

Dominic Casillano

647-870-7095 | dominiccasillano710@gmail.com | [linkedin.com/in/dcasillano](https://www.linkedin.com/in/dcasillano) | github.com/casillano

EDUCATION

University of Toronto

September 2018 - June 2023

Honours Bachelor of Science, Computer Science Major

Toronto, ON

- GPA: 3.56/4.0
- Relevant Coursework:
 - Data Structures and Analysis, Algorithm Design and Analysis, Operating Systems, Introduction to Software Engineering, Database System Technology, Compilers and Interpreters, Computer Networks

TECHNICAL SKILLS

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

Frameworks: React, NodeJS, EmberJS, Express, Django REST Framework, PostgreSQL, MySQL

Developer Tools: Git, VS Code, AWS, Android Studio

EXPERIENCE

Software Engineer Intern

May 2021 - April 2022

iManage Canada Technologies

Toronto, ON

- Investigated slow REST API endpoints used for data logging and improved their performance up to 99% by implementing database indexing and modifying the Django ORM to optimize queries
- Designed a database schema change and engineered new APIs using Python and the Django REST Framework to give customers access to specific data for their usage reports
- Refactored the signing package generator using the XState state machine library to create a module that manages complex data states clearly and resolves 100% of the pre-existing issues
- Developed the sorting of deals by most recently accessed and implemented the feature into the deals page UI using EmberJS to allow users to quickly find their frequently used deal checklists
- Redesigned the database backfill process and wrote a training document for 12 engineers to prevent application slowdown by minimizing AWS CPU credit usage

PROJECTS

Palyglot

- Developed a web application using JavaScript that gives language learners a platform to connect and chat with others to improve their written literacy in a foreign language
- Built a REST API using NodeJS, Express, and MongoDB to handle requests from the React frontend
- Created a matchmaking system to match users who are learning the same language(s) and have similar interests
- Implemented real-time messaging using the Pusher API

YouTube Karaoke

- Built a web application using JavaScript and React that takes a YouTube music video and adds lyrics that get highlighted as they are spoken to create a karaoke experience
- Retrieved videos and their respective lyrics using the YouTube API and Genius API
- Created an API using NodeJS and Express that uses the Gentle forced-aligner to align the video speech with the lyrics

Key-Value Database

- Worked in a team of 3 to create a database using C++ that stores key-value pairs of integers
- Built a memory table with an AVL tree structure to optimize data fetching and batch writes to minimize I/O operations
- Implemented an extendible hash table to store frequently accessed data in memory, resulting in constant-time performance for some queries
- Optimized writes and lookups by refactoring the database directory to store files in an LSM tree with bloom filters