GameSpace

DISSERTATION

SUBMITTED TO MANGALORE UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF BACHELOR OF COMPUTER SCIENCE (BCA)

**A logo of a university

Description automatically generated**

**SUBMITTEDBY:**

**RAJESH K N (U05FM21S0054)**

# UNDER THE SUPERVISION AND GUIDANCE OF

PRABHAVATHI M

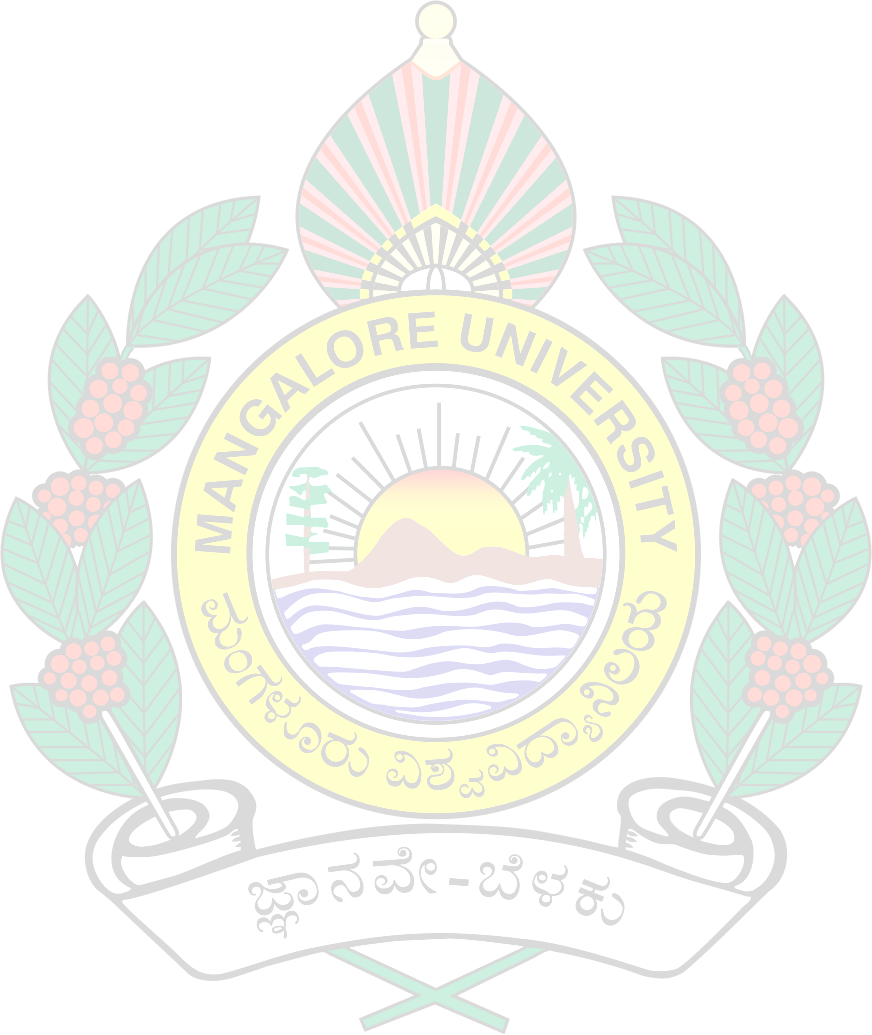
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE

FIELD MARSHAL K M CARIAPPA COLLEGE

MADIKERI-571201

2023-24

**MANGALORE UNIVERSITY**

**DEPARTMENT OF COMPUTER SCIENCE**

**FIELD MARSHAL K M CARIAPPA COLLEGE**

**[A Constituent college of Mangalore University]**

**Madikeri-571201**

**(Re-Accredited with B++ Grade by NAAC)**

**CERTIFICATE**

This is to certify that the project work entitled “**GAMESPACE**” submitted in partial fulfillment of requirement for the award of **Bachelor of Computer Applications** degree of Mangalore University is a result of the bonafede work carried out by

**RAJESH K N (U05FM21S0054)**

During the academic year 2023-24.

**Guide:**

**Ravishankar M N**

Assistant Professor and Head of the

Department of Computer Science,

Field Marshal K M Cariappa College,

Madikeri.

**Prabhavathi M**

Assistant Professor,

Department of Computer Science,

Field Marshal K M Cariappa College,

Madikeri.

