# Jaz Winn Ng

**Mobile:** +65 9107 7490
**Email:** ngjazwinn@gmail.com
**LinkedIn:** https://www.linkedin.com/in/jazwinn-ng
**GitHub:** https://github.com/jazwinn


## PROFILE

Computer Science (Real-Time Interactive Simulation) undergraduate at Singapore Institute of Technology with experience in software development. Able to work in a fast-paced environment, demonstrate problem-solving skills and continuously seek to expand technical knowledge. Seeking the opportunity to grow and contribute meaningfully as a software engineer.

## EDUCATION

**Singapore Institute of Technology Singapore (Digipen)** August 2023 – Present

- Bachelor of Computer Science in Real-Time Interactive Simulation
- **Relevant Modules:** Machine Learning | Parellel programming | Low-Level Programming

## SKILLS

- **Programming Language:** C++, C, C#, Java, JavaScript, HTML, CSS
- **Concepts:** Object-Oriented Programming, Spatial Data Structures and Algorithms
- **Frameworks & Libraries:** OpenGL, OpenCV, FFmpeg, Dear ImGui , GoogleTest, Spring Boot, React
- **Database:** MySQL, MongoDB
- **Tools:** Git (Bitbucket & GitHub), CI/CD, CMake, Docker

## PROJECTS

**Push-Up Counter**

- Built using YOLO pose estimation and OpenCV.
- Detects exercise form and posture angles in real time.
- Provides immediate audio feedback to correct form during workouts.

**Spatial Data Structures**

- Implemented advanced spatial data structures in C++ to optimise rendering and collision detection.
- Utilised structures such as Bounding Volume Hierarchy (BVH), Binary Space Partitioning (BSP), and K-Dimensional Tree to efficiently manage scene geometry.
- Achieved performance improvements of up to 5 times in rendering and collision queries through optimised spatial partitioning and traversal algorithms.

**Procedural Cave Generation**

- Developed using C++20 and OpenGL to create a 3D cave system.
- Implemented binary space partitioning, Perlin noise, and cellular automata algorithms to generate cave structures procedurally.
- Applied greedy meshing and the marching cubes algorithm to improve rendering performance.

**Custom Game Engine 1** | **Custom Game Engine 2**

- Led as Technical Lead to design and develop a C++ game engine from scratch.
- Defined system architecture and implemented key systems, including compile-time reflection, scripting, video player, asset manager, and scene manager.
- Managed project milestones, coordinated Git workflow and code review to ensure smooth collaboration, improving efficiency and streamlining workflows.

## WORK EXPERIENCE

**Teaching Assistant, Digipen Institute of Technology | Singapore** September 2024 – April 2025

- Mentored freshmen in computing and mathematics modules, sharing insights and practical tips to help them succeed.