

# **Visvesvaraya Technological University**

**Jnana Sangama, Belagavi – 590018, Karnataka**



## **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**A Report on**

### **HANGMAN GAME**

**In partial fulfillment of WEB TECHNOLOGY LABORATORY [17CSL77]  
in Computer Science and Engineering for the Academic Year 2020-2021**

**SANDEEP VY (1GA17CS134)**

**Under the Guidance of  
Medha kudri  
Assistance professor**



## **GLOBAL ACADEMY OF TECHNOLOGY**

**Department of Computer Science and Engineering**

**(Accredited by NBA 2019-2022)**

**Rajarajeshwari Nagar, Bengaluru – 560 098**





# GLOBAL ACADEMY OF TECHNOLOGY

Department of Computer Science and Engineering

(Accredited by NBA 2019-2022)

Rajarajeshwari Nagar, Bengaluru – 560 098



## Certificate

This is to certify that the project entitled **“HANGMAN”** is a bonafide work carried out by **SANDEEP VY(1GA17CS134)** as a partial fulfillment for the award of Bachelors Degree in Computer Science and Engineering for Web Technology Laboratory as prescribed by **Visvesvaraya Technological University, Belagavi** during the year 2020-2021.

-----  
Mrs. Medha kudari  
Assistant Professor,  
Dept of CSE,  
GAT, Bangaluru.

-----  
Dr Srikanta Murthy K  
Professor & Head,  
Dept of CSE,  
GAT, Bangaluru.

### External Viva

Name of the Examiner

Signature with date

1. \_\_\_\_\_

\_\_\_\_\_

2. \_\_\_\_\_

\_\_\_\_\_

## **ABSTRACT**

The Hangman is a web project developed using HTML5, CSS, and JavaScript. This game is about guessing letters (A-Z) to form the words. You have to find the word by selecting a letter each time. If the player guesses the right letter that is within the word, the letter appears at its correct positions.

You win if you find the word in certain amount of chances, if he/she fails to complete the word then the game is over.

## ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible and whose constant encouragement and guidance crowned our efforts with success.

I consider our self proud, to be part of **Global Academy of Technology** family, the institution which stood by the way in endeavors.

I express our deep and sincere thanks to our Principal **Dr. N. Rana Pratap Reddy** for his support.

I am grateful to **Dr. Srikanta Murthy K**, Professor and HOD, Dept of CSE who is source of inspiration and of invaluable help in channelizing my efforts in right direction.

I wish to thank our internal guide **Mrs. Medha Kudari**, Assistant Professor, Dept. of CSE for guiding and correcting various documents of mine with attention and care. They have taken lot of pain to go through the document and make necessary corrections as and when needed.

I would like to thank the faculty members and supporting staff of the Department of CSE, GAT for providing all the support for completing the Project work.

Finally, I am grateful to our parents and friends for their unconditional support and help during the course of our Project work.

**SANDEEP VY [1GA17CS134]**

# TABLE OF CONTENT

|           |                                  |           |
|-----------|----------------------------------|-----------|
| <b>1.</b> | <b>INTRODUCTION</b>              | <b>1</b>  |
| <b>2.</b> | <b>REQUIREMENT SPECIFICATION</b> | <b>2</b>  |
| 2.1       | SOFTWARE REQUIREMENTS            | 2         |
| 2.2       | HARDWARE REQUIREMENTS            | 2         |
| 2.3       | FUNCTIONAL REQUIREMENTS          | 2         |
| 2.4       | NON FUNCTIONAL REQUIREMENTS      | 3         |
| <b>3.</b> | <b>OBJECTIVE OF THE PROJECT</b>  | <b>4</b>  |
| <b>4.</b> | <b>IMPLEMENTATION</b>            | <b>5</b>  |
| 4.1       | SOURCE CODE                      | 5         |
| <b>5.</b> | <b>TESTING</b>                   | <b>21</b> |
| 5.1       | TESTING PROCESS                  | 21        |
| 5.2       | TESTING OBJECTIVES               | 22        |
| <b>6.</b> | <b>RESULTS</b>                   | <b>23</b> |
| 6.1       | SNAPSHOTS                        | 23        |
|           | <b>CONCLUSION</b>                |           |
|           | <b>REFERENCES</b>                |           |

## LIST OF FIGURES

| Figure No. | Title             | Page No. |
|------------|-------------------|----------|
| 6.1        | Home Page         | 23       |
| 6.2        | Guessing Page     | 24       |
| 6.3        | Lose dialogue box | 24       |
| 6.4        | Won dialogue box  | 25       |

## CHAPTER 1

### INTRODUCTION

The Hangman is a simple project developed using HTML5, CSS, and JavaScript. This game is about guessing letters (A-Z) to form the words. You have to find the word by selecting a letter each time. If the player guesses the right letter that is within the word, the letter appears at its correct positions. You win if you find the word in certain amount of chances, if he/she fails to complete the word then the game is over. The Hangman project is simply in HTML, CSS, and JavaScript. Taking about the features of this game, the user has to guess the correct letters to form the correct word within certain amount of chances. This game includes a lot of JavaScript for making the functioning of the game.