**Principal**

**Field Marshal K M Cariappa College**

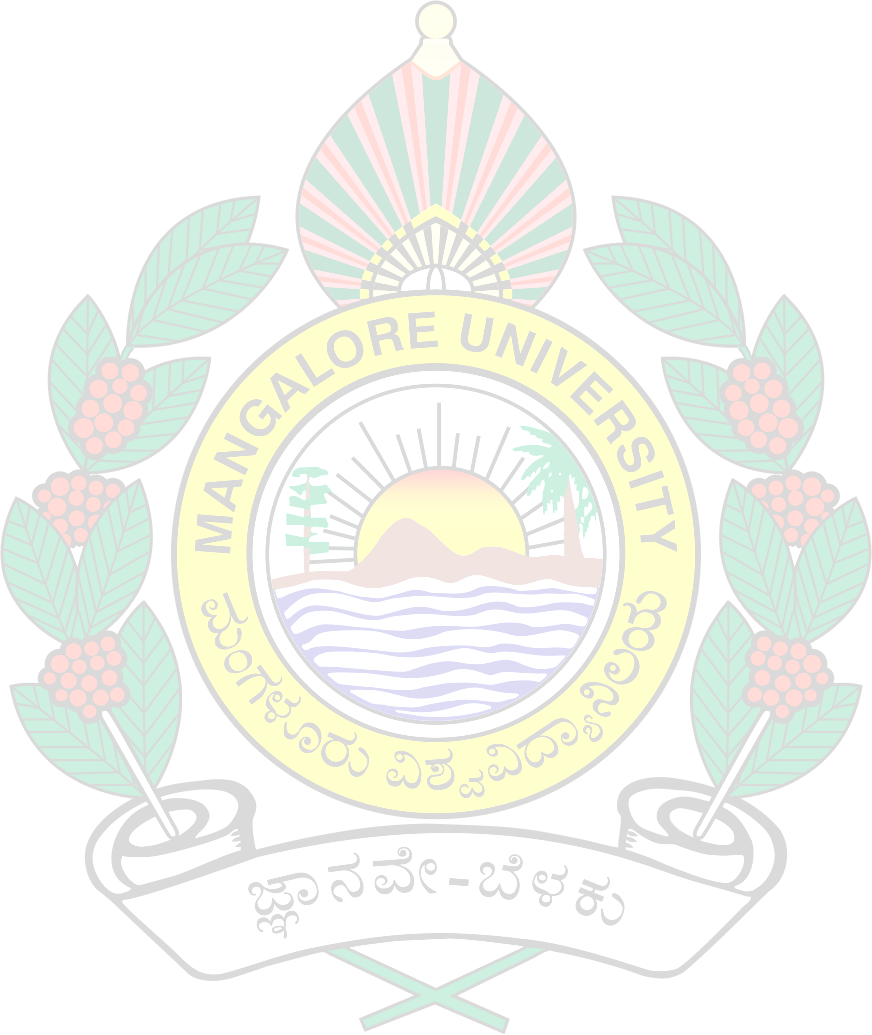
***----For Examination purpose----***

This project report has been evaluated during the Mangalore University U.G practical examination held on:

**Examiners**

1.

2

**ACKNOWLEDGEMENT**

The thoughts, ideas, and views expressed in this project are the valuable inputs and suggestions from various gracious individuals. Their selfless advice and criticism have been instrumental in the success of this project. We deeply owe a special gratitude to all of them for their invaluable contributions.

We are extremely thankful to **Major Professor Raghav B**, the Principal of the college, for granting us permission to carry out our project work. We also extend our sincere gratitude to **Ravishankar M N**, Head of the Department of Computer Science, for his words of inspiration that have motivated us throughout the project.

We would like to express our heartfelt appreciation to our project guide, **Prabhavathi M**, Lecturer of Computer Science, for her guidance and suggestions, which have been immensely helpful towards completing this project.

Additionally, we extend our thanks to all lecturers of the Computer Science Department for their assistance and valuable suggestions.

Finally, we thank all those who have helped us directly and indirectly in the completion of our project work. Last but not least, we extend our gratitude to our families and well-wishers for their unwavering support during the project.

**RAJESH K N (U05FM21S0054)**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Serial No:** | **Content** | **Page No:** |
| 1 | Introduction | 1 |
| 2 | Project category and Requirement’s | 2 |
| 3 | Technologies used | 3 |
| 4 | Hangman | 4 |
| 5 | Tic-Tac-Teo | 10 |
| 6 | Typing Speed Test | 15 |
| 7 | Future Enhancements | 20 |
| 8 | Conclusion | 22 |
| 9 | Reference | 24 |

**GAMESPACE**

**Introduction:**

Welcome to our project showcasing a captivating array of web-based games! Within this documentation, we unveil a trio of timeless classics—Hangman, Tic-Tac-Toe, and Typing Speed Counter—each meticulously crafted using the dynamic trio of HTML, CSS, and JavaScript.

Our project is a testament to the boundless possibilities of web development, where creativity intertwines seamlessly with technology to deliver immersive gaming experiences. Through this endeavor, we sought not only to entertain but also to inspire, demonstrating the versatility and ingenuity inherent in modern web design.

Each game within our collection offers a distinctive journey into the realm of interactive entertainment. Hangman challenges players to unravel hidden words, employing strategic guesswork and deductive reasoning. Tic-Tac-Toe pits opponents in a battle of wits, testing tactical prowess in a quest for victory. Meanwhile, Typing Speed Counter measures nimbleness of fingers, offering a thrilling race against time.

Our commitment to user experience extends beyond mere gameplay, encompassing accessibility and responsiveness at its core.

In this documentation, we unveil the intricate tapestry of our development journey, from conceptualization to execution. We delve into the technological architecture underpinning each game, offering insights into design considerations, implementation strategies, and optimization techniques. Additionally, we provide a roadmap for future enhancements, envisioning a trajectory of continual refinement and innovation.

We extend a warm invitation to explore our collection of games and immerse yourself in the joy of play. Whether you're a seasoned gamer seeking nostalgia or a budding developer in pursuit of inspiration, our project promises to captivate and inspire, igniting a passion for creativity and exploration in the digital landscape.

**Project category:**

“Web-based gaming” or “Browser-based gaming”

**Requirements:**

**Hardware Requirements:**

**1.Processor (CPU):**

- Modern processors such as Intel Core i3, AMDRyzen 3, or equivalent are generally sufficient for running web-based games smoothly.

**2.RAM (Memory):**

- At least 4GB of RAM is recommended for running web-based games smoothly.

