Sankeeth Ganeswaran

647-822-3351 | s2ganesw@uwaterloo.ca | LinkedIn | GitHub | Website

Education

University of Waterloo Sept. 2021 - April 2026
Bachelor of Computer Science, Co-op *Waterloo, ON*

Technical Skills

Languages: C#, C++, JavaScript, TypeScript, Python, Java, Kotlin, C, SQL, HTML/CSS, R, ActionScript 3
Frameworks: Unity, Unreal Engine, JavaFX, TensorFlow, PyTorch, Android, React, Flask, Swing, JUnit
Developer Tools: Git, Docker, Blender, Gradle, Postman, AWS, Oculus
Libraries: Pygame, NumPy, ThreeJS, Photon PUN, Unity VR, OpenAI Gym, JSoup

Experience

Game Developer Sept. 2023 – Aug 2024
ArenaX Labs *Toronto, ON*

- Developed features and machine learning systems for **AI Arena**, a platform fighter with **200k+** active players.
- Implemented core gameplay mechanics for the combat system in **JavaScript**, ranging from **finite state logic**, **animation systems**, **projectile physics**, **collision handling**, and implementing **20+** unique **elemental VFX** in **ThreeJS**.
- Designed solutions for **50+** unique issues, including visual and gameplay fixes, revamping **raycast systems**, as well as optimizing performance to cut down memory usage and load times by **40%**.
- Created a built-in **interactive tutorial** for the game, implementing **15+** in-game demonstrations and using **React** to design a **UI** considering player experience and aesthetics.
- Worked on the AI agent for a final boss, designing a cohesive moveset and implementing **30+** state animations.
- Built several minigames in **Pygame** to train **reinforcement learning models** using **OpenAI Gym** environments.
- Aggregated feedback from players and designed effective solutions to maintain gameplay balance and player satisfaction.

Gameplay Programmer Jan. 2023 – April 2023
Lucky VR *Toronto, ON*

- Developed gameplay mechanics for the popular game **PokerStars VR** in **Unity**, for **PC, Quest, and PSVR**.
- Implemented scalable functionality in **C#** for **50+** new props and apparel for the **VR environment**.
- Fixed critical bugs and made significant QOL improvements for **70+** issues, ranging from **collision and physics interactions**, **backend integration**, **texturing**, and **networking** using **Photon PUN RPC** calls.
- Worked on implementing audio SFX and ambient noise using **WWise** integration.
- Revamped several weapon systems and VR avatar interactions to improve performance and eliminate latency by **60%.**

Autonomous Vehicle Android Developer May 2022 – Aug. 2022
Ford Motor Company of Canada *Oakville, ON*

- Developed an **Android application** using **Kotlin** to send and receive **CAN and SOA messages** through the VHAL to set **60+** fundamental automobile property functions.
- Implemented an audio service for the infotainment system in Java using **Android Open Source Project Automotive** and **Google TTS service**, with the ability to play **40+** different prompts in **3** languages.
- Created **50+** unit tests with **Mockito** and **Robolectric** frameworks, resolving **70+** application bugs and issues.
- Wrote **20+** UI tests using the **Facebook Litho** framework to test sub-component structures in the DXP FAQ app.

Projects

Creature Clash | *Unity, C#, Photon PUN*

- Developed a physics-based mobile game using **Unity** engine and exported to **Android** to publish to the Play Store.
- Used **Photon PUN** framework to create an online multiplayer lobby, with an **RPC** model.

MeetKicker | *JavaScript, HTML, CSS*

- Developed and published an extension to the **Chrome Web Store** that allows the user to kick themselves from a **Google Meet** once the members go below a customizable threshold, reaching **250+ active users** at its peak.
- Utilized **CSS** to design the extension panel and **Chrome API** to store user preferences in **browser storage**.

TypingTest | *Python, HTML*

- Built a user-friendly **typing-test application** that extracts articles from the web as typing material.
- Provides feedback and statistics on typing performance in **real-time**.
- Scrapes articles and parses **HTML** using **BeautifulSoup**, and displays them on **Tkinter UI**, with **Pygame** for audio.

SelfDrivingCar | *JavaScript, HTML, CSS*

- Created a simulation of a **self-driving car** that learns to navigate randomly generated traffic using a neural network.
- Designed a **real-time visualizer** of the neural network, as well as a system of **loading and storing previously trained networks** through local storage in **JSON** format.