Mariano Antonio Regalado Iglesias
Student ID: REG23556163
Last day and time of update: 25/03/25 at 16:46

Wildlife Preservation Strategy Simulator Report

Overview

This is a Python-based educational tool designed to simulate strategic decision-making in conservation. It uses as a concept of the classic Tic-Tac-Toe game, where the player impersonates a conservationist, and the AI opponent represents environmental threats such as urbanization or invasive species. This analogy provides a simplified platform to explore biodiversity protection concepts in a virtual format.

The simulator challenges users to consider real-world implications of conservation planning through playful interaction, while introducing technical concepts like artificial intelligence and game logic in programming.

You can find everything that there is to know on my GitHub repository:

https://github.com/marianor03/Wildlife-Preservation-Strategy-Simulator

How the Program Works

The simulator operates on a 3x3 grid representing habitat zones. The player (using "X") makes conservation decisions like habitat restoration or pollinator introduction, while the AI (using "O") plays the role of environmental pressures that threaten biodiversity. The goal is to line up three conservation efforts before environmental threats dominate.

The game flow includes:

- An introduction explaining the conservation analogy
- A **coin toss** (heads or tails) to determine who goes first
- A turn-by-turn system that updates the board
- A **minimax-based AI** that strategically responds to the player's moves
- A final outcome message indicating success, tie, or failure in habitat preservation

Technical Features

- **Python Core Features**: The program uses fundamental Python structures such as lists, loops, conditionals, and functions.

- **Minimax Algorithm**: A recursive decision-making algorithm is implemented to simulate intelligent AI behavior. It explores all possible game states to choose the move with the best worst-case outcome (Russell & Norvig, 2021).
- **Formatted Grid Display**: A clean, index-labeled console grid helps the player easily select moves and visualize their strategy.
- **Turn Logic with Input Validation**: The game handles invalid moves or inputs smoothly, ensuring usability and robustness.
- **Game State Tracking**: Functions are used to check for winning conditions or a tie after every turn.
- **User Experience**: Includes prompts, icons (emojis), and natural language messages to engage players and explain the game's results clearly.

Ethical and Educational Reflections

This project blends programming with **ecological awareness**, aligning technical education with social and environmental responsibility. By personifying the conservationist and environmental threats, users are encouraged to empathize with the real-world complexity of biodiversity protection. The simulator doesn't just teach algorithms — it invites reflection on human impact and stewardship of the natural world.

Ethically, this simulation avoids oversimplifying environmental issues. It deliberately stays abstract but grounded in truth: just as in the real world, even the best plans may result in compromise (a tie), or setbacks (a loss). This creates space for understanding that conservation is both urgent and uncertain — a concept backed by environmental ethics literature (Leopold, 1949).

Reflections and Areas for Improvement

While the simulator successfully combines conservation themes with strategic AI, there are several areas for potential improvement:

- **Habitat Types**: Cells could be labeled with real-world ecosystems (wetland, forest, etc.), affecting gameplay logic.
- **Real Data Integration**: Including datasets on species trends or urban development would deepen the realism.
- **Scoring System**: A biodiversity index or health score could quantify the impact of each move.

- **Accessibility & Interface**: Expanding to a GUI (Graphical User Interface) would make the game more user-friendly for a wider audience.
- **Dynamic Difficulty**: Currently, the AI plays perfectly. Introducing adjustable difficulty would support varied player skill levels.

These enhancements would make the simulator more immersive, informative, and adaptable for environmental education.

Conclusion

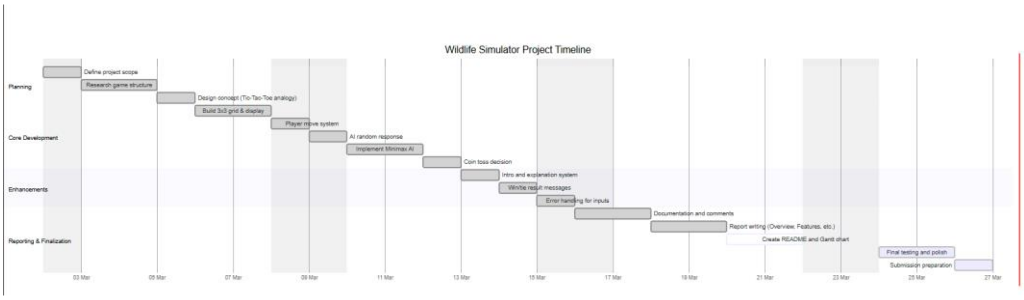
The Wildlife Preservation Strategy Simulator merges coding, strategy, and conservation into an engaging and educational experience. It demonstrates how computer science can model real-world systems and teach complex ideas through simple interfaces. By applying the minimax algorithm and user-driven decision-making, it not only strengthens programming skills but also fosters environmental thinking. With further development, this simulator could be used in classrooms or workshops to spark dialogue about biodiversity, sustainability, and responsible development.