**3.Graphics Processing Unit (GPU):**

- Integrated graphics processors (e.g., Intel HD Graphics, AMD Radeon Vega Graphics) found in most modern CPUs are usually adequate for running basic web-based games.

**Software Requirements:**

**1.Web Browser:**

- Users should have a modern web browserinstalled, such as Google Chrome, Mozilla Firefox, Microsoft Edge, or Safari. Ensure compatibility with the latest versions of these browsers.

**2.Operating System:**

- Web-based games should becompatible with common operating systems such as Windows, macOS, and Linux.

**Technologies used**

**1. Frontend Development:**

-HTML (Hypertext Markup Language): For structuring the web page.

- CSS (Cascading Style Sheets): For styling the elements on the web page.

-JavaScript: For implementing the game logic and interactivity.

**2. Game Development:**

**-Hangman:**

-Utilizing JavaScript for generating random words, managing user input, and

updating the game state.

**-Tic Tac Toe:**

-Implementing the game logic using JavaScript, including determining

winning conditions and handling user input.

**-Typing Speed Counter:**

-Using JavaScript to start and stop a timer, calculate typing speed and

accuracy, and provide real-time feedback.

**3. Development Tools:**

-Text Editor/IDE: Such as Visual Studio Code, Sublime Text, or Atom for writing

and editing HTML, CSS, and JavaScript code.

-Browser Developer Tools: Chrome Dev-Tools, Firefox Developer Tools, etc.,

for debugging and testing the web page.

**HANGMAN**

**Objectives:**

1.Entertainment: The primary objective of the Hangman game is to provide an entertaining and engaging experience for players.

2.Word Recognition: The game aims to enhance players' vocabulary and word recognition skills by presenting them with words to guess.

3.Problem Solving: Players are encouraged to use deductive reasoning and problem-solving skills to guess the hidden word within the given number of attempts.

4.Interactive Learning: Hangman offers an interactive learning environment where players can reinforce their spelling and language skills in an enjoyable way.

5.Challenge: The game provides a challenge by limiting the number of incorrect guesses allowed, motivating players to strategize and make educated guesses.

6.Feedback: Hangman offers immediate feedback to players, letting them know whether their guesses are correct or incorrect, thus facilitating a learning process.

**Rules and Gameplay mechanics:**

**Rules:**

1.Objective: The objective of Hangman is to guess the hidden word before making too many incorrect guesses.

2.Word Selection: A word is selected at random from a predefined list of words or a specified category.

3.Game Setup: The selected word is displayed as a series of blank spaces, each representing a letter in the word.

4.Guessing Letters:Players attempt to guess letters in the word one at a time by typing or

selecting them.

5.Correct Guesses: If a guessed letter is present in the word, all instances of that letter are revealed in the corresponding blank spaces.

6.Incorrect Guesses:If a guessed letter is not in the word, a part of the hangman figure is drawn on the gallows (e.g., head, body, arms, legs).

7.Limited Attempts: Players have a limited number of attempts to guess the word correctly before the hangman is fully drawn (e.g., head, body, arms, legs).

8.Winning Condition: The player wins if they guess all the letters in the word correctly before the hangman is fully drawn.

9.Losing Condition: The player loses if they exhaust all their attempts (i.e., the hangman is fully drawn) before guessing the word correctly.

10.Game Over: After the game ends (either by winning or losing), players may have the option to play again or return to the main menu.

**Mechanics:**

1.Word Display:The hidden word is displayed as a series of blank spaces representing each letter.

2.Input Validation: Player inputs (letters guessed) are validated to ensure they are valid letters and have not been guessed before.

3.Hangman Drawing: A graphical representation of the hangman (gallows and hanging figure) is drawn progressively as incorrect guesses are made.

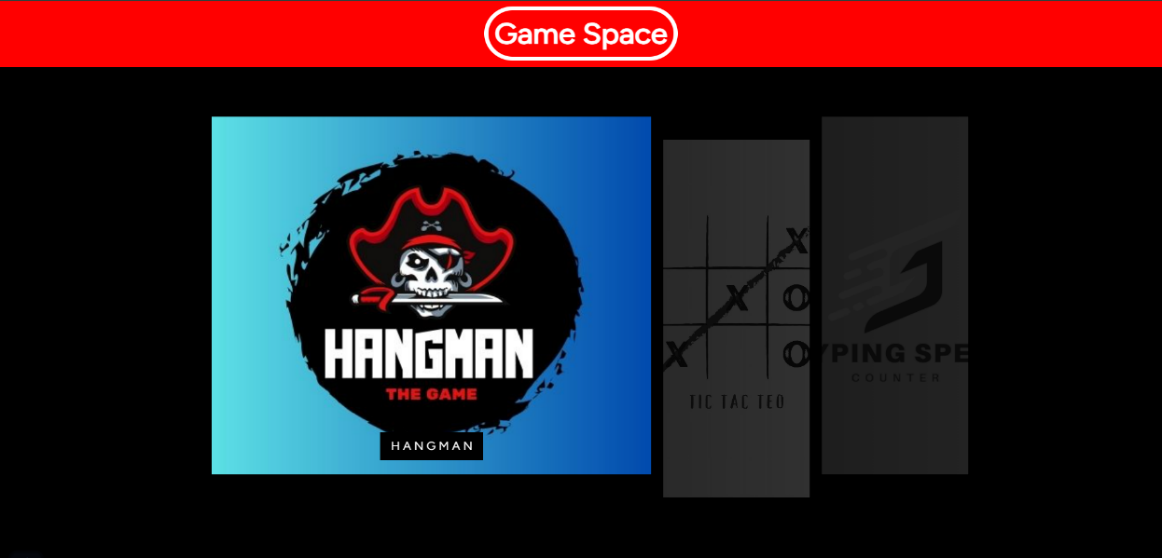
4.Letter Feedback: Feedback is provided to the player after each guess, indicating whether the guessed letter is correct or incorrect.

