# HANGMAN GAME

MINOR PROJECT REPORT

By

**SURYA KP (RA2211026010378)**

**KOUSHIK VISHAL (RA2211026010384)**

**AKASH RV (RA2211026010376)**

Under the guidance of   
**Dr Maivizhi R***In partial fulfilment for the Course*

of

**21CSC203P – ADVANCED PROGRAMMING PRACTICE**

in Computer Science – Artificial Intelligence and Machine Learning



**FACULTY OF ENGINEERING AND TECHNOLOGY**

**SCHOOL OF COMPUTING**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**KATTANKULATHUR**

**NOVEMBER 2023**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**(Under Section 3 of UGC Act, 1956)**

**BONAFIDE CERTIFICATE**

Certified that this minor project report for the course **21CSC203P** **ADVANCED PROGRAMMING PRACTICE** entitled in "**Hangman game**" is the bonafide work of **Surya KP (RA2211026010378), Koushik Vishal (RA2211026010384)** and **Akash RV (RA2211026010376)** who carried out the work under my supervision.

# PROJECT GUIDE HEAD OF THE DEPARTMENT

Dr Maivizhi R Dr Annie Uthra

# Assistant Professor Professor & Head

**Department of Computational Intelligence Department of Computational Intelligence**

SRM Institute of Science and Technology SRM Institute of Science and Technology

Kattankulathur Kattankulathur

# ABSTRACT

The Python-based Hangman game stands as a compelling fusion of traditional wordplay and modern interactive gaming, promising an enriching and enjoyable experience for players of all ages. Rooted in the classic pen-and-paper pastime, this digital adaptation combines nostalgia with innovation, offering an engaging platform for word puzzle enthusiasts. In this rendition, players embark on a linguistic journey, attempting to unveil a concealed word by suggesting individual letters. Each incorrect guess brings them closer to the gradual creation of a hangman figure, ratcheting up the suspense. The ultimate goal is to solve the word before the hangman is fully illustrated. The game's user-friendly interface ensures accessibility, catering to a diverse audience, from casual gamers looking for entertainment to language aficionados seeking intellectual stimulation. The Hangman game isn't merely a source of amusement; it's also an educational tool for language improvement. Players can refine their vocabulary, enhance spelling skills, and stimulate critical thinking through the captivating gameplay. Moreover, the Python-based Hangman game fosters a sense of accomplishment as players successfully decipher words, providing a unique blend of entertainment and mental exercise. This abstract invites players to delve into a world where letters become clues and solving riddles turns into an exhilarating adventure.

# ACKNOWLEDGEMENT

We express our heartfelt thanks to our honorable **Vice Chancellor Dr. C. MUTHAMIZHCHELVAN**, for being the beacon in all our endeavors.

We would like to express my warmth of gratitude to our **Registrar Dr. S. Ponnusamy,** for his encouragement.

We express our profound gratitude to our **Dean (College of Engineering and Technology) Dr. T. V.Gopal,** for bringing out novelty in all executions.

We would like to express my heartfelt thanks to Chairperson, School of Computing **Dr. Revathi Venkataraman,** for imparting confidence to complete my course project

We wish to express my sincere thanks to **Course Audit Professors Dr. Vadivu. G , Professor, Department of Data Science and Business Systems and Dr. Sasikala. E Professor, Department of Data Science and Business Systems** and **Course Coordinators** for their constant encouragement and support.

We are highly thankful to our my Course project Faculty **Dr Maivizhi R , Assistant Professor , Department of Computational Intelligence** for herassistance, timely suggestion and guidance throughout the duration of this course project.

We extend my gratitude to our **HoD Dr Annie Uthra , Professor & Head, Department of Computational Intelligence** and my Departmental colleagues for their Support.

Finally, we thank our parents and friends near and dear ones who directly and indirectly contributed to the successful completion of our project. Above all, I thank the almighty for showering his blessings on me to complete my Course project.