References

- Leopold, A. (1949). *A Sand County Almanac*. Oxford University Press.
- Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson.
- Convention on Biological Diversity. (n.d.). *What is biodiversity?*. Retrieved from https://www.cbd.int/
- IPBES. (2019). *Global Assessment Report on Biodiversity and Ecosystem Services*. https://ipbes.net/global-assessment

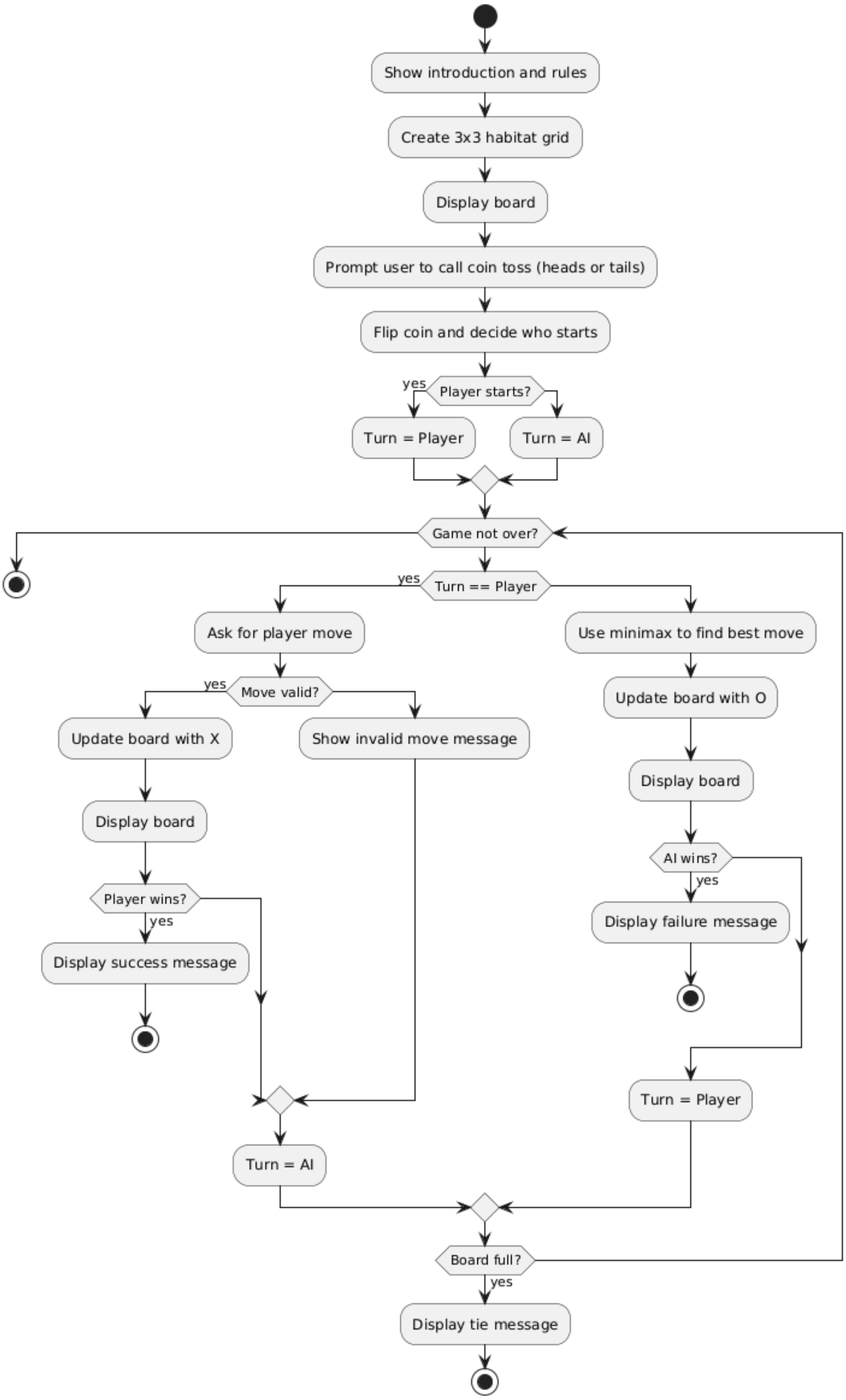
Gallery

Gantt Chart:



Mariano Antonio Regalado Iglesias
Student ID: REG23556163
Last day and time of update: 25/03/25 at 16:46

Flow Chart:


Show introduction and rules
Create 3x3 habitat grid
Display board
Prompt user to call coin toss (heads or tails)
Flip coin and decide who starts
yes
Player starts?
Turn = Player
Turn = AI
Game not over?
yes
Turn == Player
Ask for player move
Use minimax to find best move
yes
Move valid?
Update board with X
Show invalid move message
Update board with O
Display board
Display board
AI wins?
yes
Player wins?
yes
Display failure message
Display success message
Turn = Player
Turn = AI
Board full?
yes
Display tie message