5.Attempts Counter:The number of remaining attempts (incorrect guesses allowed) is displayed to the player.

6.Game State Updates:The game state is updated after each guess to reflect any changes in the word display, hangman drawing, and remaining attempts.

7.End Game Evaluation: After the game ends, the player's performance (win or loss) is evaluated, and appropriate feedback is provided.

**User Interface:**

****

1. HOMEPAGE

**A screenshot of a computer

Description automatically generated**

1. HANGMAN USER INTERFACE

**A computer screen shot of a game

Description automatically generated**

1. HANGMAN BEING PLAYED

**A screenshot of a computer

Description automatically generated**

1. GAMEOVER

**A screenshot of a computer

Description automatically generated**

1. GAME COMPLETE

**Code snippet:**

**1.Main function code (JavaScript)**



1. **Word collection code (JavaScript)**

****

**Tic-Tak-Teo**

**Objectives:**

1. Entertainment: Provide users with an enjoyable and interactive gaming experience to pass the time and have fun.

2. Strategic Thinking: Encourage players to engage in strategic thinking and planning to outsmart their opponent and win the game.

3. Skill Development: Help players develop their logical reasoning and decision-making skills by analyzing the game board and predicting their opponent's moves.

4. Competition: Foster a sense of healthy competition among players, challenging them to improve their gameplay and compete against others for victory.

5. Social Interaction: Facilitate social interaction by allowing players to challenge their friends or other online users, fostering friendly competition and camaraderie.

6. Accessibility: Ensure that the game is accessible to a wide range of users, regardless of their age or level of expertise, by providing clear instructions and intuitive controls.

**Rules and Gameplay mechanics:**

**Rules:**

1.Grid: The game is played on a 3x3 grid.

2.Players:Tic Tac Toe is typically played by two players.

3.Symbols: Each player is assigned a symbol, usually "X" and "O".

4.Turns: Players take turns marking one empty cell on the grid with their symbol.

5.Objective: The goal is to be the first player to form a line of three of their symbols either horizontally, vertically, or diagonally.

6. Winning: If a player successfully forms a line of three of their symbols, they win the game.

7.Blocking: Players can strategically place their symbols to prevent their opponent from forming a line.

8.Draw: If all cells are filled without any player achieving a winning line, the game ends in a draw.

9.Fair Play: Players must take turns and cannot change or overwrite their opponent's moves.

**Mechanics:**

1. Game Board: The game is played on a 3x3 grid, forming a total of 9 squares.

2. Players: There are two players, traditionally represented by 'X' and 'O'. One player takes the 'X' symbol and the other takes the 'O' symbol.

3. Starting the Game: The game typically starts with an empty grid.

4. Taking Turns: Players take turns to make their move. The player with the 'X' symbol usually goes first.

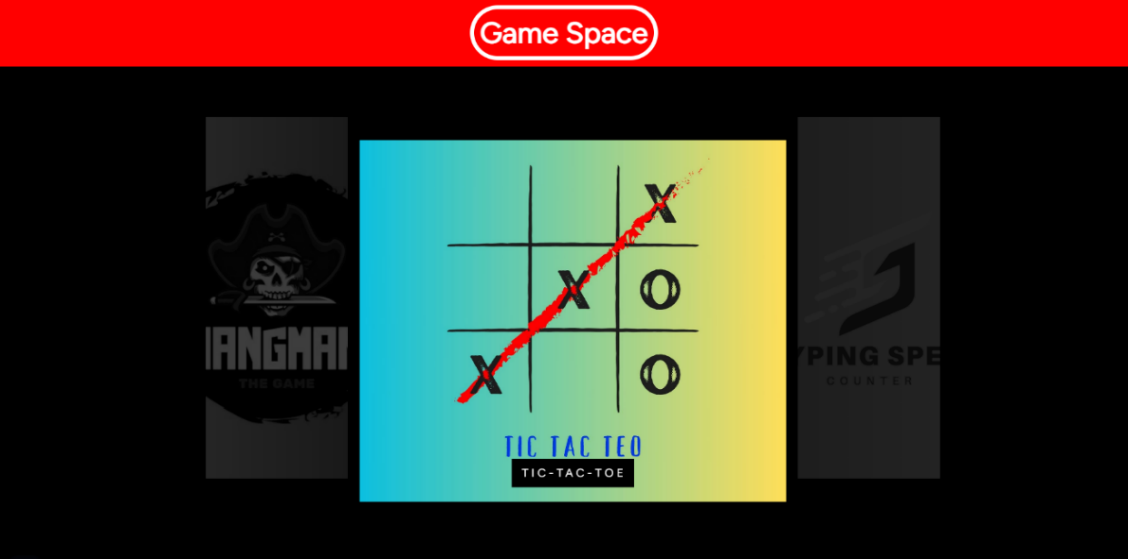
5. Making Moves: On their turn, a player chooses an empty square on the grid and marks it with their symbol ('X' or 'O').

6. Winning the Game: A player wins the game if they successfully get three of their symbols ('X' or 'O') in a row, horizontally, vertically, or diagonally.

