services towards an event-driven architecture, enabled by Flink's stateful stream processing.
- Leading an engineering squad, then leading the DevOps and infrastructure team, then transferring to an individual contributor role.
- Forming and providing technical leadership for the data engineering team.
- Providing guidance for other team members in systems design and implementation.

Achievements:

- Led the migration of infra to another AWS region, k8s-ifying it at the same time.
- Significantly grew the engineering muscles of the infra team and the data engineering team.
- Engineered the internal Flink platform.
- Outlined the key principles and guiding data lake efforts.
- Championed many new technologies, technical innovations and good practices: scheduling system's efficiency, WAL-based CDC, data-driven AWS cost control, systematic observability, modern analytics stack...
- Designed and implemented an in-house DSL that enabled semi-technical users to define custom business rules.

Arimo — Senior Big Data Backend Engineer, March 2018 - November 2018

This was a company that focuses on behavioral AI in IoT systems.

In addition to continuing the work on the internal ML platform (which I designed while working here in the past), my responsibilities included:

- Improving security practices: cert-based SSH/VPN, tightened AWS access, secret management, infrastructure auditing...
- Revamping infrastructure tooling and practices to be more pervasively infrastructure-as-code.
- Designing customized systems for projects with special needs around data ingestion, model building/selection/serving/presentation.

Anduin Transactions — Software Engineer, May 2017 - January 2018

This was a startup that built suites of tightly-integrated tools to streamline private market transaction workflows.

My work here focused on infrastructural aspects of the systems:

- Designing and building fully IaC, heavily customized CI workflows based on Jenkins.
- Optimizing sbt-based builds to significantly reduce Scala build time.
- Designing and setting up tooling for writing MS Office add-ins in ScalaJS.
- Revamping authentication system, mail handling system.

Arimo — Senior Software Engineer, August 2015 - January 2017

This was a startup in the space of big data and machine learning. It was based in Mountain View, and had a Saigon office. Their work culture significantly improved my product-oriented mindset. While working here, I made frequent business trips to the Mountain View office to better collaborate with other teams.

My responsibilities included:

- Collaborating with product team to iterate through different product ideas.
- Building internal tools for data scientists.

- Designing and guiding infrastructure efforts.
- Coaching and providing technical support to other team members.
- Studying engineering details of deep learning.

BigApps and PredictiveEngine

BigApps was an application suite that empowered business users with self-service big data analytics for simple use cases, while at the same time enabled tight collaboration with data scientists for more advanced analytics. PredictiveEngine was the analytics system that allowed seamless integration of various data sources and processing engines, e.g. RDBMS, files on HDFS/S3, BigQuery, Spark.

I was responsible for refactoring multiple components of both systems to improve their production readiness and extensibility.

PENG (PredictiveEngine - Next Generation)

This was an internal platform that helped streamline different phases and activities of a data science project:

- Preparation: data acquisition, cleaning, exploration.
- Model building: feature engineering, experiment tracking, assisted hyperparameter search.
- Model operationalization: packaging, deployment, online/batch serving, model evaluation and monitoring.

To achieve its goal, the platform was designed to provide a set of tightly integrated tools:

- On-demand Jupyter notebook system with built-in integration with Spark and TensorFlow.
- Centrally managed repository of models and associated artifacts.
- Batch workflow system to run batch (training/serving) workflows, on-demand or on a schedule.
- Scalable online prediction server that exposed built models via REST/RPC endpoints.
- APIs and CLI tools to manage and automate most tasks.

The platform was designed to integrate well with Kubernetes. I was responsible for designing its architecture, implementing a large part of the system, coordinating the deployment, collecting feedback from data scientists to iteratively improve the platform.

Fram — Principal Software Developer, November 2014 - July 2015

This was a consulting firm specializing in Nordic markets, with a strong focus on the ”dedicated team” model. Here I worked with a Swedish client on a benchmarking platform for the hospitality industry. This platform collected daily/monthly raw data from hotels, then helped them analyze their performance compared to competitors, without leaking details of any particular hotel to another. My most significant contributions were:

- Designing and executing the database migration from MongoDB to PostgreSQL, which made the analytics system much more flexible, while at the same time increased performance by a factor of 2-3.
- Completing other refactoring efforts and infrastructure improvements to make the system more reliable, and to streamline the development process.
- Hardening the data protection measures to prevent hotels to abuse the system to gain unfair advantages over competitors.

My other internal responsibilities included:

- Automating the deployment of gitlab, jira, mail server, and maintaining them.
- Defining standards and best practices.
- Training junior developers.

Cogini — Senior Software Developer, February 2011 – September 2014

This was a consulting company with expertise in a wide array of technologies. In addition to technical tasks, I also acted as the main point of communication with some customers. For 5 months, I spent half

of the time working on-site at a customer's office in Taipei, in order to reduce communication friction so that their business needs can be quickly understood and transformed into technical requirements.

As I improved and became a senior developer, my responsibilities increased and included:

- Analyzing business needs and requirements.
- Designing, implementing, maintaining web systems.
- Researching new technologies and training junior developers.
- Defining standards and best practices.
- Coordinating teamwork.

Bid and Buy **March 2011 – November 2013**

This was one of the leading penny auction sites in Taiwan (bidandbuy.com.tw). Erlang was used to build the core real-time bid server, which talked to clients using a variety of techniques: AJAX, long polling and WebSocket (for the web frontend built with Yii), TCP-based APIs (for the iOS app). To support the core service, an admin site was built with an inhouse Python web framework built on top of Twisted. This admin system supported editing and scheduling auctions, keeping track of products and their inventories, monitoring orders and receipts, managing mail subscriptions, ads, coupons, testimonials, third-party integration... A helper tool was being built using igraph to combat collusion among bidders.

