

Igor Chovpan

437-665-0196 | i.chovpan@mail.utoronto.ca | chopikus.dev | github.com/chopikus | linkedin.com/in/chopikus

EXPERIENCE

Junior Software Developer

July 2022 – July 2023

Keepit; a backup solution for cloud services

Krakow, Poland

- Launched *Azure Devops* backup coverage working with 3 teammates over the course of 12 months;
- Optimized the *REST API* efficiency by up to 99% in extreme cases by preventing the redownload of *Work Items*;
- Collaborated with *Quality Assurance*, *Customer Support*, *Product Design* teams, presented projects to the *CTO*;
- Developed a file restore scheduling algorithm handling up to 10 million file dependencies;
- Participated in refactoring *C++* development and *Java* testing code, unifying *JSON* parsing approach.

TECHNICAL SKILLS

Languages: C++, Javascript, Java, Python, Go, Rust

Other: Linux, REST API, Svelte, bash

EDUCATION

University of Toronto

Expected Spring 2027

Computer Science Major, Mathematics Major, Coop student

Relevant coursework:

- CSC265 – Enriched Data Structures and Analysis – A+;
- CSC311 – Introduction to Machine Learning – in progress;
- CSC207 – Software Design – A.

Supporting the Ukrainian community outside of class. Volunteering note-taking for Calculus and ML courses.

PROJECTS

rm-exporter

- Researched limitations of note export for a *reMarkable* tablet, including inability to select a folder and failure to download notes larger than 10MB;
- Made a GUI client exporting any combination of folders and large notes using *Go*, *Typescript* and *Svelte*;
- Added the project to the *awesome-remarkable* list, fixed bugs found by the community, reached 100 downloads.

spydle

- Developed a real-time multiplayer word-guessing game in a team of 5 people over the course of 2 months using *Java*, *Spring Boot*, *Kubernetes*;
- Implemented core logic of the game, deciding the next turn's player, handling game time, validating user guesses;
- Improved reliability of the game by preventing concurrency issues and implementing unit, end-to-end tests.

game-of-life

- Wrote a robust implementation for *Conway's Game of Life* mathematical simulation in *Rust*, *Javascript*, *WebAssembly*, ensuring memory safety and multiplatform support;
- Optimized time usage by using *Hashlife* high-performance algorithm and running it on a separate thread, allowing to render millions of state updates per second;
- Shared technical details by writing an explanation blog and implementing integration tests.