Ollscoil Teicneolaíochta an Atlantaigh

Atlantic Technological University

G00383507

LOKESH PANTI MAZE GAME

FINAL YEAR PROJECT

BENG (HONS) SOFTWARE & ELECTRONIC ENGINEERING

OBJECTIVE

To observe Machine Learning (ML) Agents, placed within a randomly generated maze to navigate and find the flag or goal. The agents will have to spent time learning how they can achieve this. The user will also be able to play against these agents to reach the goal first.

FEATURES

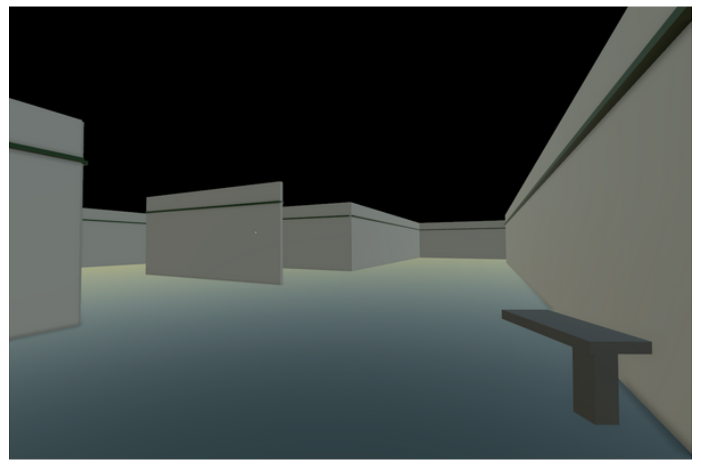
- Random Map generating algorithm so you are in a new map every time.
- ML-Agents to fight and compete with.
- Fighting and shooting mechanics.
- Able to play the game as player.

SOFTWARE USED

- C#, for generating scripts
- Python, to run Machine Learning Agents
- Pytorch
- Anaconda prompt
- ML-Agents, acting as enemies
- Unity Gaming Engine
- VSCode IDE

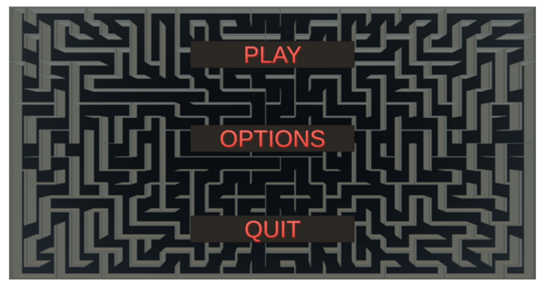
WEAPON

The user is given a weapon that shoots slow bullets. The user has full control over where they shoot.