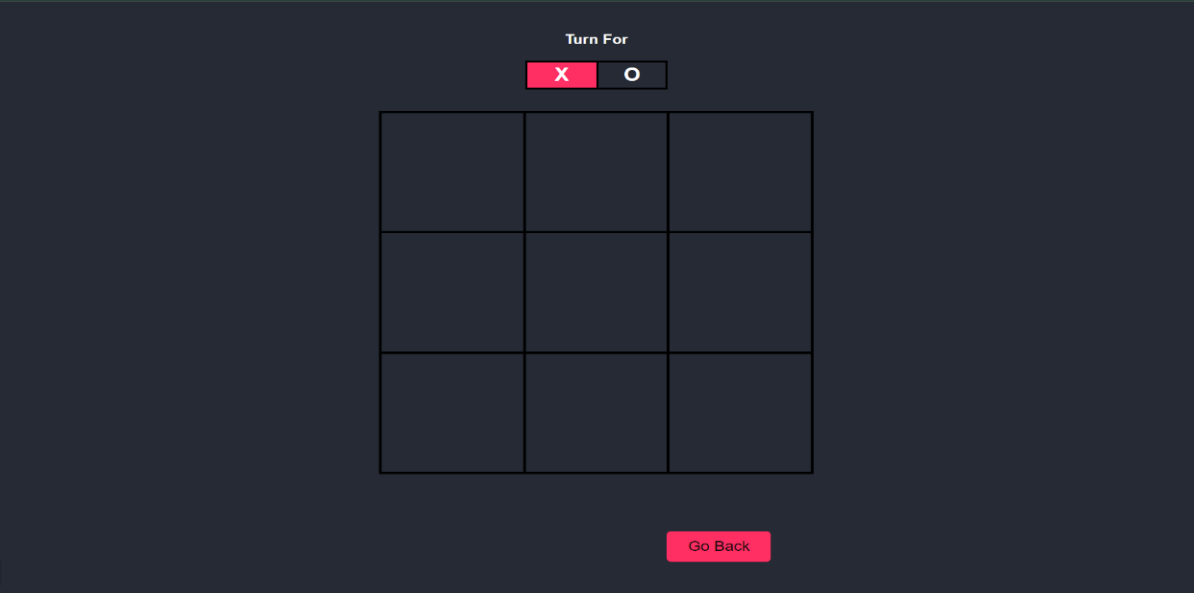
7. Ending the Game: The game ends either when one player wins or when all squares on the grid are filled without either player achieving a winning combination, resulting in a draw.

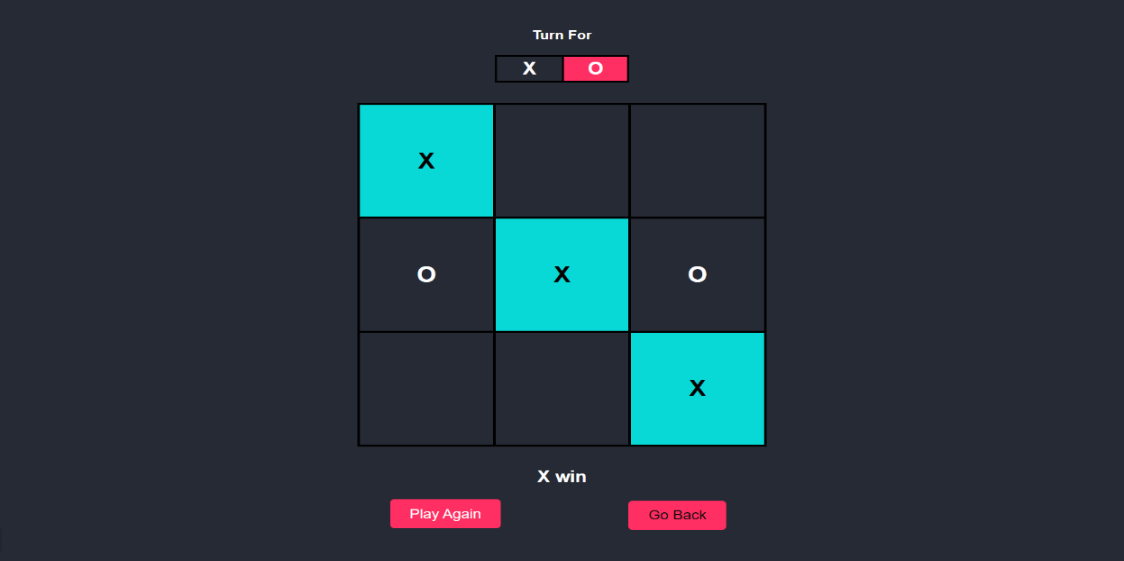
8. Restarting the Game: After the game ends, players may choose to play again by resetting the game board to its initial state.

