

# Sid Verma

[sidverma.io](https://sidverma.io)   [@mavidser](https://twitter.com/mavidser)   [me@sidverma.io](mailto:me@sidverma.io)   Goa, India [UTC+05:30]

Thanks for visiting me. I'm a software developer, amateur artist, and a maker.

## Skills

### Full-stack development

Primary backend languages: **Golang**, **Python** and **Javascript**.

Most used frameworks: **Django**, **Flask** and **Express.js**.

Primary frontend frameworks these days: **React**, **Vue.js**

Big fan of simple, structured **HTML/CSS** with **vanilla JS**. Example: this webpage.

I use many kinds of production environments. I have set up and extensively worked with:

- Bare metal deployments using SSH and **Ansible**, on-prem and off-site.
- Containerized deployments on bare metal + VPSs, managed using **Docker**, **Kubernetes** and **Terraform**.
- Hosted kubernetes and other services on **AWS**, **GCP**, **DigitalOcean**, and **OpenStack**.

## Experience

I've spent the last decade working as a full-stack generalist and mentor, often working across domains in projects. This often includes **conceptualization**, **rapid prototyping**, **functional UX**, **workflow automation**, **development**, **code review**, **testing**, **infrastructure** and **deployment** setups. I also enjoy writing technical documentation + explainers.

### Opencraft (2020-21, 2023-24)

Worked extensively with OpenEdx at this **fully remote** and **distributed** consultancy, focusing on core platform development, deployment, client management, and process improvement.

- **Collaboration**: Worked with open-source maintainers and clients to develop effective, generic solutions.
- **Recruitment**: Led screening, interviewing, and onboarding processes.
- **Process Improvement**: Identified and resolved issues in communication, project planning, and team productivity.

Developed frameworks for better planning.

- **Sprint planning**: Enhanced team productivity and communication by developing bespoke tools for planning and monitoring.
- **Software development**: Architected and developed the core platform, including plugins and orchestration. Most of it in Python, Django, React and Golang.
- **Cloud-agnostic Infrastructure management**: Built tools and platforms to manage client infrastructure (cloud/bare-metal) using Ansible, Terraform, Kubernetes, OpenStack, AWS, GCP, and DO.

This list is quite reductive, and mostly reflects the primary technologies I've been using in recent past, for this specific role. I also work with **Kotlin**, **Swift**, **Rust**, and **C++** for mobile apps, microcontrollers, and other projects.

Additionally, I use **Audacity**, **Blender**, **OpenSCAD**, **Autodesk Fusion**, and the **Adobe Suite** for A/V production, 3D modeling and printing.

OpenEdx is an open-source education platform that enables anyone to spin up their own edX like instances.

It's a large platform with tons of capabilities. We provided the platform as a managed service, developed custom solutions for clients small and large, and worked closely with the open source community and maintainers to add new capabilities to the platform.

Here is an example of some of my open source work.

## Shortform (2021-22) , Solar Accounts (2020-21)

- **Developed Critical APIs:** Created robust APIs using Python and Node.js.
- **Built Vue.js Apps:** Transformed design mockups into functional Vue.js applications.
- **Infrastructure:** Improved the availability, cost and resilience of the AWS infrastructure.
- **Enhanced Security:** Integrated end-to-end encryption to secure accounting software.

## McLaren College (2020-21)

- **Course development:** Developed courses on Programming, Web Development, Dart, and JavaScript.
- **Classroom:** Instructed beginner software developers in these subjects.
- **Coding:** Built web-based tools for managing multiple concurrent classrooms.
- **Workflow optimization:** Developed efficient backend workflows for scoring, grading, and feedback.

## Tower Research Capital (2017-19)

My team built tools to manage the highly complex infrastructure. I worked on, among other things:

- **Monitoring System:** Developed an accessible monitoring and alerting system with robust data security for all teams and systems.
- **Orchestration System:** Developed an internal platform to easily provision different kind of systems.
- **Infrastructure Visualization:** Led development of tools for detailed physical and logical infrastructure visualization.
- **Automation Framework:** Redesigned the central automation framework to be generic and extensible, automating software operations and organizational processes.
- **Process Automation:** Developed new services and automations to speed up organizational processes, eliminating redundant authorizations and reducing wait times.
- **Performance Optimization:** Optimized core platform services and APIs for significant performance gains.
- **API, UI, & Infra Development:** Created APIs and UIs for various other internal organizational products. Core tech was - Python, Go, Java, C++, Vue.js, SQL, Neo4J, Kafka, Elastic, Docker

My team developed software to manage and secure the global trading infrastructure at this **High-Frequency Trading** firm.

We built tools for provisioning, monitoring, and debugging different kinds of systems, applications, and networks – ensuring seamless collaboration and customizable access across trading and operations teams worldwide, often with very complex Access Control.

## Hello English (2015-17)

- **Social news feed:** Designed and built a highly cost-efficient, fast social news-feed feature.
- **Chatbot platform:** Developed a chatbot platform and a user-friendly templating language for writing interactive stories, to teach foreign languages.
- **Speech recognition:** Built a multi-platform, on-device speech recognition and grading system for servers, web-browsers, and mobile.
- **CI/CD:** Introduced automation pipelines for testing, deployment, and rollbacks across all apps and services.
- **Electron app + PWA:** Created a framework to build an offline electron app, and a PWA from an existing dynamic website. This allowed the web-app to be packaged for running locally.
- Wrote **data aggregation services, browser extensions, IVR platforms**, and several **Android** modules.

## **Tickertape (2015-16)**

- Developed the main product, Stock Screener from scratch using **Node.js** and **AngularJS**; now serving **200K+** users.
- Designed intuitive **UI** and engineered **backend, frontend, and data pipelines**.
- Created **internal libraries** for seamless REST and SOAP API integration from external vendors.

## **Education**

### **Jaypee University of Engineering and Technology (2012-16)**

B.Tech, Computer Science and Engineering

I was the **sole developer** of the initial version of this product. I designed a functional UI in half a day, followed by months of coding.

Like the Ship of Theseus, most code I wrote has likely been cycled out. And yet, the UI still survives, almost exactly the same.

I am simultaneously annoyed and proud of this, given that I am *not* a professional UX designer.