AI-Powered Game NPCs

Assignment Overview

In this project, students will explore how artificial intelligence can shape realistic and responsive non-player characters (NPCs) in games. Students will use Python and machine learning libraries (e.g., scikit-learn or TensorFlow) to design intelligent behaviors that could be integrated into Unity-based game environments using Unity ML-Agents or simulated environments.

This project emphasizes not only the technical side of AI but also ethical considerations, such as bias in training data, NPC decision-making, and the implications of autonomous behavior in games. Students will work collaboratively, develop documentation, and present their findings and prototypes.

Learning Objectives

- Understand how machine learning can be applied to control NPC behavior in games
- Use Python to train basic models that predict or respond to in-game scenarios
- Explore ethical concerns in AI decision-making and game design
- Develop technical documentation that describes AI model logic and behavior
- Collaborate to design, test, and refine intelligent systems in game-like environments

Tools and Materials Needed

- Python (Anaconda, Google Colab, or local setup)
- Machine learning libraries (e.g., scikit-learn, Keras, TensorFlow)
- Unity (optional – for use with ML-Agents toolkit)
- Jupyter Notebook or similar for documentation
- Ethical case study readings or scenario prompts

Steps and Instructions

1. Explore sample NPC behaviors in games (e.g., patrol, chase, flee, dialogue branching)
2. Choose one behavior to simulate or improve using Python and AI

3. Train a machine learning model (e.g., classification, reinforcement learning) on a relevant dataset
4. Optionally, integrate your model into Unity using Unity ML-Agents (or simulate behavior through Python-based environments)
5. Write a technical documentation file explaining your model: input/output, logic, accuracy, behavior mapping
6. Conduct a team discussion on ethical concerns of NPCs and AI decision-making
7. Present your project and findings to the class

AI Behavior Design Table

Behavior Type	Training Data	ML Algorithm Used	Input Features	NPC Action

ISTE Standards Addressed

- **1.5.a – Computational Thinker**: Students develop and test solutions using logical reasoning and algorithmic thinking
- **1.7.b – Global Collaborator**: Students work in teams and explore diverse perspectives when creating with AI
- **1.3.c – Knowledge Constructor**: Students evaluate sources and use data to construct knowledge in a responsible way

Assessment Rubric – AI-Powered Game NPCs

Criteria	Beginning (1 Point)	Developing (2 Points)	Proficient (3 Points)
Model Design	AI model is unclear or does not function as intended	AI model works but has limited logic or application	AI model is functional, relevant, and well-aligned to NPC behavior
Python Implementation	Code contains many errors or lacks clarity	Code runs with some issues; logic is partially explained	Python code is clean, efficient, and clearly documented

Technical Documentation	Missing key components or vague explanation	Includes most elements; lacks depth or clarity	Clearly written, well-organized with thorough explanations of model behavior
Ethical Discussion	Superficial or missing ethical reflection	Addresses some concerns; lacks depth	Insightful analysis of ethical implications of AI behavior in games
Collaboration & Presentation	Limited collaboration or unclear presentation	Group presents project with basic collaboration shown	Group works cohesively and presents findings clearly and confidently

Total: /15