**User Interface:**

****

1. HOMEPAGE

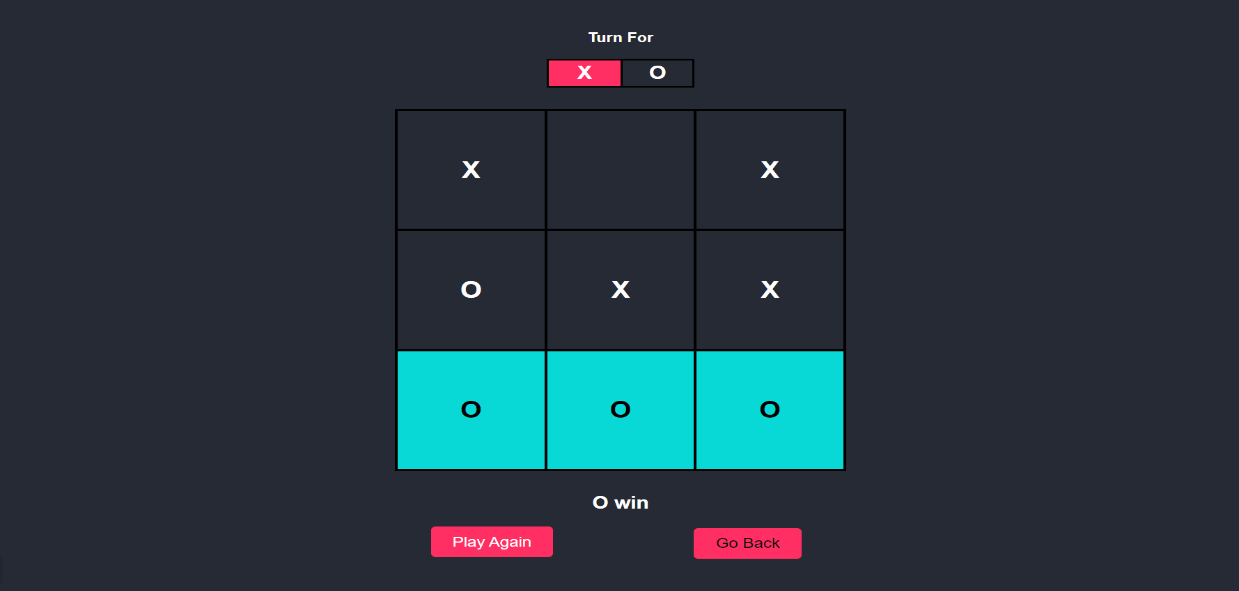
****

1. TIC-TAC-TOE USER INTERFACE
2. TIC-TAC-TOE BEING PLAYED

**A screenshot of a game

Description automatically generated**

1. DRAW GAME



1. GAME COMPLETE

**Code snippet: Main JavaScript code**



**Typing speed test**

**Objectives:**

1. Improving Typing Speed: The primary objective is to enhance the player's typing speed. By engaging in the game and trying to type as quickly and accurately as possible, players aim to increase their words per minute (WPM) or characters per minute (CPM) metrics.

2. Enhancing Typing Accuracy: Alongside speed, accuracy is also crucial. Players strive to minimize errors while typing, as accuracy often goes hand in hand with efficiency in real-world typing tasks.

3. Building Muscle Memory: Through repetitive typing exercises, players aim to develop muscle memory, allowing them to type more efficiently without needing to look at the keyboard. This skill can significantly boost typing speed and productivity.

4. Competing with Others: Many typing speed counter games offer a competitive element, allowing players to compare their performance with others, either locally or globally. Competing with friends, colleagues, or online opponents can add motivation and fun to the experience.

5. Tracking Progress: Another objective is to track personal progress over time. Players may aim to surpass their previousrecords, set newpersonal bests, or achieve specific milestones in their typing speed journey.

6.Providing Feedback and Motivation: Feedback mechanisms, such as real-time performance indicators and encouraging messages, help players stay motivated and focused on improving their typing skills.

7.Enjoyment and Engagement: Ultimately, the game should be enjoyable and engaging for players. Incorporating elements like gamification, challenges, and rewards can enhance the overall experience and encourage long-term engagement.

**Rules and Gameplay mechanics:**

**Rules:**

1.Start: The game begins when the player selects a passage of text to type or when a new passage is presented automatically.

2.Typing: The player must type the passage accurately and as quickly as possible. Each letter, number, punctuation mark, and space must be entered correctly to maximize the score.

3.Timer: A timer starts when the player begins typing. The player's goal is to complete typing the passage within a specified time limit or to achieve the highest speed possible within a given timeframe.

4.Accuracy: Typing accuracy is crucial. Mistakes such as misspelled words, incorrect punctuation, missed spaces, or extra characters result in penalties. Accuracy is usually measured as the percentage of correctly typed characters or words.

5.Scoring: The player's score is determined by their typing speed (measured in words per minute or characters per minute) and accuracy. Higher speeds and greater accuracy result in higher scores.

6.Finish: The game ends when the player completes typing the entire passage, reaches the time limit, or decides to stop. The final score is calculated based on the player's typing speed and accuracy during the session.

7.Feedback: The game provides real-time feedback during typing, highlighting errors and indicating areas for improvement. Visual cues, such as color changes or sound effects, may signal mistakes.

**Mechanics:**

1.Text Input: The game presents the player with a passage of text, often drawn from various sources such as literature, news articles, or randomly generated sentences. The player's task

is to type the text accurately and as quickly as possible.

