Hangman Game

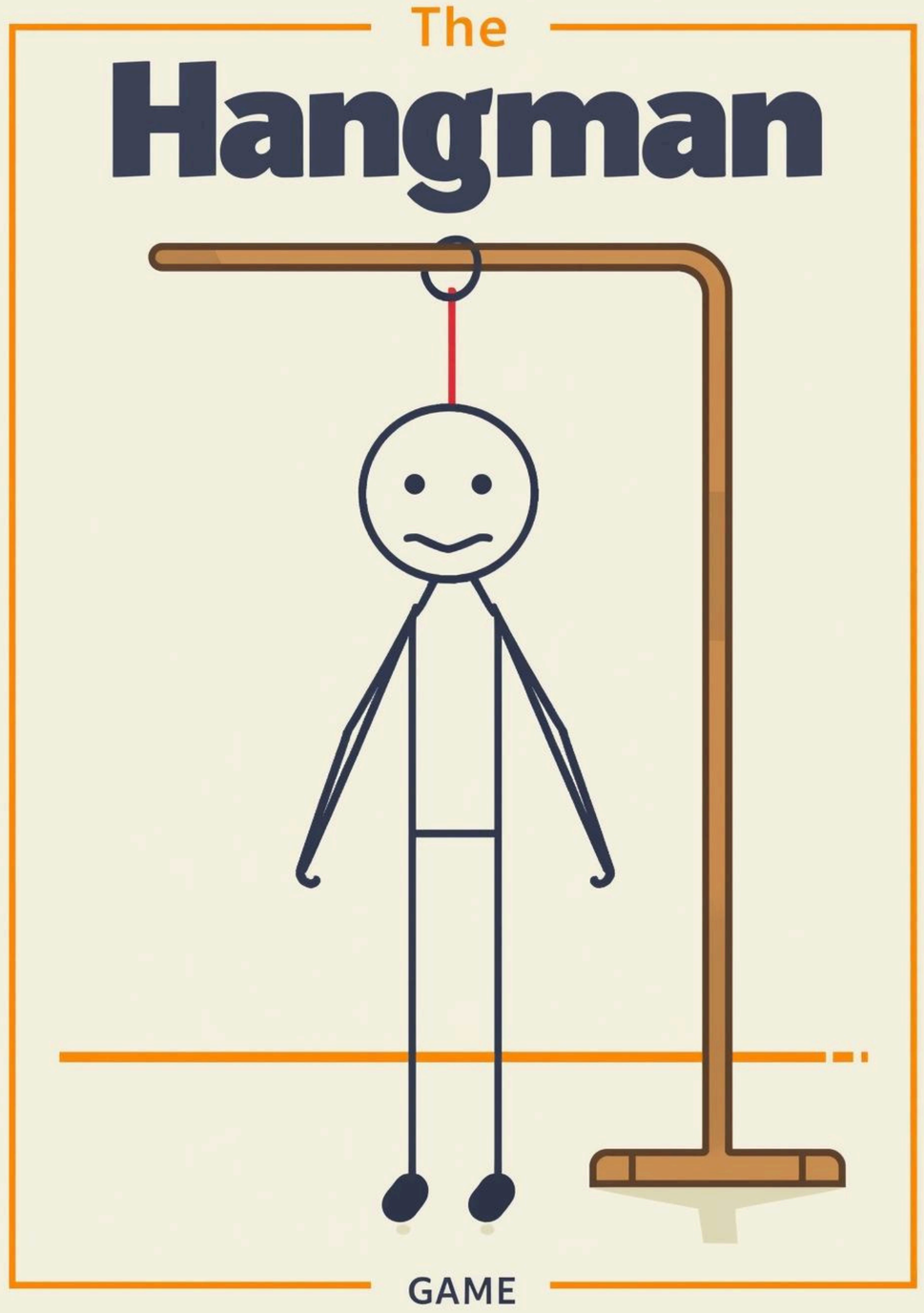
Console-Based Word Guessing Game in Python

Krishnav Rastogi | Mahek Desai
Symbiosis Artificial Intelligence Institute

Introduction to Hangman

A classic word-guessing game where players guess letters one at a time.

