

# Alan Yong

Computer Science Student

alanyongy@gmail.com | GitHub: <https://github.com/alanyongy> | LinkedIn: [linkedin.com/in/alan-yong-abb154395/](https://linkedin.com/in/alan-yong-abb154395/)

---

## EDUCATION

### BSc Computer Science

University of Calgary — Expected Graduation: 2027

GPA: 3.51 / 4.00

Relevant Coursework: Data Structures & Algorithms, Operating Systems, Distributed Systems, Artificial Intelligence, Software Engineering

---

## TECHNICAL SKILLS

**Languages:** Python, C#, Java, AutoHotkey

**Technologies & Tools:** Unity, GitHub, Simulation Systems, OCR, Real-time Systems, AI Coordination

**Concepts:** Object-Oriented Design, Distributed Systems, Game AI, Simulation, UI/Input Systems

---

## PROJECTS

### TFT Matchup Predictor (2021)

*AutoHotkey — Real-time Overlay & OCR Tool*

GitHub: <https://github.com/alanyongy/tft-matchup-predictor>

- Built a real-time overlay tool that predicts upcoming opponents in **Teamfight Tactics** using only live screen data
  - Implemented a **custom OCR system** using pixel-based image matching (ImageSearch) to identify and associate player names across multiple UI regions
  - Designed robust screen-region calibration using static UI anchors for reliable detection across resolutions
  - Reverse-engineered matchmaking rules and replicated the game's opponent-selection logic **before official support existed**
-

## AEGIS Multi-Agent Rescue AI (2025)

*Python — Multi-Agent Simulation & Coordination*

GitHub: <https://github.com/alanyongy/aegis-multiagent-rescue-ai>

- Developed a multi-agent rescue AI coordinating autonomous robots under **1-turn communication delays**
  - Implemented per-turn local simulation of all agents' future actions to maintain synchronized state despite communication delays
  - Solved distributed coordination constraints through deterministic simulation rather than increased communication
  - Led design and implementation of the core coordination strategy within a team environment
  - Achieved **100% assignment score** and significantly reduced total rescue turns across varied test scenarios
- 

## Scalable Skill System & Multi-Unit Control (2025)

*C#, Unity — Systems Design & Input Handling*

GitHub: <https://github.com/alanyongy/game-systems-showcase>

- Designed a **modular, extensible skill architecture** supporting rapid creation of diverse abilities through class hierarchies and inspector-driven configuration
- Implemented polished **real-time multi-unit selection and movement**, supporting click, drag, deselection, and continuous command updates
- Built a **priority-based skill queue system** enabling autonomous AI and player-controlled skill execution with dynamic condition checks
- Emphasized clean architecture, scalability, and maintainability for long-term project growth