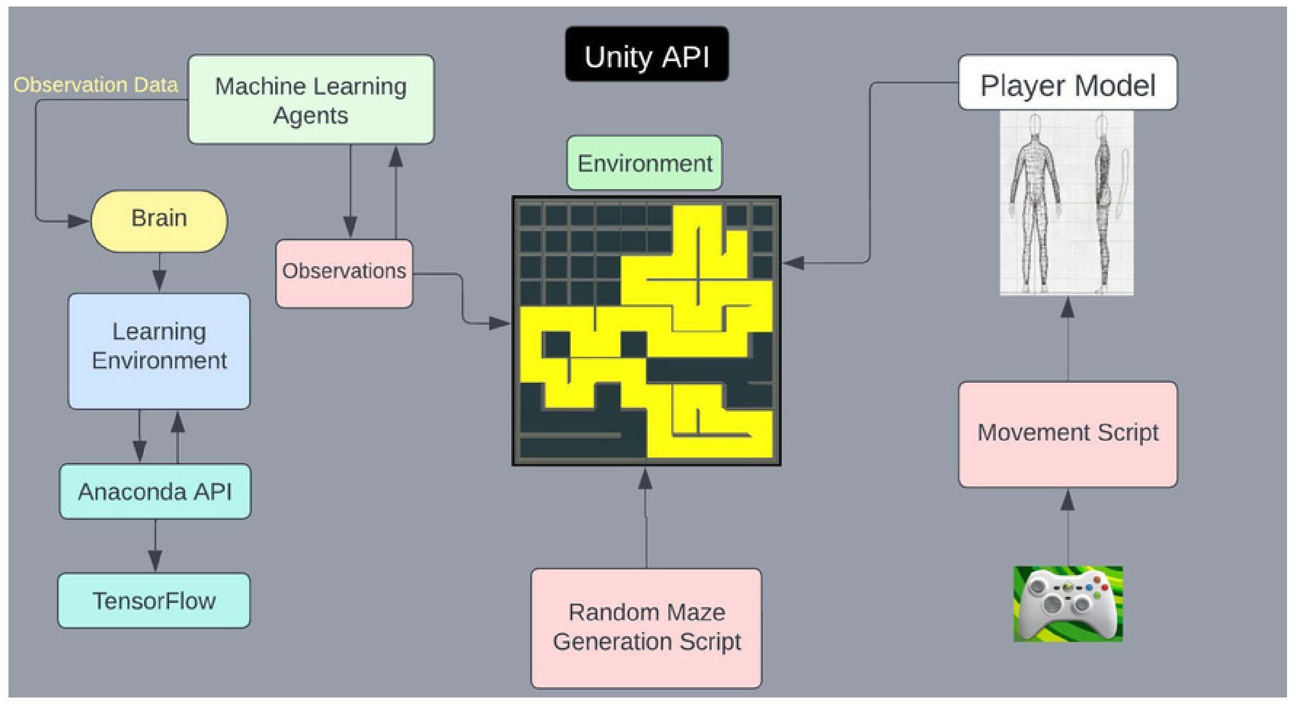


MENU

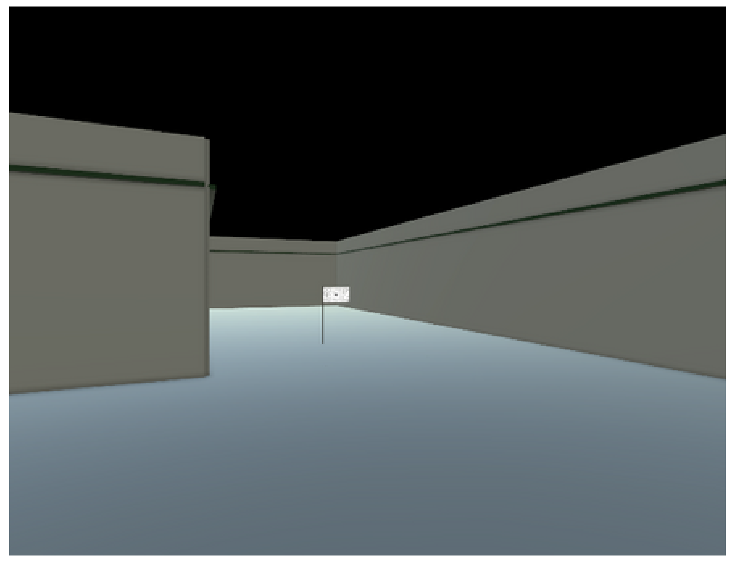
The Player can watch the maze as it generates and they are placed inside the maze. They also have a basic Menu system and pause Menu



ARCHITECTURE DIAGRAM

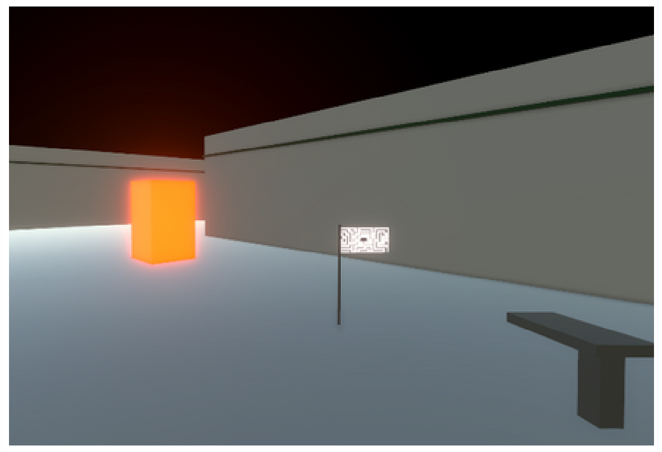


FINAL PRODUCT



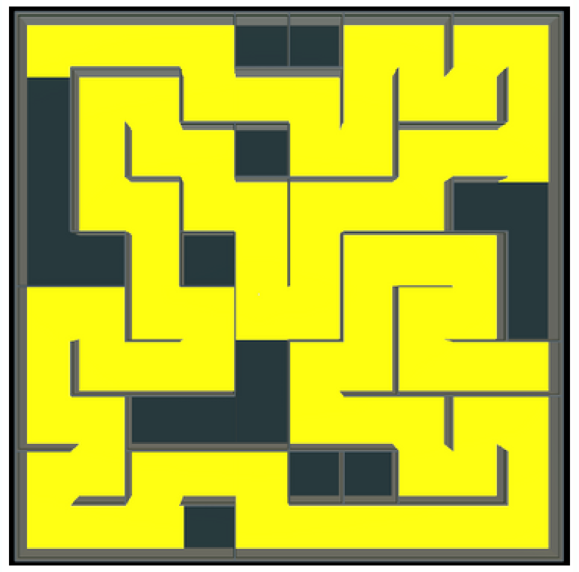
You are able to play the game against the ML-Agents as they navigate the maze. When the agents see you, as both the player and agent navigate the maze, the agent will then try and reach you to tag you and the game will be over.