Each wrong guess progresses the hangman drawing. Guess the word before the hangman is complete-six wrong guesses and it's game over.



Project Aims

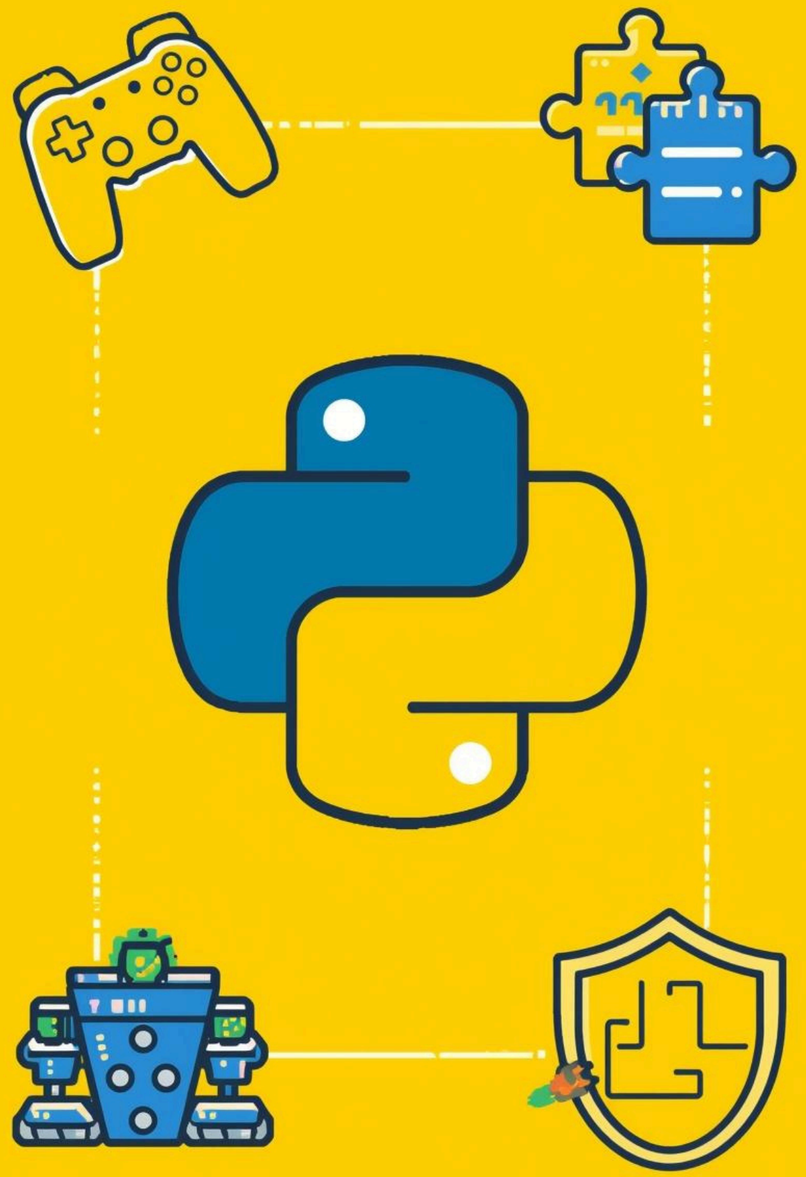
Develop a fully functional hangman word-guessing game in Python