## CHAPTER 2

### REQUIREMENT SPECIFICATION

A high-level requirements specification is required. The purpose of the requirements analysis is to identify requirements for the proposed system.

#### 2.1 SOFTWARE REQUIREMENTS

Operating System : Windows 7 or any compatible operating system.

Code-Editor : Brackets

Browser : Chrome

#### 2.2 HARDWARE REQUIREMENTS

Processor : Any Processor above 500 MHz

RAM : 4GB

Hard Disk : 2 GB free space

Input device : Keyboard, Mouse

Output device : Monitor

System type : 32-bit or 64-bit operating system

#### 2.3 FUNCTIONAL REQUIREMENTS

**Home page:** Home page is the only page of the website. It contains hangman game, player has to guess the right word to win.

#### 2.4 NONFUNCTIONAL REQUIREMENTS:

##### PERFORMANCE:

Performance requirements define acceptable response times for system functionality.

- The load time for user interface screens shall take no longer than five seconds.
- Dialog-box shall return results within five seconds.



**RELIABILITY:**

- Appropriate responses will be popped.

**FLEXIBILITY:**

- The game runs on any browser.

**TIMELINESS:**

- The system carries out all the operations with consumptions of very less time.

## CHAPTER 3

### OBJECTIVE OF THE PROJECT

The Hangman is a simple project developed using HTML5, CSS, and JavaScript. This game is about guessing letters (A-Z) to form the words. You have to find the word by selecting a letter each time.

If the player guesses the right letter that is within the word, the letter appears at its correct positions. You win if you find the word in certain amount of chances, if he/she fails to complete the word then the game is over.

---

## CHAPTER 4

### IMPLEMENTATION

#### 4.1 SOURCE CODE

##### HTML Source Code

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <link rel="preconnect" href="https://fonts.gstatic.com"
/>
    <link
href="https://fonts.googleapis.com/css2?family=Roboto+Slab:w
ght@300;400;500;600;700&family=Rubik:wght@300;400;500;600;70
0&display=swap"
    rel="stylesheet"
  />
    <link rel="stylesheet" href="assets/styles/style.css" />
    <title>Hangman Game</title>
  </head>
  <body>
    <div class="container">
      <h1 style="color:red;">Hangman Game</h1>
      <p>
        Try to find the hidden word by typing a letter each
time of your choice.
      </p>
      <div class="game-box">
        <svg class="figure">
          <!-- Stand -->
          <line x1="25%" y1="5%" x2="65%" y2="5%" />
          <line x1="65%" y1="5%" x2="65%" y2="20%" />
          <line x1="25%" y1="5%" x2="25%" y2="95%" />
          <line x1="5%" y1="95%" x2="45%" y2="95%" />

          <!-- Head -->
          <circle r="10%" cx="65%" cy="30%" class="body-
part" />
```

```
<!-- Body -->
<line x1="65%" y1="40%" x2="65%" y2="60%"
class="body-part" />

<!-- Arms -->
<line x1="50%" y1="40%" x2="65%" y2="50%"
class="body-part" />
<line x1="80%" y1="40%" x2="65%" y2="50%"
class="body-part" />

<!-- Legs -->
<line x1="65%" y1="60%" x2="80%" y2="70%"
class="body-part" />
<line x1="65%" y1="60%" x2="50%" y2="70%"
class="body-part" />
</svg>
<div class="content">
  <ul id="word" class="word"></ul>
  <div id="incorrect" class="incorrect">
    <h2>Incorrect :</h2>
    <p></p>
  </div>
</div>
<div id="backdrop" class="backdrop"></div>
<div id="no-work" class="no-work">
  <p>This app doesn't work in touch screen
devices.</p>
</div>
<div id="final-msg" class="final-msg">
  <p id="msg-info" class="msg-info"></p>
  <button id="play" class="play">Play Again</button>
</div>
<div id="indication" class="indication">
  <p>You have already entered this letter</p>
</div>
</div>
<script src="assets/scripts/script.js"></script>
</body>
</html>
</div>
<div class="clear"></div>
</div>
```