**TABLE OF CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **CHAPTER NO** | | **CONTENTS** | **PAGE NO** |
| **1** | **INTRODUCTION** | |  |
|  | 1.1 Motivation | |  |
|  | 1.2 Objective | |  |
|  | 1.3 Problem Statement | |  |
|  | 1.4 Challenges | |  |
| **2** | **LITERATURE SURVEY** | |  |
| **3** | **REQUIREMENT ANALYSIS** | |  |
| **4** | **ARCHITECTURE & DESIGN** | |  |
| **5** | **IMPLEMENTATION** | |  |
| **6** | **EXPERIMENT RESULTS & ANALYSIS** | |  |
| **7** | **CONCLUSION** | |  |
| **8** | **REFERENCES** | |  |

1. **INTRODUCTION**

**Motivation:**

The creation of the Hangman game in Python was motivated by a desire to preserve the charm of a beloved word game while introducing it to a new generation of players. Word games have long been cherished for their ability to entertain, educate, and challenge players of all ages. This game aims to capture the essence of this classic pastime and make it accessible to a broader audience in a digital format. The motivation lies in fostering a love for wordplay, language skills, and mental agility.

**Objective:**

The primary objective of the Python Hangman game is to provide players with an entertaining and educational platform for honing their word-related skills. Players are invited to solve concealed words by guessing letters, employing their vocabulary, spelling, and deduction abilities. The game also seeks to create a sense of achievement and satisfaction when players successfully decipher words, adding a competitive edge to the gaming experience. Through this objective, the game bridges the gap between entertainment and mental exercise.

**Problem Statement:**

The challenge that the Hangman game addresses is two-fold. Firstly, it seeks to offer an engaging and user-friendly digital version of a traditional game while retaining the core elements that make Hangman so beloved. Secondly, the game endeavors to provide an effective and enjoyable tool for enhancing language skills, vocabulary, and cognitive abilities. The problem statement revolves around striking the right balance between entertainment and education, creating an intuitive interface, and offering an enjoyable gaming experience.

**Challenges:**

The development of the Python Hangman game was not without its challenges. Challenges included designing an appealing and interactive user interface, ensuring a fair and engaging word selection mechanism, and implementing a smooth gaming experience.

1. **LITERATURE SURVEY**

**1. Educational Benefits of Word Games:**

A study by Plass and Jones (2019) highlighted the educational benefits of word games like Hangman. These games can improve vocabulary, language skills, and cognitive abilities in players of all ages.

**2. The Use of Hangman Game in Motivating Students in Learning English:**

Academic Journal Perspective Education Language and Literature

Author: Rika Mandasari Manan

**2. Gamification in Education:**

The concept of gamification in education has gained attention in recent years. Gamification involves using game elements, such as points, challenges, and rewards, to engage and motivate learners. Hangman and similar games can be seen as a form of gamification in language learning.

**3. Player Engagement and Motivation:**

Research by Steinkuehler and Duncan (2008) discussed how video games, including word games, can engage and motivate players. The Hangman game's interactive and competitive nature can enhance player engagement.

**4. ASCII Art in Games:**

ASCII art, as used in the Hangman program, is a unique and creative way to represent visuals in text-based games. While not specific to Hangman, studies on ASCII art in gaming can shed light on its impact on player experience.

**5. Code Structure and Game Development:**

Numerous books and resources discuss game development in Python, providing insights into code structure, game design, and user interface. These resources can inform the technical implementation of the Hangman game program.

**6. User-Centered Design in Games:**

The user experience in game design is crucial. Studies on user-centered design principles in game development can guide the creation of player-friendly interfaces and engaging gameplay, as observed in the Hangman game.

**7. Accessibility in Games:**

Research on accessibility in video games is essential to ensure that games, including word games like Hangman, can be enjoyed by players with diverse abilities. It's important to make games accessible to a wide audience.

1. **REQUIREMENTS** 
   1. **Requirement Analysis:**

**Functional Requirements:**

**Game Rules:**

* Word Selection: The game should pick a random word.
* Player Input: Players input single letters for guessing.
* Word Reveal: Show correct letters in the hidden word.
* Incorrect Guess Handling: Wrong guesses draw the hangman.
* Win Condition: Recognize when the word is guessed correctly.
* Loss Condition: Identify when the hangman figure is drawn.

**User Interface:**

* Display: Show the hangman, concealed word, and guessed letters.
* Input: Accept player input.
* Feedback: Provide feedback on guesses.

**Game Logic:**

* Hangman Stages: Create stages for drawing the hangman.
* Word Selection: Fair word selection.
* Word Difficulty: Include different difficulty levels.
* Word Reveal: Reveal guessed letters correctly.