Implement proper game logic with clear win/loss conditions

Create user-friendly terminal interface with visual feedback

Ensure comprehensive input validation and error handling

Demonstrate modular programming with clean code structure



Key Features of Hangman

ASCII ART DISPLAY

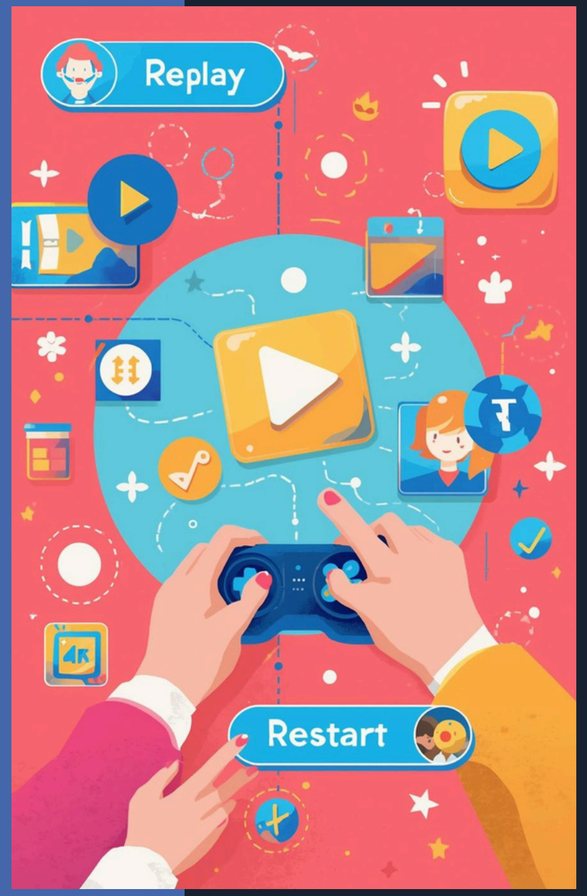
Illustrative design

REPLAY FUNCTIONALITY

User engagement

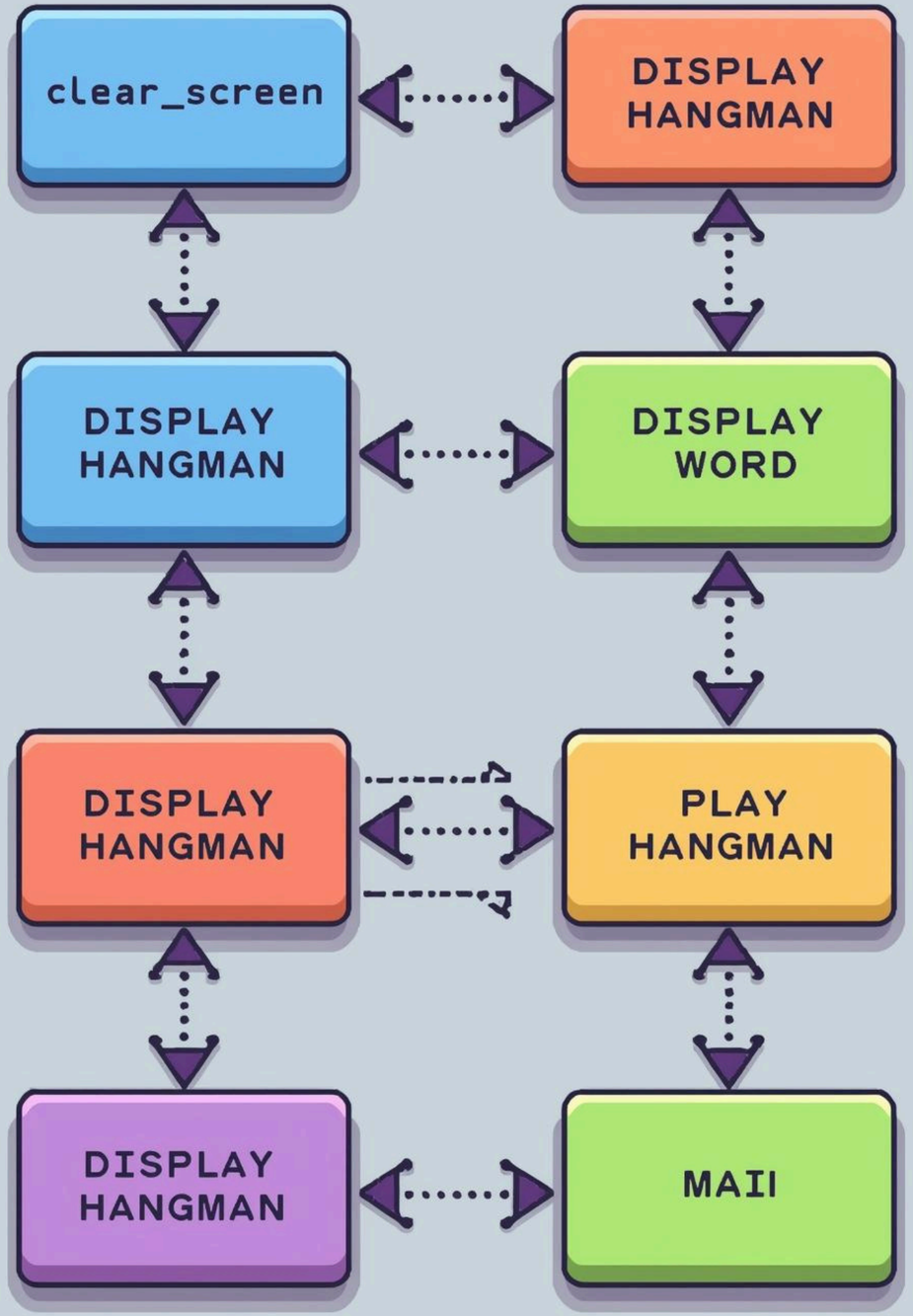
THEMED WORD BANK

Diverse vocabulary



