Detective Escape Game

Introduction

The Detective Escape Game is an interactive Python-based game that combines storytelling, decision-making, and artificial intelligence to create an engaging user experience. The player assumes the role of a detective investigating a murder in a mansion. The game challenges players to interrogate suspects, collect evidence, and deduce the identity of the killer.

Artificial intelligence in this project is designed to monitor player choices, evaluate clues, and maintain hidden suspicion scores for each suspect. This simple AI logic provides subtle guidance to the player without explicitly revealing the outcome, thereby making the game both challenging and immersive.

Algorithm

The AI in this game is a decision-support system that tracks player interactions and evaluates suspects based on clues and interrogation observations.

1. Initialization:

Create a dictionary AI_scores for all suspects (e.g., Meera, Kabir, Raghav).

Set initial scores to zero.

2. During Interrogation:

Each suspect interaction provides hints about their potential guilt.AI increases scores slightly based on observations such as nervousness, inconsistencies, or suspicious behavior.

3. Clue Collection Phase:

Each clue collected by the player affects one or more suspects.

AI updates scores according to the relevance of the clue.

4. Suspicion Analysis:

At the end of clue collection, AI compares all scores.

Highest scoring suspect is considered the AI's predicted killer.

5. Final Accusation:

Player chooses the suspect to accuse.

AI evaluates the player's choice against the actual killer and provides feedback.

Game Structure

The game is divided into sequential scenes guiding the player through the investigation:

1. Introduction: Sets the murder scenario and objective; investigation begins with a button click.

2. Interrogations: Three suspects—Meera, Kabir, and Raghav—are interrogated in sequence. Each provides an alibi and subtle AI hints to help the player assess suspicion.

3. Clue Collection: Players examine three main clues and can proceed to the next clue or accuse a suspect at any point. AI updates suspicion scores based on choices.

4. Accusation: The player selects a suspect. AI evaluates the choice against the hidden correct answer.

5. Ending: Shows whether the accusation is correct and provides a summary of AI analysis.

Additional Features

AI Hints: Subtle indications of contradictions or suspicious behavior without revealing the solution.

Dynamic Session Management: Tracks clues, scores, and progress until game completion.

Mobile-Friendly GUI: Layout adapts to desktop and mobile screens.

Conclusion

The AI Detective Escape Game integrates simple AI logic into an interactive narrative, showing how player decisions can influence outcomes without giving away the solution. Using Python and Streamlit, the project offers both a fun and educational experience, with potential for multiple endings, more suspects, and enhanced AI analysis in future versions.