**Non-Functional Requirements:**

**Performance:**

* Responsiveness: Promptly respond to user input.
* Smooth Gameplay: Ensure smooth gameplay.

**User Experience:**

* Intuitiveness: Make the game easy to understand.
* Visual Appeal: Create an engaging user interface.

**Educational Value:**

* Vocabulary Building: Enhance player vocabulary.
* Cognitive Development: Stimulate critical thinking**.**
  1. **Hardware Requirement:**
* Personal computer or mobile device
* Memory (RAM): 2 GB or more
* Storage: 50 MB of available space
* Compiler: Any python compiler
* Input: Keyboard or touch input
* Internet connection (for updates and online features, if any)

1. **ARCHITECTURE AND DESIGN**

**1. Software Architecture:**

The Hangman game follows a simple yet effective architectural pattern:

* **Client-Server Architecture:** The game primarily operates on a standalone client architecture, where the player interacts with the game locally. There is no need for a centralized server as the game does not involve multiplayer or network gameplay.

**2. Game Design:**

**2.1. User Interface (UI) Design:**

* **Visual Elements:** ASCII art is used to display the hangman figure, and the concealed word and guessed letters are presented clearly.
* **Text-Based Feedback:** The game provides text-based feedback on player input and the game's progress.

**2.2. Game Logic Design:**

* **Word Selection:** A random word is selected from a predefined word list at the beginning of each game.
* **Game Flow:** The game progresses by accepting player input, validating it, and updating the game state accordingly.
* **Win and Lose Conditions:** The game logic determines the conditions for winning (guessing the word) and losing (drawing the complete hangman).

**2.3. Data Structures:**

* **Word List:** A list of words from which random words are selected.
* **Player Data:** Player-specific data such as scores or game progress can be stored in data structures as needed.

**2.4. Code Organization:**

* The code is organized into functions and classes for clarity and modularity.
* Key functions include word selection, input validation, hangman figure drawing, and game state management.

**2.5. User Experience (UX):**

* The game design focuses on providing a simple and intuitive user experience, allowing players to focus on word-guessing and gameplay.
* Feedback is given at each step to inform players of their progress.

**3. Design Challenges:**

* **User Interface:** Designing an engaging and informative console-based UI, incorporating visual elements like ASCII art, and handling user input and feedback effectively.
* **Word Selection:** Ensuring fair and unbiased word selection from the word list.
* **Game Logic:** Creating a game logic that correctly manages the state of the hangman figure, word reveal, and win/lose conditions.
* **Performance:** Optimizing code and data structures for smooth gameplay.

The architecture and design of the Python Hangman game are geared toward offering a seamless and enjoyable gaming experience while maintaining simplicity and educational value. The game's clear and concise design allows it to appeal to a wide range of players, from word game enthusiasts to those seeking an educational yet entertaining pastime.

**\_\_\_\_\_\_\_**

**| |**

**| O**

**| /|\**

**| / \**

**\_\_|\_\_**

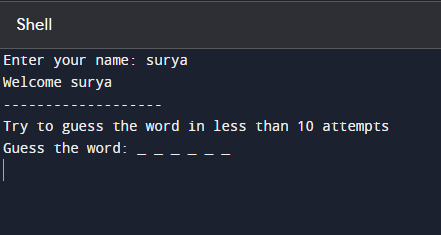
**This represents the hangman figure with different stages.**

* 1. **IMPLEMENTATION**

1. The game starts by randomly selecting a word from a predefined list of animal names, such as "panda," "lion," "tiger," and others.
2. The player is prompted to enter their name, and a welcome message is displayed.
3. The player is informed that they have ten attempts to guess the word. The game loop begins.
4. Inside the game loop, the code keeps track of the concealed word, guessed letters, and the remaining turns (attempts).
5. The game loop continues until one of the following conditions is met:
   * The player successfully guesses the word, in which case they win.
   * The player runs out of turns (attempts), in which case they lose.
6. In each iteration of the game loop, the player is presented with the current state of the concealed word with underscores representing unknown letters and any correctly guessed letters filled in. For example, "l\_\_n" might be displayed if the word is "lion."
7. The player is prompted to guess a letter. The code validates that the input is a valid lowercase letter.
8. If the guessed letter is in the word, it is revealed in the concealed word. If not, the number of remaining turns is decreased, and the hangman figure is drawn step by step. The hangman figure is displayed in ASCII art.
9. If the concealed word becomes equal to the selected word, the player wins, and the game loop exits.
10. If the player runs out of turns (turns become zero), the player loses, and the game loop exits.
11. The game informs the player of the outcome (win or lose) and displays the corresponding hangman figure.
12. The game concludes, and the player has the option to start a new game by running the hangman() function again.