System Architecture

This section outlines:
the core functions of our Hangman game's system architecture,
emphasizing modularity,
and interaction among functions for efficient gameplay and maintainability.

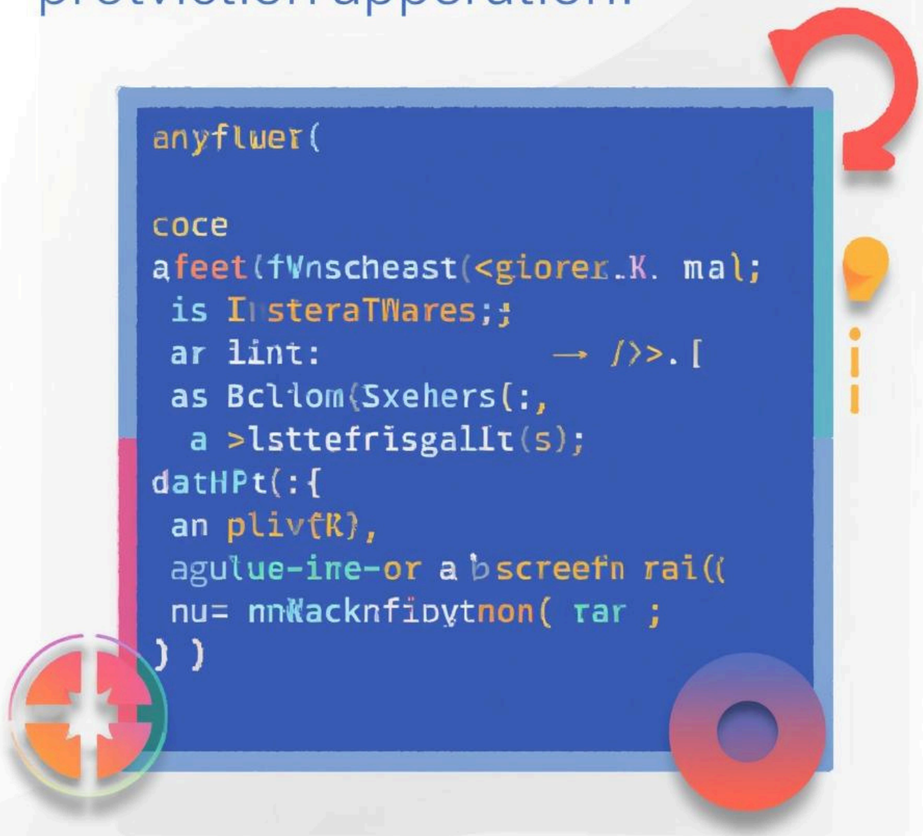


Screen Management

Effective screen management is crucial for a seamless gaming experience.

This function detects the operating system and clears the console, ensuring clarity and organization during gameplay.

