

# Research Project Proposal

Team Members (Last name in caps)

Victor Meng Hui LEE	lee.v@digipen.edu
Joel TEO	t.joel@digipen.edu
Dylan Chiak Hong LAU	d.lau@digipen.edu

## **What we will be researching**

Our team will be researching on NPC Behavior Tuner via Reinforcement Learning (RL). In this project, we are researching the use of RL—specifically Q-learning—to train non-player characters (NPCs) with distinct and tunable behaviors in a custom C++ game environment. Instead of relying on manually coded behavior trees or scripted logic, our goal is to build an AI system where designers can influence NPC personalities (e.g. aggressive, defensive, evasive) through customizable reward functions. We aim to develop a self-contained RL module that integrates directly with a C++ game engine, allowing for adaptive, intelligent behavior without requiring external Machine Learning libraries or engines like Unity.

## **What the demo will consist of**

Our demo will showcase multiple NPC agents trained using different reward configurations inside a 2D game simulation, built on a base C++ engine. The environment will include basic obstacles, a player target, and multiple NPCs exhibiting behaviors such as chasing, patrolling, hiding, or avoiding. Users will be able to observe each NPC's strategy in real-time and switch between different behavior profiles. The demo will include visual indicators and logs of training progress, as well as a simple interface to adjust reward structures and retrain behaviors, making it both interactive and educational.

### **Why the topic is worthwhile to professional developers**

This research provides a practical framework for creating dynamic and scalable NPC behaviors without heavy scripting, which is especially useful for open-world, strategy, and simulation games. Instead of manually fine-tuning every behavior pattern, developers can train agents to behave according to desired traits by defining high-level goals. This approach could benefit titles like Shadow of Mordor (enemy personalities via the Nemesis system), Mount & Blade II: Bannerlord (varied troop AI), Total War series (tactical army behavior), or Far Cry (dynamic enemy outposts). By automating and tuning NPC behavior through reinforcement learning, developers can reduce iteration time and increase replayability and realism.