**MAJOR PROJECT**

**SYNOPSIS on**

**SNAKE SENSEI**

Submitted By:

|  |  |  |
| --- | --- | --- |
| **Name** | **Roll No** | **Branch** |
| Shahan Ali | R2142211305 | CSE GG |
| Richa Yadav | R2142211299 | CSE CCVT |



School of Computer Science

**UNIVERSITY OF PETROLEUM AND ENERGY STUDIES**

**Dehradun-248007 2024 25**

**Activity Cordinator : Bhupender sir (GG) , Sandeep pratap sir(CCVT)**

Alind Sir

**Project Guide Department Head**

Table of Contents

1. **Project Title**
2. **Abstract**
3. **Introduction**
4. **Literature Review**
5. **Problem Statement**
6. **Objective**
7. **Methodology**
8. **System Requirements**
9. **Schedule (Pert Chart)**
10. **References**

**School of Computer Science**

University of Petroleum & Energy Studies, Dehradun

**Synopsis Report (2024-25)**

* **1-Project Title**
* Snake Sensei : a reinforcement machine learning classical snake game
* **2- Abstract**

This project focuses on developing a reinforcement learning model to play the classic Snake game using Python, PyTorch, and Pygame. The aim is to create an AI agent that can learn and improve its performance in the game through trial and error, leveraging deep Q-learning techniques. The project will explore the fundamentals of reinforcement learning, neural network design, and game development, providing a comprehensive understanding of AI in gaming.

* **3- Introduction-**

The Snake game is a classic example of a simple yet engaging game that has been used extensively in AI research. This project aims to implement a reinforcement learning model to play Snake, using Python for programming, PyTorch for neural network implementation, and Pygame for game development. The project will cover the basics of reinforcement learning, the setup of the game environment, and the training of the AI agent.

* **4- Literature Review**

Reinforcement Learning: This area of machine learning involves training agents to make sequences of decisions by rewarding desired behaviors and punishing undesired ones. Key references include “Reinforcement Learning: An Introduction” by Sutton and Barto.

Deep Q-Learning: A popular algorithm in reinforcement learning that combines Q-learning with deep neural networks. Key references include “Playing Atari with Deep Reinforcement Learning” by Mnih et al.

Game Development with Pygame: Pygame is a set of Python modules designed for writing video games. Key references include “Making Games with Python & Pygame” by Al Sweigart.

* **5- Problem Statement**
* Lack of Practical AI Applications in Simple Games: Many AI projects focus on complex games, leaving a gap in the application of AI to simpler, classic games like Snake.
* Need for Educational Resources: There is a need for comprehensive educational resources that combine AI, game development, and practical coding skills.
* **6- Objectives**
* Develop a Reinforcement Learning Model: Create an AI agent capable of playing Snake using deep Q-learning.
* Implement the Game Environment: Use Pygame to develop the Snake game environment.
* Train and Evaluate the AI Agent: Train the AI agent and evaluate its performance in the game.
* **7- Methodology**
* Setup the Game Environment: Develop the Snake game using Pygame.
* Implement the AI Agent: Use PyTorch to create a neural network for the AI agent.
* Train the AI Agent: Apply deep Q-learning techniques to train the AI agent.
* Evaluate Performance: Test the AI agent’s performance and make necessary adjustments.
* **8- System Requirements (Software/Hardware)**
* Hardware Interface:
* 64-bit processor architecture supported by Windows.
* Minimum 4 GB RAM.
* Required input and output devices.
* Sufficient graphic card for rendering the game.
* Software Interface:
* Python 3.x
* PyTorch
* Pygame
* **9-Schedule (Pert Chart)**



* **References**
* Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction.
* Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., … & Hassabis, D. (2015). Playing Atari with Deep Reinforcement Learning.
* Sweigart, A. (2012). Making Games with Python & Pygame.