PROJECT ON WORDLE

Tribhuvan University
Institute of Engineering
Thapathali Campus, Thapathali

TEAM MEMBERS

1. Kushal Saud
2. Yujen Shrestha
3. Yangdi Bhote Shalaka
4. Suhan Man Pradhan

Presented To:
Department of Electronics and Computer Engineering

Presented on:
March 14, 2025

Table of Contents

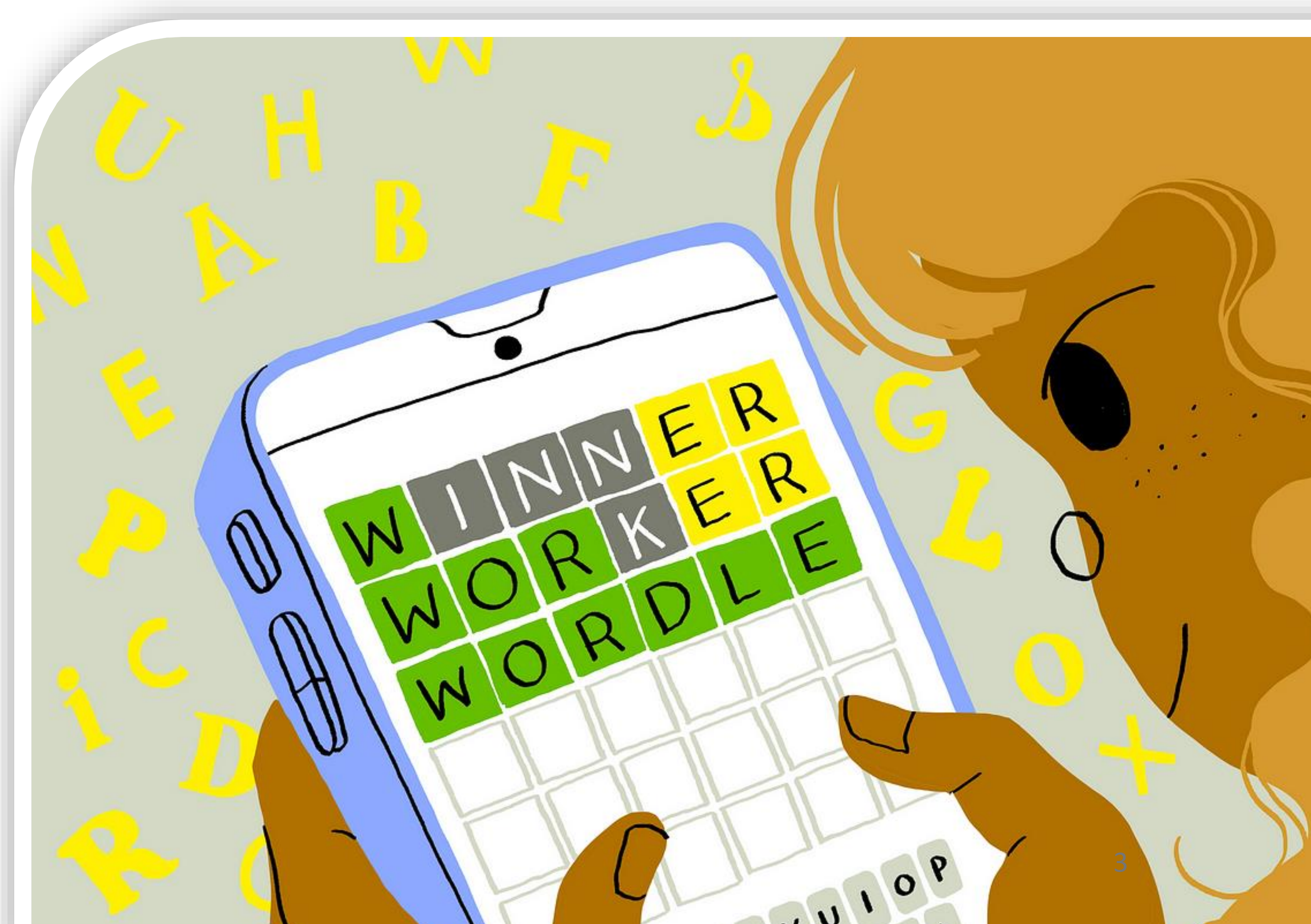
- Introduction
- Objective
- How to Play Wordle
- Project Architecture
- Challenges & Solutions
- Live Demo
- Conclusion
- Q&A

Introduction

- Wordle is a word puzzle game where players have six attempts to guess a five-letter word.

- Originally created by Josh Wardle in 2021 as a fun project for his partner

- The game quickly gained immense popularity due to social media trends



Objective

- Recreate the core gameplay of Wordle
- Develop a dynamic word list, containing all words in the Wordle glossary
- Include a stats section to track player progress
- To improve logical thinking capabilities

How to Play Wordle

1. **Guess a 5-letter Word:** You have six attempts to find the correct word

2. **Get Feedback on your Guess:**

   - **Green** – Letter is in the word and in the right spot.

   - **Yellow** – Letter is in the word but in the wrong spot.

   - **Gray** – Letter is not in the word at all.

3. **Use the Hints** – Based on the colors, adjust your next guess.

4. **Win or Lose** – You win if you guess the correct word within six tries! If not, the word is revealed at the end.

Project Architecture

- The system consists of a main page, where you can switch between four options. We can access these options by selecting them and pressing ‘Enter’.

  - ➢ **Start Game** – To start a new game
  - ➢ **How to Play** – Gives info on how the game is played
  - ➢ **Stats** – Shows the records and data of the games played
  - ➢ **Exit Game** – Exits the game

Start Game

- **startgame()** – Contains the main body. Takes in input, and prints the necessary output
- **wordpicker()** – Generates random word from “wordleanswers.txt” for user to guess using srand() and rand()
- **validword()** – Checks whether the word exists in the Wordle glossary stored in “totalwords.txt” using binary search
- **wordchecker()** – Checks the status of each letter of the inputted word to the word to guess, and assigns certain value to each letter (2 for correct, 1 for incorrect placement, 0 for incorrect letter)
- **statswin()** and **statslose()** – Edits the stats.txt file after each game based on the win/lose

Stats

- Keeps track of helpful statistics for players to keep track of their performance by storing the data in a text file “stats.txt”.
- Displays the required statistics along with a convenient bar graph to show guess distribution.
- Allows users to reset the stats data.

How to Play and **Exit Game**

- **How to Play** gives the user information on how the game is played.

- **Exit Game** exits the game.

Libraries Used

- stdio.h
- conio.h
- stdlib.h
- windows.h
- string.h
- time.h
- unistd.h

Challenges & Solution

- **Challenge** – Difficult to check validity of word among >11,000 words.
  Solution – Used Binary Search
- **Challenge** – Handling Case Sensitivity
  Solution – All inputted lowercase characters are saved immediately as uppercase characters
- **Challenge** – Providing letter feedback
  Solution – Used ‘windows.h’ to change text and background color for intuitive understanding

Live Demo

- Here, we will showcase a live demonstration of our Wordle game in action

- We will walk through how a player guesses words and receives feedback based on letter positions

Conclusion

- Our Wordle project successfully recreates the fun and challenge of the original game while enhancing our programming and problem-solving skills
- Through this project, we learned about logic building
- Wordle’s popularity comes from its simplicity and engaging mechanics, and our version reflects those core elements
- This project has been a great learning experience, and we look forward to applying these skills to future developments

ANY QUERIES

Thank you!!!