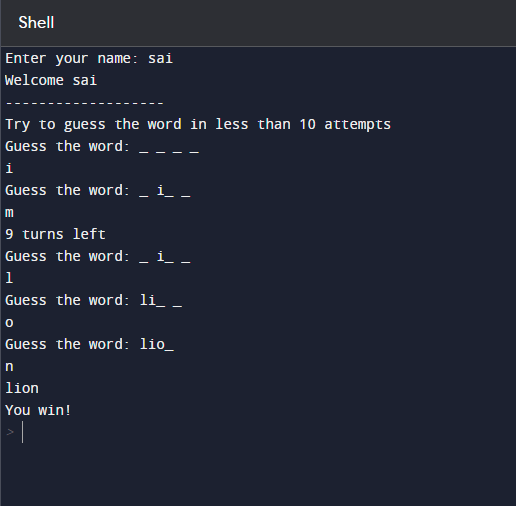
Overall, this Hangman game implementation provides an interactive and engaging way to enjoy the classic word-guessing game. Players can test their vocabulary and deduction skills while having fun. The code's graphical representation of the hangman figure adds an extra layer of visual engagement to the game.

**6. RESULTS AND DISCUSSION**

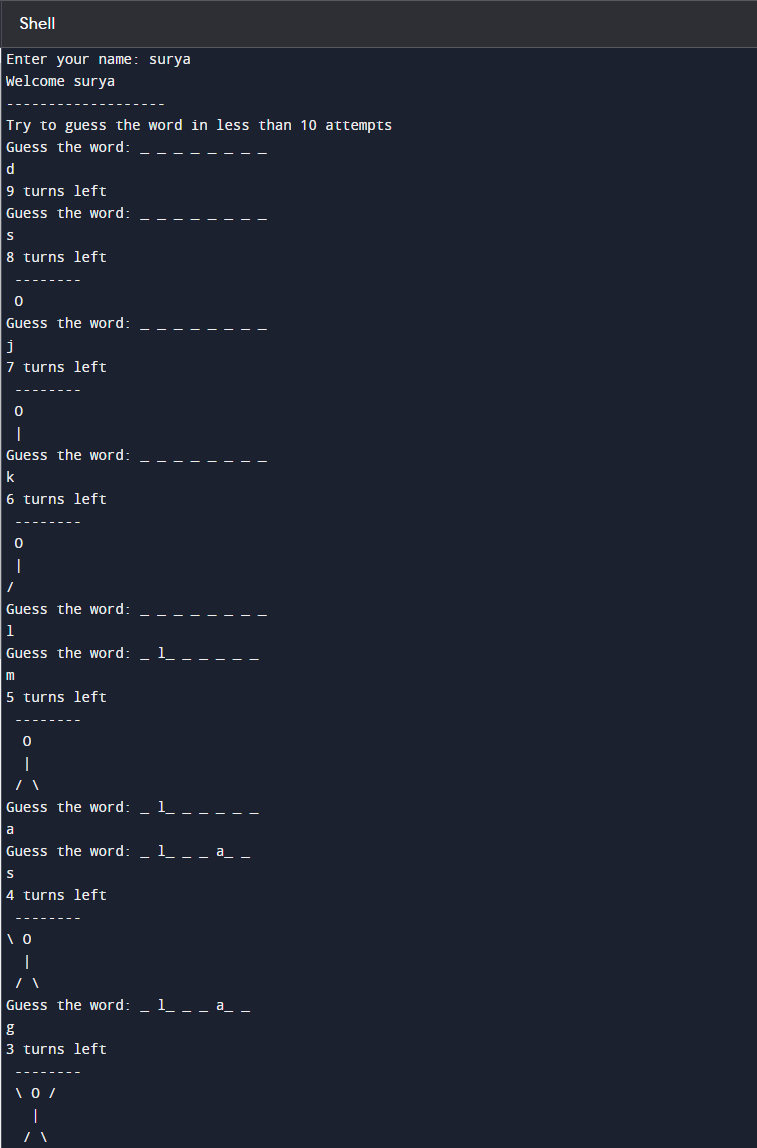
****

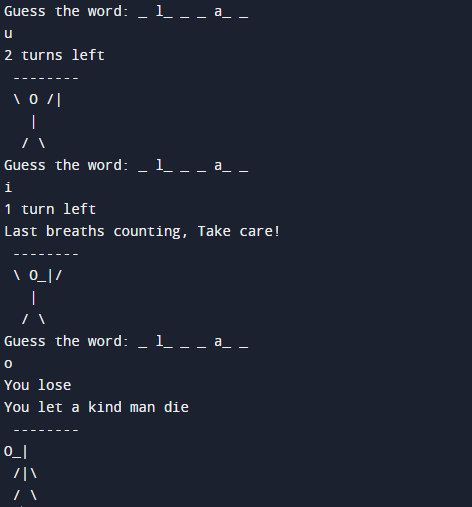
In this game, the user is first prompted to input their name, which is stored to create a personalized experience. Subsequently, a welcoming message is displayed, addressing the user by name, inviting them into the gaming environment. A random word, selected from a predefined list of words or phrases, is then presented to the player as a series of underscores, signifying the hidden letters of the word. This underscore representation allows the user to visualize the length of the word while concealing its specific letters.

As the game progresses, the player engages in a guessing loop where they input letters they believe to be part of the hidden word. With each correct guess, the underscores are updated to reveal the guessed letters in their respective positions. This cycle continues until the player successfully guesses the entire word or exhausts the maximum allowable incorrect attempts, culminating in either a triumphant reveal of the complete word or the depiction of the hangman symbol, indicating the end of the game.

****

When the user successfully guesses all the letters comprising the hidden word in the Python Hangman game, a celebratory moment ensues, marking their victory. As each correct letter is guessed and revealed in its corresponding position within the word, the excitement builds, culminating in the complete unveiling of the word. The game acknowledges the player's success with a congratulatory message, affirming their triumph. The victorious moment is accompanied by a sense of accomplishment and satisfaction as the user successfully deciphered the concealed word, showcasing their linguistic prowess and deduction skills. This achievement reinforces a positive gaming experience, fostering a sense of achievement and encouragement for the player to engage further in the game or relish their victory before embarking on the next challenge.