```
</div>
</div>
</div>

    </div>
    <div class="clear"></div>
<div class="ftr-bg">
<div class="wrap">
<div class="footer">
    <div class="f_nav">
        <ul>
            <li class="active"><a
href="index.php">Home</a></li>
            <li><a href="donar.php">Donor</a></li>
            <li><a href="login.php">log In</a></li>
            <li><a href="aboutus.php">About</a></li>
            <li><a href="contact.php">Contact Us</a></li>

        </ul>
    </div>
    <div class="copy">
        <p class="title">© All Rights Reserved </p>
    </div>
    <div class="clear"></div>
</div>
</div>
</div>
</body>
</html>
```

### CSS Source Code

```
* {
    padding: 0;
    margin: 0;
    box-sizing: border-box;
}

html {
    font-size: 62.5%;
}

body {
    font-family: 'Rubik', sans-serif;
    color: #fff;
```

```
    background-color: #233d4d;
}

.container {
    max-width: 70rem;
    width: 100%;
    padding: 2rem;
    margin: auto;
}

.container h1 {
    font-family: 'Roboto Slab', sans-serif;
    font-size: 3.4rem;
    text-align: center;
    margin: 2rem 0;
    color: #31cf1d;
}

.container p {
    font-size: 1.8rem;
    text-align: center;
    margin: 2rem 0;
}

.game-box {
    padding: 1rem;
    margin: 6rem 0 2rem;
    display: flex;
    justify-content: space-evenly;
}

.figure {
    display: block;
    height: 24rem;
    width: 20rem;
    fill: transparent;
    stroke: #fff;
    stroke-width: 4px;
    stroke-linecap: round;
    flex-shrink: 0;
}

.body-part {
    display: none;
}
```

```
}

.content {
  height: 24rem;
  max-width: 50rem;
  width: 100%;
  flex-shrink: 1;
  display: flex;
  flex-direction: column;
  overflow: hidden;
}

.word {
  font-size: 3rem;
  height: 70%;
  list-style-type: none;
  display: flex;
  justify-content: center;
  align-items: center;
}

.letter {
  text-align: center;
  display: block;
  height: 4rem;
  width: 3rem;
  margin-right: 0.8rem;
  border-bottom: 4px #46a9fc solid;
}

.incorrect {
  height: 30%;
  padding: 1rem;
  display: flex;
  display: none;
}

.incorrect.visible {
  display: flex;
}

.incorrect h2 {
  display: inline-block;
  padding: 0 0 0 4rem;
}
```

```
    font-size: 2rem;
    font-weight: 500;
}

.incorrect p {
    font-size: 2rem;
    display: inline-block;
    margin: 0;
    padding: 0 1rem;
}

.backdrop {
    position: fixed;
    top: 0;
    left: 0;
    height: 100%;
    width: 100%;
    background-color: rgba(0, 0, 0, 0.75);
    z-index: 10;
    display: none;
}

.backdrop.visible {
    display: block;
}

.no-work {
    position: fixed;
    top: 0;
    left: 0;
    bottom: 0;
    right: 0;
    z-index: 100;
    max-width: 40rem;
    width: 75%;
    height: 15rem;
    padding: 2rem;
    line-height: 1.6;
    margin: auto;
    background-color: #c532e2;
    box-shadow: 0 0 8px rgba(0, 0, 0, 0.75);
    border-radius: 8px;
    display: flex;
    justify-content: center;
```



---

```
    align-items: center;
    display: none;
}

.no-work p {
    font-size: 2rem;
}

.final-msg {
    position: fixed;
    top: 0;
    left: 0;
    bottom: 0;
    right: 0;
    z-index: 100;
    max-width: 40rem;
    width: 75%;
    height: 20rem;
    padding: 2rem;
    line-height: 1.6;
    margin: auto;
    background-color: #233d4d;
    box-shadow: 0 0 8px rgba(0, 0, 0, 0.75);
    border-radius: 8px;
    display: flex;
    flex-direction: column;
    justify-content: center;
    align-items: center;
    display: none;
}

.final-msg.visible {
    display: flex;
}

.final-msg p {
    font-size: 2rem;
    margin: 0;
}

.final-msg .play {
    font: inherit;
    font-size: 1.6rem;
    padding: 1.2rem;
```

