# Igor Chovpan

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

## Experience

**Junior Software Developer (C++)** **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* usage up to 99% in extreme cases by preventing the redownload of *Work Items*;
- Participated in daily meetings, discussions in a team, argued projects' details with the tech lead and the CTO;
- Refactored *JSON* parsing in the *C++* development & *Java* testing for 4 months, unifying *JSON* parsing.

## Technical Skills

**Languages**: C++, Golang, Rust, Javascript, Java, Python

**Other**: Linux, REST API, bash

## Education

**University of Toronto** **Expected Spring 2027**

*Computer Science Major, Mathematics Major, Coop student*

**Coursework:** CSC265 – Enriched Data Structures and Analysis – A+, CSC311 – Introduction to ML.

**uoftctf**: top 80 out of 1225 teams participating in the University 'Capture The Flag' cybersecurity tournament.

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

## Projects

**rm-exporter**

- Found limitations of the local export on a *reMarkable* tablet, including inability to export a folder and failure to download notes larger than 10MB;
- Developed a GUI client supporting export of any combination of folders and large notes by interacting with a tablet's local *HTTP* server using *Golang* and *Typescript*;
- Added the project to the *awesome-remarkable* list, fixed bugs found by the community.

**game-of-life**

- Improved implementation for *Conway's Game of Life* mathematical simulation by writing a client in *Rust* & *Javascript*, ensuring memory safety and supporting mobile devices;
- 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.

**raytracing-bench**

- Measured performance of 3D sphere raytracing renderers in *Java*, *Python*, *Numpy*;
- Achieved 7x – 993x speedup compared to the *CPU*-based renderers by migrating to the *CUDA* architecture;
- Contributed to the *TornadoVM* computation project by reporting an issue and writing an installation guide for a specific platform.