****

****

In this game, game, when the user exceeds the maximum allowable incorrect guesses, signifying an unsuccessful attempt to decipher the hidden word, the game acknowledges their loss. With each incorrect guess, a part of the hangman figure is gradually drawn, symbolizing the user's progress towards failure. As the hangman drawing nears completion, tension mounts, and upon reaching the maximum incorrect attempts, the game concludes, signaling the user's defeat. A message or graphical display signifies the loss, often displaying the complete hangman figure, indicating the unsuccessful attempt to solve the word. Despite the outcome, the game encourages resilience and offers an opportunity for learning from the unsuccessful attempt, motivating the user to engage again and apply new strategies or word-solving techniques in subsequent gameplay.

**7. CONCLUSION**

To conclude, the Hangman game program provides a delightful and interactive word-guessing experience, successfully capturing the essence of this classic game. Its strength lies in its capacity to engage players, prompt them to enter their names for a personalized touch, and offer a well-defined set of instructions for easy gameplay comprehension.

One notable feature is the addition of a hangman figure represented in ASCII art, which not only adds a visually engaging element but also allows players to track their progress. The game has both entertainment and educational value, as players can enhance their vocabulary while honing their problem-solving skills.

Code clarity is a strong suit of this program, with each aspect of the game clearly outlined, from word selection to win or lose conditions. This structure not only makes it easy to comprehend but also facilitates potential modifications or enhancements.

Nonetheless, it's important to note that the program is limited to a predefined list of words, specifically related to animals. Expanding the word categories and including a broader range of words would further enhance the game's appeal and replayability.

In a nutshell, the Hangman game program is a well-executed rendition of a classic word game. It offers an engaging and interactive gaming experience, promoting both entertainment and learning. With its clear structure and engaging visual elements, it provides an excellent platform for players to test their word-guessing skills, have fun, and continue to enjoy multiple rounds of play.

**8.REFERENCES**

<https://www.geeksforgeeks.org/hangman-game-python/?ref=lbp>

<https://www.pythonforbeginners.com/code-snippets-source-code/game-hangman>