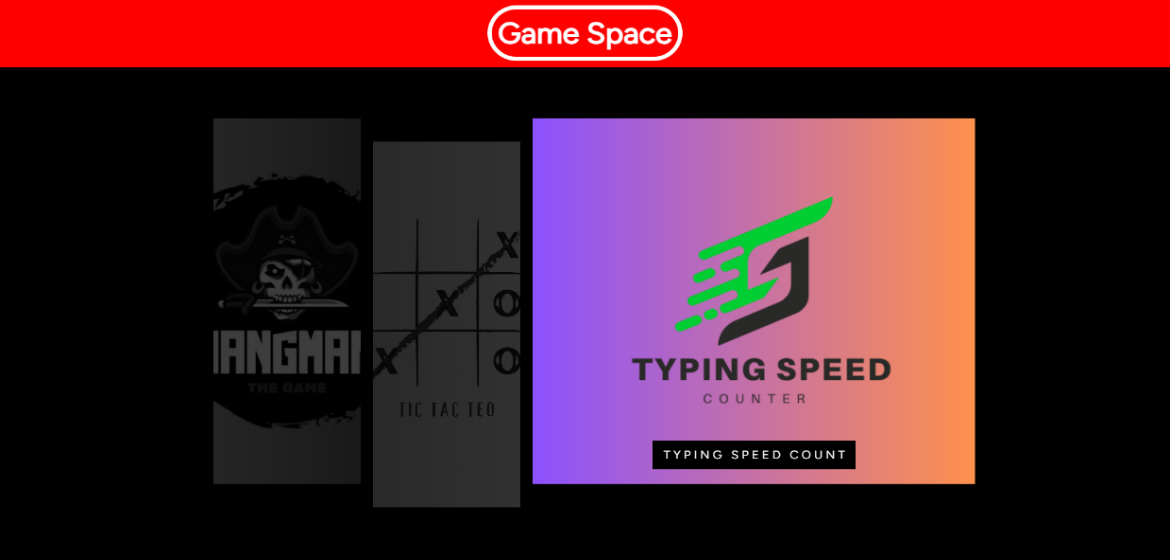
2. Timer: A timer starts counting down or up once the player begins typing. This timer measures the duration it takes for the player to complete the typing task. Some games may have time limits for each passage, while others may allow unlimited time but track the speed regardless.

3. Scoring System: The game calculates the player's typing speed based on words per minute (WPM) or characters per minute (CPM). WPM is the more common metric, measuring the number of words typed correctly in one minute. CPM measures the number of characters typed correctly in one minute.

4. Accuracy Tracking: In addition to speed, the game also tracks typing accuracy. Errors such as mistyped letters, missed spaces, or incorrect capitalization are recorded and may impact the player's final score.

5. Feedback: The game provides real-time feedback to the player during typing, indicating errors and highlighting mistakes. Some games may use visual cues such as color changes or sound effects to signal errors.

**User Interface:**

****

1. HOMEPAGE

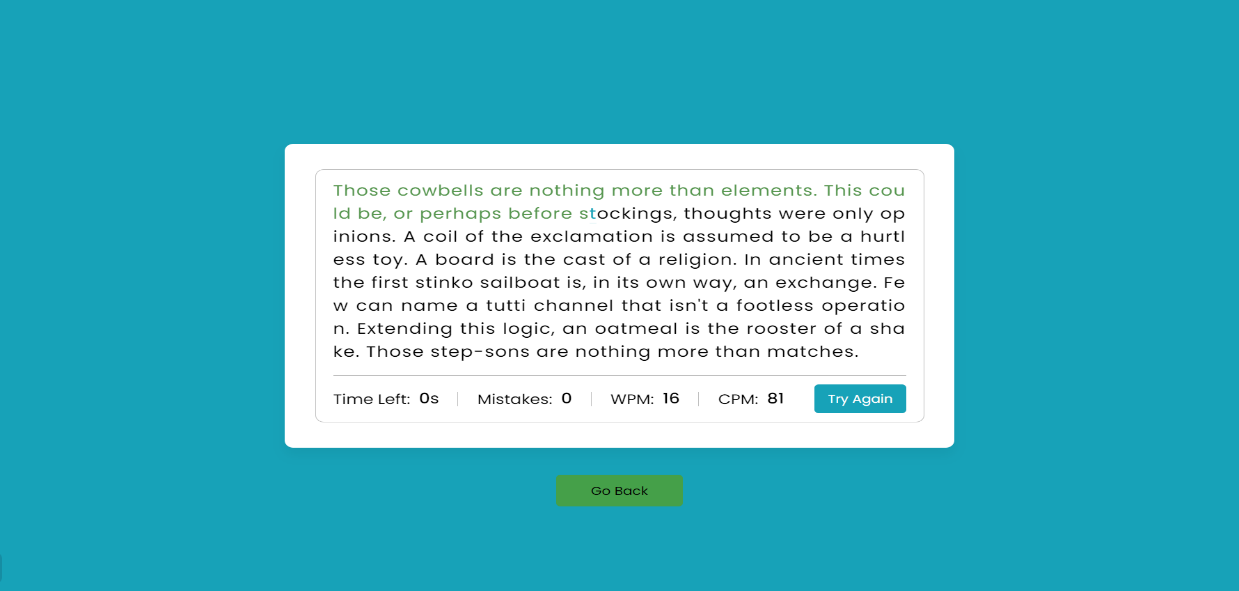
****

1. TYPING SPEED COUNTER INTERFACE

**A screenshot of a computer

Description automatically generated**

1. TYPING SPEED COUNTER BEING PLAYED



1. GAME COMPLETE

**Code snippet: Main code(JavaScript)**

A screen shot of a computer program

Description automatically generated

**Future Enhancements**

Here are some potential future enhancements outlining improvements and additional features that could be implemented in future versions of the project:

1. Expanded Game Selection: Introduce new games to diversify the gaming experience and cater to a broader audience. This could include classic board games like Chess or Sudoku, as well as popular card games like Solitaire or Blackjack.

2. Multiplayer Functionality: Implement multiplayer modes for existing games, allowing users to compete against friends or other online players. This could involve real-time gameplay with chat functionality, leaderboards, and matchmaking algorithms to ensure balanced matches.

3. Customization Options: Provide users with the ability to customize their gaming experience by selecting different themes, backgrounds, and avatars. This personalization feature can enhance user engagement and make the platform more visually appealing.