```
    cursor: pointer;
    margin-top: 3rem;
    background-color: #008080;
    border-radius: 4px;
    color: #fff;
    border: none;
    outline: none;
    transition: opacity 200ms ease-in;
}

@media (hover: hover) {
    .final-msg .play:hover {
        opacity: 0.75;
    }
}

.indication {
    border: 2px #fff solid;
    border-radius: 8px;
    background-color: #e34754;
    position: fixed;
    left: 0;
    right: 0;
    bottom: 0;
    max-width: 40rem;
    width: 75%;
    margin: auto;
    transform: translateY(100%);
    transition: transform 200ms ease-in;
}

.indication.visible {
    transform: translateY(-20%);
}

.indication p {
    margin: 2rem;
    font-size: 1.8rem;
}

@media (max-width: 600px) {
    .game-box {
        flex-direction: column;
        align-items: center;
    }
}
```

```
        justify-content: center;
    }

    .content {
        height: 20rem;
    }
}

@media (hover: none) {
    .backdrop {
        display: block;
    }

    .no-work {
        display: flex;
    }
}
```

---

**JavaScript Source Code**

```
const word = document.getElementById('word');

const incorrect = document.getElementById('incorrect');

const incorrectLettersEl = document.querySelector('#incorrect
p');

const backdrop = document.getElementById('backdrop');

const finalMsg = document.getElementById('final-msg');

const msgInfo = document.getElementById('msg-info');

const playBtn = document.getElementById('play');

const indication = document.getElementById('indication');

const bodyParts = document.getElementsByClassName('body-
part');

// List of words

const wordList = [

    'sandeep',

    'santosh',

    'global',

    'academy',

    'technology',

    'covid',

    'corona',

];
```

```
// Word that is selected to play

let selectedWord = null;

// Stores the count of no.of incorrectly typed letters

let incorrectCount = 0;

// Correct letters typed by the player

const correctLetters = [];

// Incorrect letters typed by the player

const incorrectLetters = [];


// Select a word randomly from wordList and initialize in the
DOM

function initializeWord() {

    selectedWord = wordList[Math.floor(Math.random() *
wordList.length)];

    const noOfLetters = selectedWord.length;

    for (let i = 0; i < noOfLetters; i++) {

        const listItem = document.createElement('li');

        listItem.classList.add('letter');

        word.append(listItem);

    }

}


// Displays an indication sliding from the bottom

function displayIndication() {
```

```
    indication.classList.add('visible');

    setTimeout(() => {

        indication.classList.remove('visible');

    }, 2400);

}

// Update the figure when incorrect letters typed

function updateFigure() {

    try {

        bodyParts[incorrectCount].style.display = 'block';

        incorrectCount++;

    } catch (error) {}

}

// When player wins

function successState() {

    setTimeout(() => {

        backdrop.classList.add('visible');

        finalMsg.classList.add('visible');

        msgInfo.textContent = 'Hurrah! You won.';

    }, 400);

}
```

```
// When player loses

function failureState() {

    setTimeout(() => {

        backdrop.classList.add('visible');

        finalMsg.classList.add('visible');

        msgInfo.textContent = `Oops! You lost. The right word is
"${selectedWord}"`;

        }, 400);

}

// Check if typed key is part of the selected word and update
in the DOM if required

function check(ev) {

    const letterElements = document.querySelectorAll('.word
.letter');

    const character = ev.key;

    // Handle keyboard events

    if (

        !backdrop.classList.contains('visible') &&

        !indication.classList.contains('visible') &&

        ev.keyCode >= 65 &&

        ev.keyCode <= 90
```

```
) {  
  
  if (selectedWord.includes(character)) {  
  
    if (correctLetters.includes(character)) {  
  
      displayIndication();  
  
    } else {  
  
      correctLetters.push(character);  
  
      const indexes = [];  
  
      [...selectedWord].forEach((value, index) => {  
  
        if (value === character) {  
  
          indexes.push(index);  
  
        }  
  
      });  
  
      indexes.forEach((value) => {  
  
        letterElements[value].textContent = character;  
  
      });  
  
    }  
  
  } else {  
  
    if (incorrectLetters.includes(character)) {  
  
      displayIndication();  
  
    } else {  
  
      incorrectLetters.push(character);  
  
      if (!incorrect.classList.contains('visible')) {  
  
        incorrect.classList.add('visible');
```



```
    }

    incorrectLettersEl.textContent =
`${incorrectLetters.join(', ')}`;

    updateFigure();