Screen. Management
Functimiste Micktingj, man fon protviction apperation:

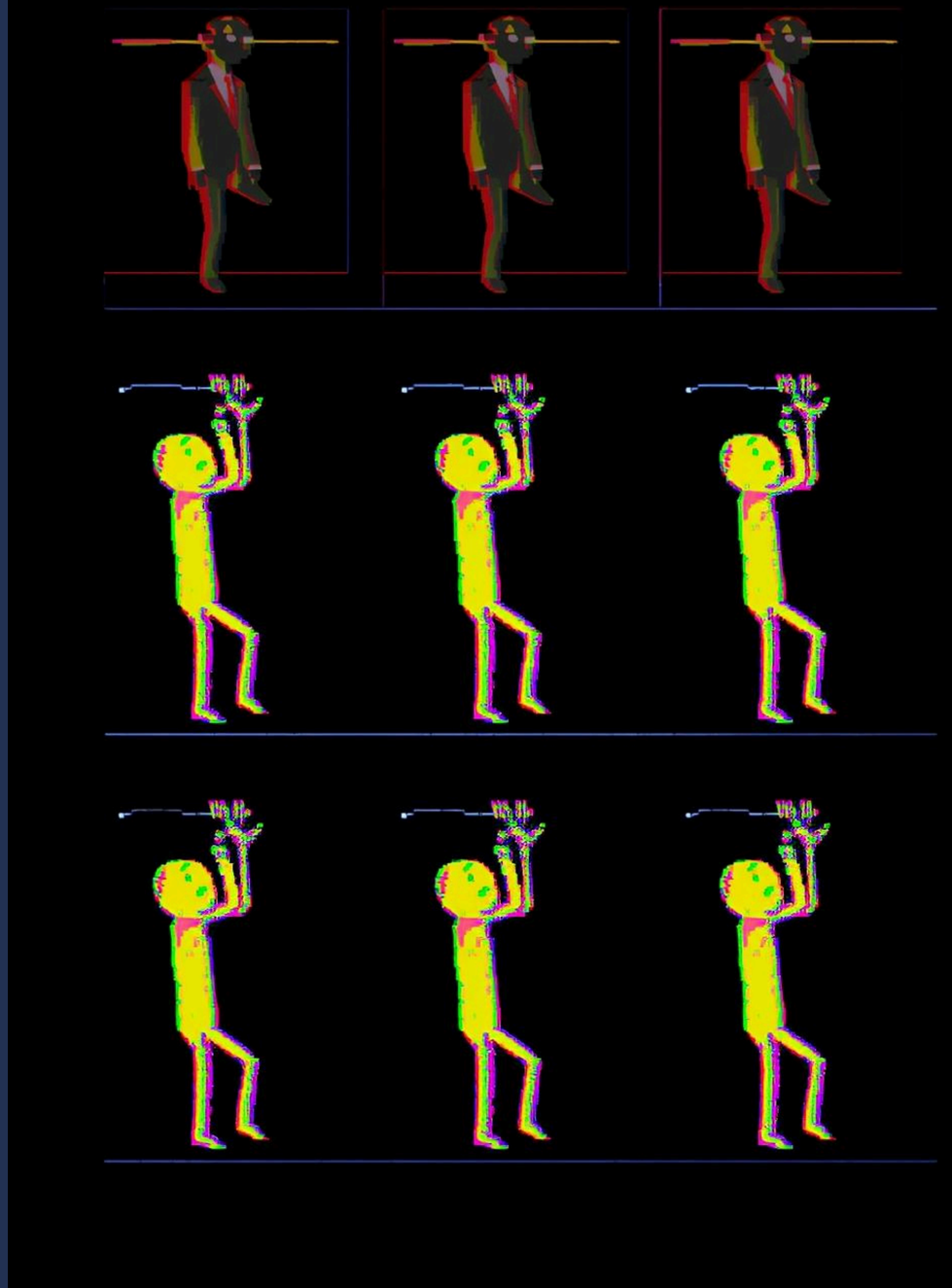


Scree n-Management

Rueck lworin Pythin of Scnes

ASCII Art Display

The Hangman game features dynamic ASCII art that evolves with each incorrect guess, enhancing the visual experience and engaging players as they attempt to save the hangman.



Game Flow Logic

The game follows a structured flow including setup, gameplay loop, and end game processing. This ensures smooth transitions and engaging interaction throughout the Hangman experience.

Input Validation

Effective input validation is crucial for ensuring user input is processed correctly, enhancing gameplay experience, and maintaining game integrity through robust error handling and clear feedback.