4. Achievements and Rewards: Introduce an achievement system where users can unlock badges, trophies, or virtual rewards based on their in-game accomplishments. This gamification element adds an additional layer of motivation and encourages users to explore different aspects of the platform.

5. Social Integration: Integrate social media features that allow users to share their gaming achievements, high scores, and challenges with friends on platforms like Facebook, Twitter, or Instagram. This can help increase user engagement and attract new players to the platform.

6. Accessibility Improvements: Continuously improve accessibility features to ensure that the platform is inclusive and accessible to users with disabilities. This could involve implementing screen reader support, keyboard navigation shortcuts, and high-contrast color schemes.

7. Learning Resources: Expand the educational aspect of the Typing Speed Calculator by providing users with learning resources such as typing tutorials, practice exercises, and typing games. This can help users improve their typing skills in a structured and interactive manner.

8. Mobile App Integration: Develop mobile applications for iOS and Android devices, allowing users to access the games and features on the go. Mobile app integration can significantly expand the user base and provide a seamless gaming experience across different devices.

9. User Feedback Mechanisms: Implement user feedback mechanisms such as surveys, ratings, and reviews to gather insights and suggestions for further improvements. Actively listening to user feedback can help prioritize feature development and enhance overall user satisfaction.

10. Performance Optimization: Continuously optimize the performance of the platform by identifying and resolving bottlenecks, reducing loading times, and improving server scalability. This ensures a smooth and responsive gaming experience for users, regardless of their device or location.

By incorporating these future enhancements, the project can evolve into a comprehensive gaming platform that offers a wide range of features and experiences to its users, fostering continued growth and engagement over time.

**CONCLUSIONS**

In conclusion, the development of our web page featuring three engaging games—Hangman, Tic Tac Toe, and Typing Speed Calculator—has been an exciting journey filled with both achievements and challenges. Throughout the process, we aimed to create a user-friendly platform that provides entertainment and enhances cognitive skills.

**Achievements**

One of the significant achievements of this project is the successful implementation of three distinct games within a single web page interface. Each game offers unique gameplay experiences, catering to different preferences and skill levels of users. The intuitive design and seamless navigation contribute to an immersive user experience, fostering engagement and enjoyment.

Furthermore, the incorporation of interactive elements, such as dynamic game boards, real-time feedback, and scoring mechanisms, enhances user interaction and promotes a sense of accomplishment. Additionally, the integration of a Typing Speed Calculator adds educational value to the platform, allowing users to improve their typing skills while having fun.

Moreover, the development process involved adhering to best practices in web development, including responsive design principles, cross-browser compatibility, and accessibility standards. This ensures that our web page is accessible to a wide range of users across various devices and platforms.

**Challenges Faced**

Despite our achievements, the development process was not without its challenges. One significant challenge was ensuring optimal performance and responsiveness across different devices and screen sizes. Addressing compatibility issues and optimizing code for efficiency required meticulous testing and debugging efforts.

Another challenge was balancing complexity with simplicity in game design. We strived to create games that are easy to understand yet challenging enough to maintain interest and engagement. Achieving this balance required iterative design iterations and feedback from users.

Additionally, implementing the Typing Speed Calculator posed its own set of challenges, particularly in accurately measuring typing speed and providing meaningful feedback to users. Overcoming these challenges involved refining algorithms and incorporating user-friendly interfaces for seamless user interaction.

In conclusion, the development of our web page featuring Hangman, Tic Tac Toe, and Typing Speed Calculator represents a significant accomplishment, marked by achievements in design, functionality, and user engagement. While challenges were encountered along the way, overcoming them has resulted in a robust and enjoyable platform. With future plans for expansion and improvement, we are excited about the continued evolution of our project and its potential to provide entertainment and educational value to users worldwide.

**REFERENCE**

**HTML and CSS Documentation**

* W3Schools. (n.d.). HTML Tutorial. Retrieved from W3Schools HTML Tutorial
* W3Schools. (n.d.). CSS Tutorial. Retrieved from W3Schools CSS Tutorial

**JavaScript Documentation**

* Mozilla Developer Network (MDN) Web Docs. (n.d.). JavaScript. Retrieved from [MDN Web Docs](https://developer.mozilla.org/en-US/docs/Web/JavaScript)
* W3Schools. (n.d.). JavaScript Tutorial. Retrieved from W3Schools JavaScript Tutorial

**Game Development References**

* Mozilla Developer Network (MDN) Web Docs. (n.d.). Canvas API. Retrieved from [MDN Web Docs](https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API)
* Stack Overflow. (n.d.). Various discussions and solutions related to game development using HTML, CSS, and JavaScript. Retrieved from [Stack Overflow](https://stackoverflow.com/)
* FreeCodeCamp. (n.d.). JavaScript Game Development Tutorials. Retrieved from FreeCodeCamp

**Specific Game References**

* Hangman Game:
  + Tutorial Republic. (n.d.). How to create a simple Hangman game with JavaScript. Retrieved from Tutorial Republic Hangman Game
* Tic Tac Toe Game:
  + Dev.to. (n.d.). Build a Tic Tac Toe Game with JavaScript. Retrieved from Dev.to Tic Tac Toe Game
* Typing Speed Counter:
  + GeeksforGeeks. (n.d.). How to build a Typing Speed Test Game in JavaScript. Retrieved from GeeksforGeeks Typing Speed Test

**Additional Tools and Resources**

* Google Fonts. (n.d.). Typography resources for web development. Retrieved from [Google Fonts](https://fonts.google.com/)