

word guess game

PYTHON + TKINTER



| **SR.NO.** | **TITLE** | **PAGE** |
| --- | --- | --- |
| 1. | Abstract | 1 |
| 2. | Introduction | 1 |
| 3. | Problem Statement | 1 |
| 4. | Scope of the Project | 1 |
| 5. | Methodology | 1 |
| 6. | System Requirements | 1 |
| 7. | Implementation | 1 |
| 8. | Conclusion | 2 |
| 9. | References | 2 |

**Abstract**

This project describes a Word Guess Game developed using Python and Tkinter, where players attempt to guess a randomly selected secret word, letter by letter, within a limited number of attempts. The game features an interactive graphical interface and provides feedback after each guess, making it engaging and educational for users.

**Introduction**

Word games are recognized for enhancing cognitive and language skills. This project implements a digital version of the classic word guessing challenge, offering users an intuitive interface and fostering both enjoyment and mental exercise. The use of Python's Tkinter library ensures a user-friendly experience.

**Problem Statement**

Traditional word guessing games rely on manual operation and lack interactive features. There is a need for a digital solution that maintains the engagement of physical games while improving accessibility and convenience. The challenge is to create a game that is simple to use and provides real-time feedback to players.

**Scope of the Project**

This project is limited to implementing a graphical word guessing game using Python's standard libraries. The core features include secret word selection, input validation, dynamic display updates, and management of game states (win/loss). The project does not cover multiplayer capabilities or advanced word databases.

**Methodology**

* **Algorithm Development:** Choose a random word from a predefined list and mask its characters. Track guessed letters and update feedback.
* **Interface Design:** Use Tkinter to create the main game window, entry fields, and message displays.
* **Game Logic:** Validate player inputs, update attempts left, and show game progress interactively.

**System Requirements**

* **Software Requirements:**
  + Python (3.x and above)
  + Tkinter library (standard with Python)
* **Hardware Requirements:**
  + Any computer capable of running Python scripts

**Implementation**

The Word Guess Game script uses Tkinter for GUI creation, random module for word selection, and internal logic for input validation, attempt count, and user feedback. The process includes event handling for button clicks and updating the display based on the game's state.

**Conclusion**

The Word Guess Game provides a fun and educational experience, demonstrating key principles of interactive application design using Python. Its simplicity makes it suitable for beginner programmers and casual players, while its real-time feedback mechanism encourages repeated use.

**References**

1. Python Official Documentation
2. Tkinter Library Guide
3. Source Code: Word-Guess.py