

# Vincent Zvikaramba

📞 647-220-0748  
✉ zvikovincent@gmail.com  
in zvikaram  
🌐 vince2678

## Education

2021 **Honors BSc**, *University of Toronto*, Statistics and Computer Science

## Skills

Languages Python, Bash/Unix Shell, TypeScript/JavaScript, C/C++, Java, SQL  
Infra/DevOps Docker, Ansible, GitLab, Gerrit, Jenkins, BuildKite  
Other Debian/RedHat-based systems, Git, Android Build Systems, SE Linux, CloudFlare API

## Experience

Jun 2025 – **Developer**, *Client Management System*, South-Asian Women's Rights Organization, Toronto  
Present

- Redesigned PHP and SQL based CMS using Django and Python3 on back-end, and React with Next.JS for routing on the front-end, ensuring future scalability and maintainability.
- Used Django Rest Framework to build REST API for communication between front-end and back-end.
- Used Celery for task queuing and Redis for thread communication, caching and concurrency locking on the back-end in Django.

Apr 2022– **Full Stack Engineer**, *TitanFile Inc*, Toronto

### Accomplishments

- Migrated from Python3.8 to Python3.11 as DRI of Infrastructure, improving Python performance by 20%.
- Optimized GitLab CI jobs as DRI of DevOps to skip unneeded or unchanged jobs, speeding up testing pipelines by 10%.
- Implemented MS365 integration from scratch to allow enabling seamless document editing; and reduced technical debt by adopting TypeScript in the code-base and creating a template for future front-end development in TypeScript and React.
- On-boarded and mentored 3 new engineers; directly mentoring one engineer to become DRI for MS365 integration development.
- Created new React components and views as part of Admin Settings Console rewrite on the front-end, implementing 90% of views according to requirements within 3 months.
- Resolved performance bottlenecks in PDF watermarking of large files on the back-end, reducing downtime and memory usage by 10% and allowing for more responsive PDF previewing of large files on the front-end.
- Reduced memory usage and increased up-time by analyzing slow database queries causing crashes, using SQL commands on the back-end to optimize slow Django queries.
- Implemented React Router routing as part of front-end rewrite, simplifying browser-side navigation.
- Updated DocuSign integration to accept JSON web-hooks, increasing security by using more secure XML parser implementation and enforcing future usage of JSON web-hooks, while maintaining compatibility with older XML-based web-hooks for a seamless update.
- Ported legacy Marionette views to React in TypeScript, improving code readability, reducing technical debt and minimizing uncaught bugs.
- Leveraged React Context API on the front-end to manage and share state between components, improving the scalability and maintainability of the application.

### Responsibilities

- Updated Ansible templates for services running on CentOS instances to patch security scanner vulnerabilities, helping adhere to OWASP Top 10 and maintain security compliance scores.
- Participated in code reviews, helping enhance code quality, improving maintainability and reducing bugs by encouraging usage of TypeScript over JavaScript and type hinting in Python.
- Translated Figma designs into fully responsive and interactive front-end components using React, TypeScript and SCSS, with a focus on code and asset re-usability through usage of standard icon sets such as Font Awesome.
- Wrote code and handled data with security in mind, ensuring HIPAA and GDPR compliance.
- Collaborated with other DevOps team members to migrate legacy virtual machine based systems to containerized environments using Docker.
- Created and maintained Bash shell and Python scripts to run scheduled jobs or perform back fills and administrative tasks on the back-end.