

Ty Nasello

tnasello@uwaterloo.ca | tynasello.com | linkedin.com/in/ty-nasello | github.com/tynasello

Work Experience

Software Engineer Intern @ Orbiseed Technology Inc.

Toronto, ON (Remote) | Sept 2022 - Oct 2022 (company ceased operations) | Employer Evaluation: Outstanding

- Implemented application features and business use cases enabling insurance carriers to effectively manage documents.
- Developed REST API endpoints, made client-facing web UI changes, and wrote unit and end-to-end tests. (TypeScript, Express.js, React.js, MongoDB, Jest/Chai).
- Employed agile practices, worked cross-functionally with designers, and leveraged numerous libraries such as InversifyJS (an inversion of control container) and Mongoose, as well as tools like AWS (S3 and SQS) to carry out feature specifications.
- Learned how a CI/CD pipeline is constructed, and how services like AWS (ECS, ECR, CodePipeline) can be utilized.

Software Development Intern @ MedMe Health

Toronto, ON (Remote) | Jan 2022 - April 2022 | Employer Evaluation: Outstanding

- Developed pharmacist and patient workflows for web apps enabling over 2200 pharmacies to provide clinical services at scale.
- Implemented numerous shipped features and was an initial driver in unit testing new front-end components. (React.js, TypeScript, GraphQL, PostgreSQL, Java, Spring Boot, Redux, Jest, Agile).
- Increased team efficiency and outsourced work to non-developers by independently designing a tool that expedited the process of mapping PDFs with patient information stored in PostgreSQL databases.

Projects

Clean Architecture Template | Dec 2022 | [GitHub Link](#)

- Created a template for a monolithic application that follows the Clean Architecture design philosophy. The repository offers a starting place for developers wanting to build their next project. The template adheres to SOLID design principles throughout and prioritizes modularity and loose coupling.
- The application consists of an API and databases which are managed through Docker. REST controllers are guarded via JSON Web Tokens. Web sockets are configured to emit events to authenticated connections. End-to-end, integration, and unit tests exist throughout the application. (TypeScript, NestJS, PostgreSQL, Docker).

Facial Expression Classifier | July 2022 | [GitHub Link](#)

- Constructed a convolutional neural network (CNN) that classifies images of human faces based on their emotions.
- Employed transfer learning and performed both hyperparameter tuning and data augmentation to improve model performance, reaching a test accuracy of 65.5% (would've ranked top 5 in Kaggle competition). (Python, PyTorch, NumPy, Pandas, Matplotlib).

Food Delivery App | April 2022 | [GitHub Link](#) | [Live Demo](#)

- Designed and built a food delivery web application clone. Users can log in and add and manage food items in their shopping carts.
- Constructed a GraphQL API to support CRUD functionality and user authentication, designed a multi-page UI from scratch, wrote unit and end-to-end tests, and hosted the application on Heroku. (NestJS, TypeScript, React.js, GraphQL, PostgreSQL, Docker).

Older Projects: Arduino Robot Car, Color Sorter Robot, CV Builder, Flappy Bird (Recreation).

Education

University of Waterloo | Candidate for BAsC Mechatronics Engineering | September 2021 - April 2026

CGPA: 3.97. Relevant Coursework: Algorithms and Data Structures, Digital Logic, Linear Algebra, Digital Computation, Calculus II.

Skills

Languages: TypeScript/JavaScript, Python, C++, Java, SQL, HTML/CSS

Technologies: Node.js, Docker, React.js, GraphQL, NestJS, Spring Boot, PyTorch, PostgreSQL, MongoDB, Git