

# Liam Mercier

LiamMercier@proton.me | [github.com/LiamMercier](https://github.com/LiamMercier)

## Education

### Simon Fraser University

Bachelor of Science in Computer Science, with distinction  
Minor in Mathematics, Concentration in Artificial Intelligence

2025

- Relevant courses: Deep Learning, Computer Vision, Intelligent Systems, Computational Data Science
- President's Honor Roll

## Projects

### Silent Tanks | C++, Qt, Boost, PostgreSQL, OpenSSL, CMake

[GitHub](#)

- Developed an asynchronous, multithreaded game server validated to support >3,000 concurrent users with negligible turn latency on consumer hardware
- Improved maintainability by separating server components for client connections, user management, matchmaking, database operations, and administrative actions
- Integrated PostgreSQL backed by a thread pool with per-user request serialization
- Hardened application security with Argon2id password hashing and TLS encryption for data protection
- Created a bucket-based matchmaking system and custom extension of the Elo rating system to balance match quality, queue times, and rating inflation
- Designed a cross-platform GUI client in C++ and QML with a user-managed server connection page, social features, and interactive replay viewer

### Loadshear | C++, Boost, wasmtime (WebAssembly), GTest, FTXUI, CMake

[GitHub](#)

- Developed a multithreaded, sharded load generator with a contention-free design capable of generating >16 Gbps of TCP or UDP traffic
- Implemented user-defined packet response handling with an embedded WebAssembly runtime
- Designed a domain-specific language for defining execution flow and generation parameters

### Audio Deepfake Detection | Python, PyTorch, scikit-learn

[GitHub](#)

- Designed an end-to-end data pipeline to preprocess audio, extract different features, and train models
- Compared in-context performance and transfer learning using domain-specific criteria across models shown to be promising in external works and competitions
- Performed ablation studies to compare how model architectures and feature extraction techniques affect test performance and inference time

### Social Media Sentiment Analysis | Python, scikit-learn, NLTK

[GitHub](#)

- Interfaced with external APIs to automate collection of 30,000 text samples from selected communities
- Implemented a data processing pipeline using the Natural Language Toolkit to tokenize and normalize text

## Experience

### Mathematics and Computing Tutor

2023 - Present

- Applied evidence-based learning strategies to improve mean z-scores by 0.30 and average percentile ranks by 10.8% for students in undergraduate-level courses

## Skills

**Languages:** C++, C, Python, QML, SQL

**Libraries:** Boost, Qt, GTest, wasmtime, scikit-learn, PyTorch, Pandas, NumPy