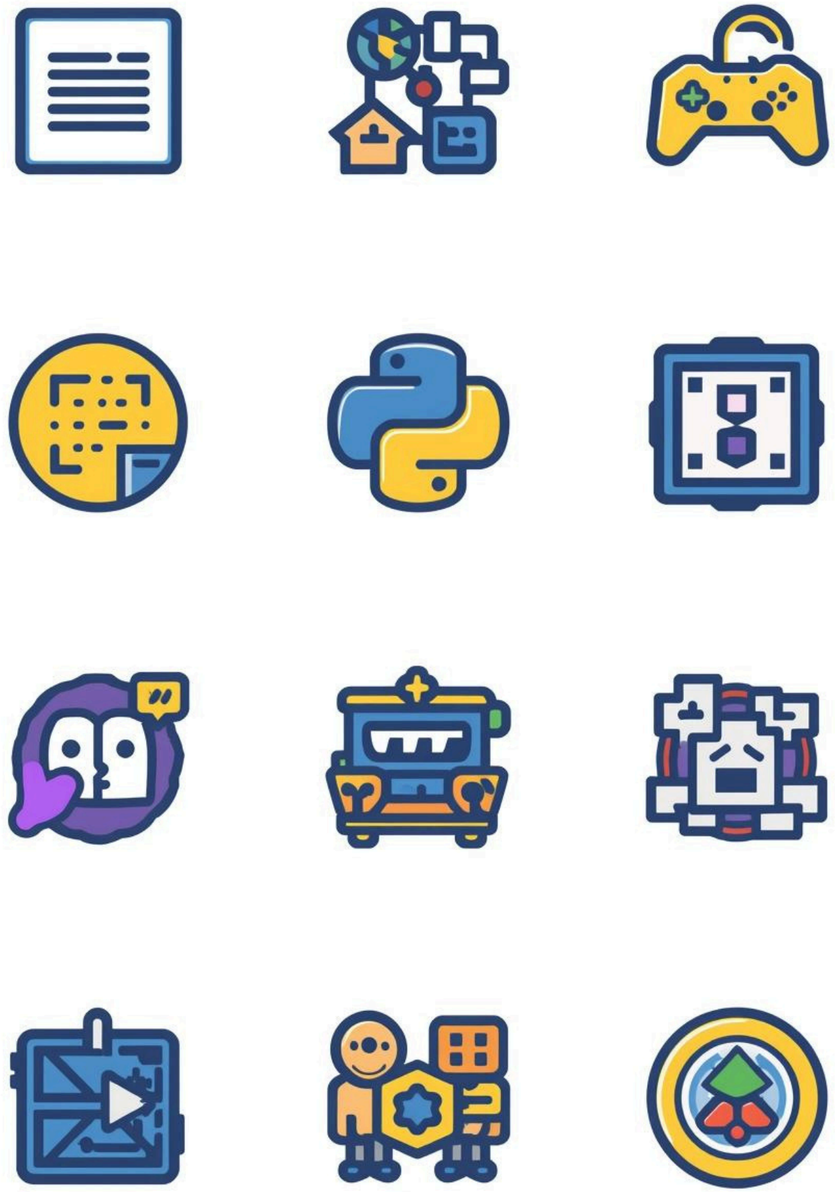
100%

Testing Success Rate

All 53 test cases passed successfully, demonstrating the robustness and reliability of the Hangman game implementation. This ensures a smooth gaming experience for users of all levels.

Project Achievements

This section highlights our clean code practices and interactive gameplay features that enhance user experience, ensuring both reliability and engagement throughout the game process.



Next Steps

GUI DEVELOPMENT

Creating a graphical user interface using Tkinter or Pygame to improve user engagement and experience.

EXPANDED WORD BANK

Incorporating a larger variety of words to increase gameplay complexity and fun, enhancing player learning.

ENHANCED DIFFICULTY LEVELS

Introducing different difficulty levels to cater to players of varying skills, promoting a more personalized experience.

Thank You!

PROJECT AUTHORS
KRISHNAV RASTOGI AND MAHEK DESAI

INSTITUTION NAME
SYmbiosis Artificial Intelligence Institute

GITHUB REPOSITORY
https://github.com/rastogikrishnav085-hue/hangman-python-game.git