I was responsible for all aspects of the project: discussing, analyzing business requirements with stakeholders, designing, implementing, maintaining the bid server, the web frontend, the admin site, coordinating work with the iOS team.

Clickmedix **February 2012 – September 2014**

This was a tele-health platform (health.clickmedix.com) that featured an asynchronous, case-centric approach. Health workers used the web site, or the iOS/Android app to collect medical data, photos, videos, documents, using questionnaires prepared by professionals. These data were then presented to doctors who can colloborate with each other to give accurate diagnosis and prescription, on a case-by-case basis.

- Interesting problems: complex access control, complex business logic, external EHR integration.
- Responsibilites: requirement analysis, main developer of the web site and API.

Academy **June 2011 – September 2014**

This was an internal web-based management system for an English school (academy.com.tw). It helped the school's admins manage courses, classes, staffs, students, scheduling lessons and tests, keeping track of tuition fees, invoices and receipts, reviewing school's performance.

- Interesting problem: complex business logic, complex reporting UIs, PDF receipt generation.
- Technologies used: PostgreSQL, Twisted, wkhtmltopdf.
- Responsibilities: requirement analysis, main developer.

Mindoula Messenger **June 2014 - September 2014**

This was an internal instant messaging mobile app for a healthcare system that helped health workers communicate with their patients better.

- Interesting problems: real-time message delivery, multi-client synchronization in the face of unreliable network.
- Technologies used: Twisted, WebSocket, PostgreSQL.
- Responsibilities: project management, backend APIs.

Small highlights from other projects

- Bitmarq: customized and improved clearread (a text extraction lib).
- Enspyre: patched Yii to make it properly communicate with a legacy SQL Server system.
- Redirect: implemented fan-out of HTTP requests to determine highest bidder for a domain.
- H2: coached client's dev teams on using Twisted.

MultiNC **Intern, May 2009 – March 2010**

This was a small company that operated more like an R&D lab. We explored interesting cutting-edge technologies that were relatively little known or unused in Vietnam at the moment, and prototyped business ideas based on these technologies. We were among the first in Vietnamese tech community to tinker with git, Rails, Hackintosh and iOS development. The most important thing, however, was the spirit of learning and sharing we were trying to spread. We had a sibling non-profit organization called MultiUni, where we organized community-driven tech talks and courses on these technologies. I was able to learn much more during the time here than at RMIT: Linux, Emacs, git, Clojure, functional programming...

Personal Projects

emacs-module-rs

Binding and tools that enable writing safe Emacs's dynamic modules in Rust, to access native functionalities otherwise not available to Lisp.

Old Projects

emacs-tree-sitter

An Emacs Lisp binding for tree-sitter, an incremental parsing library. It inspired a new breed of Emacs packages that understand code structurally, as well as the eventual inclusion of tree-sitter in Emacs core.

js-csp

An implementation of CSP (Communicating Sequential Processes) for JavaScript, inspired by Go and Clojurescript's core.async. It brings sane concurrency control to JavaScript (compared to "callback hell", promises, or other flow control libraries), together with flexible primitives to build up data streaming patterns (as an alternative to the FRP approach by RxJS, Bacon, and Kefir), with good transducer integration for data transformation.

css-sync

A tool that shortens the feedback loop of frontend development, by immediately applying css file changes to browsers, without reloading the pages. These days, the use case is covered by modern JavaScript tools, but back when css-sync was first written (early 2012), it worked much better than the alternatives.

magit-wrapper

A wrapper for the best git client (compared to git's poorly thought-out CLI), geared towards more restricted enviroments where full-on Emacs customization is impossible.

cljs-xulrunner

Intended to be a base on which to enable first-class ClojureScript support in Mozilla's systems. Abandoned after XULRunner was deprecated.

Education

RMIT Vietnam **Bachelor of Information Technology, Feb 2009 – Aug 2011**

Despite not having used English for about 5 years, my analytical approach to learning languages allowed me to ace IELTS with the score of 8.5, after 4 months of preparation. During my time here, the most important gains were not the actual program, but rather its side effects, which can be summed up like this:

- The lackluster education by the school forced me to further hone my self-learning abiltity, which was already good, and put it to full utilization.
- The focus on English and soft skills by the school built up the necessary confidence. I also participated in the Akido, Kendo, IT clubs, mentoring junior students, organizing presentations, coordinating Robocode Tournament (a programming competition).

- The chances to meet interesting people in the tech community were invaluable. Working for a teacher I came to know during the early days of the program exposed me to the world of open-source and alternative non-Windows systems.

Moscow State University **Bachelor of Physics, Sep 2004 – June 2008**

Even though I exceled in all the subjects, I spent too much time tinkering with computer instead of doing physics research, resulting in skipping classes and eventually dropping from the program. The knowledge gained are now mostly forgotten, but the inquiring skills and resourcefulness required to explore the topics remain.

- Wrote a simple one-to-one chat app over LAN, to work around Internet's unavailability.
- Dug into Windows' internals to fix annoying software (e.g. Vietnamese IMEs).
- Did computer graphics courseworks (borrowed from a friend): small 3D game of interactively-controlled/automatic garbage collecting truck (with OpenGL, Tokamak), 3D visualization of maze path finding, simple recognition of 2D chess positions, simple sound processing with FFT.