This is the end goal, to reach this flag and capture it. But the flag is placed at a random location in the maze. It could be close to you or close to the agent but both will have to navigate the maze to capture it.



RANDOM MAZE ALGORITHM

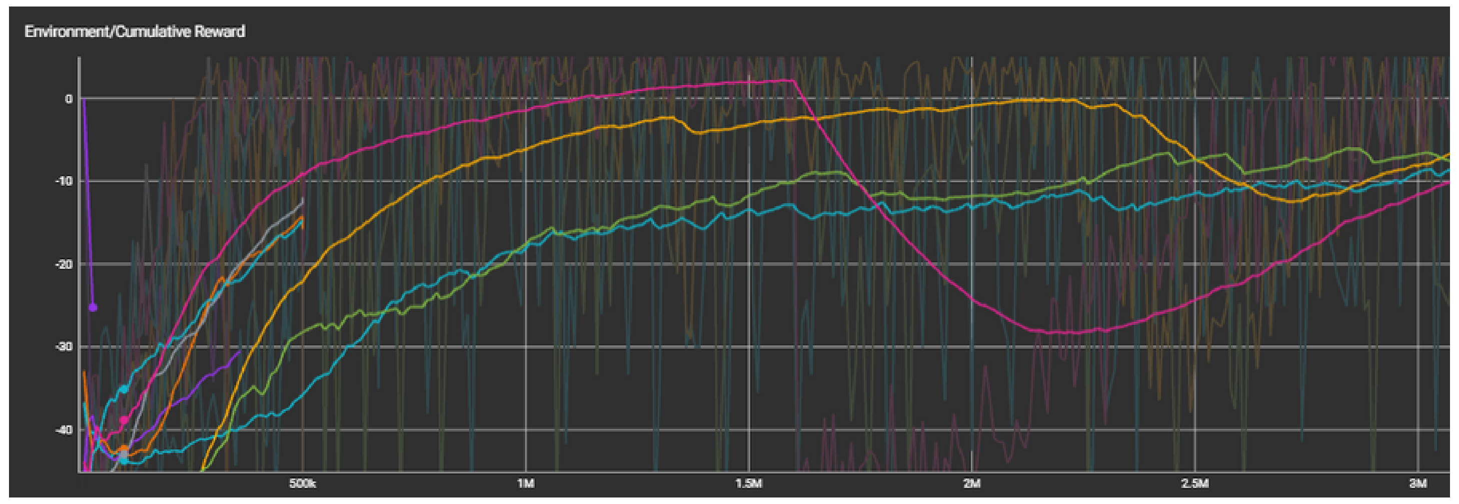
An environment model called a "Node" is used to print out a grid sized 10x10 and at random, walls on all 4 sides are removed to generate a maze which is different every time the game is launched.



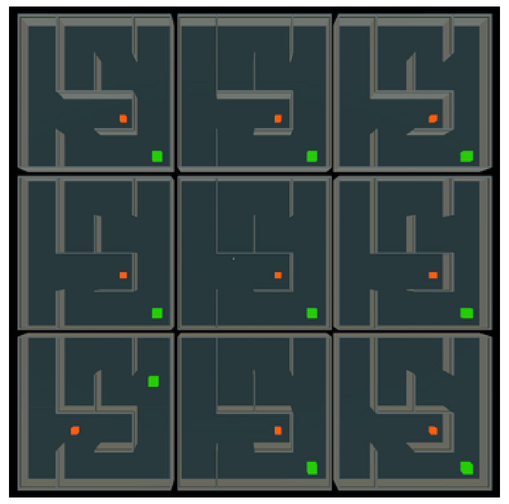
MACHINE LEARNING AGENT TRAINING

Using a unity feature called "Ray Perception Sensor 3D", The agents were placed within a maze to see the success rates of navigation. The agents are given a script to follow when training. This script allows them to take observations and move around the maze accordingly.

This graph shows the success rate of the agents. The graph indicated that the agents are learning much quicker to gain more rewards.



Train multiple agents at the same time. To increase learning speed



3D CHARACTER DESIGN

Using Blender, I generated character models for the main player. The model is shown in detail in the Architecture Diagram. I also used Blender to add fluidity and wind mechanics to the flag.