  }

}

}

// Create a word from all letter items

let formedWord = '';

letterElements.forEach((value) => {

  formedWord += value.textContent;

});

// Check if created word is correct

if (formedWord === selectedWord) {

  successState();

}

// Check if man was hung

if (incorrectCount >= 6) {

  failureState();

}
```

```
}

// Reset all variables and start a new game

function startNewGame() {

    selectedWord = null;

    incorrectCount = 0;

    correctLetters.splice(0);

    incorrectLetters.splice(0);

    word.innerHTML = '';

    Array.from(bodyParts).forEach((value) => {

        value.style.display = 'none';

    });

    incorrect.classList.remove('visible');

    backdrop.classList.remove('visible');

    finalMsg.classList.remove('visible');

    initializeWord();

}

// Start the game

initializeWord();

// Event Listeners

window.addEventListener('keyup', check);

playBtn.addEventListener('click', startNewGame);
```

## CHAPTER 5

### TESTING

This chapter gives the outline of the testing methods that are carried out to get a bug free system. Quality can be achieved by testing the product using different techniques at different phases of the project development. The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components sub assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

#### 5.1 TESTING PROCESS

Testing is an integral part of software development. Testing process certifies whether the product that is developed compiles with the standards that it was designed to. Testing process involves building of test cases against which the product has to be tested.

#### 5.2 TESTING OBJECTIVES

The main objectives of testing process are as follows.

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has high probability of finding undiscovered error.
- A successful test is one that uncovers the undiscovered error.

**Table 5.3: Test cases**

| <b>S.NO</b> | <b>CASE</b> | <b>INPUT</b> | <b>EXPECTED<br/>OUTPUT</b> | <b>ACTUAL OUTPUT</b> |
|-------------|-------------|--------------|----------------------------|----------------------|
| 1           | HTML        | Blank Field  | Html game Content          | Successful           |
| 2           | CSS         | Blank Field  | Game Design/Layout         | Successful           |
| 3           | JAVASCRIPT  | Blank Field  | Game Responses             | Successful           |

## CHAPTER 6

### RESULTS

This section describes the screens of the “HangMan”.

The snapshots are shown below for each module.

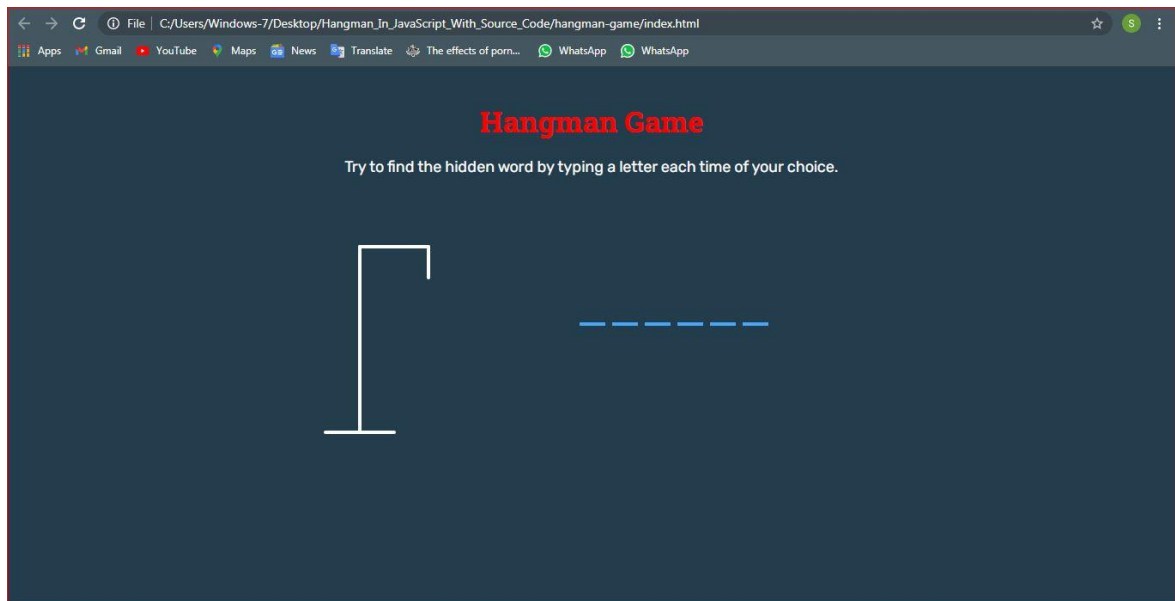


Figure 6.1: Home page

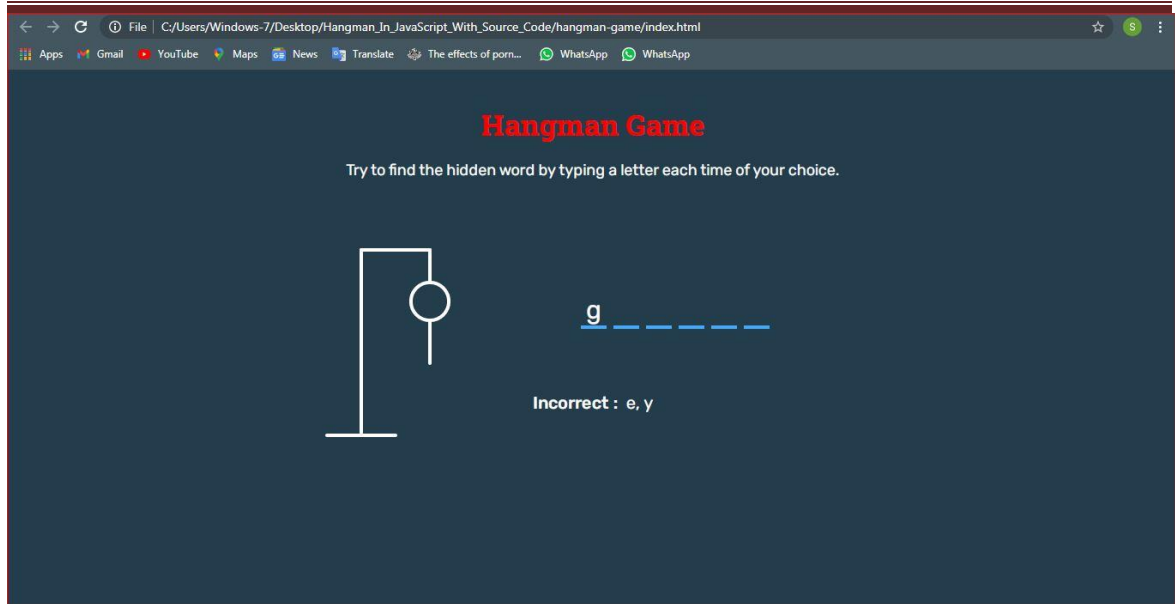


Figure 6.2: Guessing page

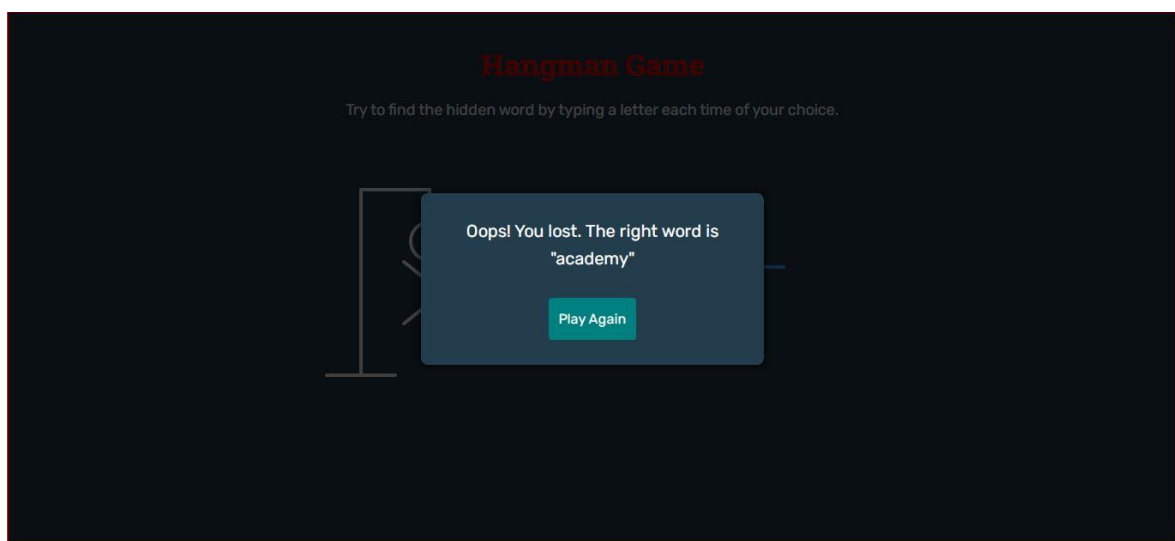
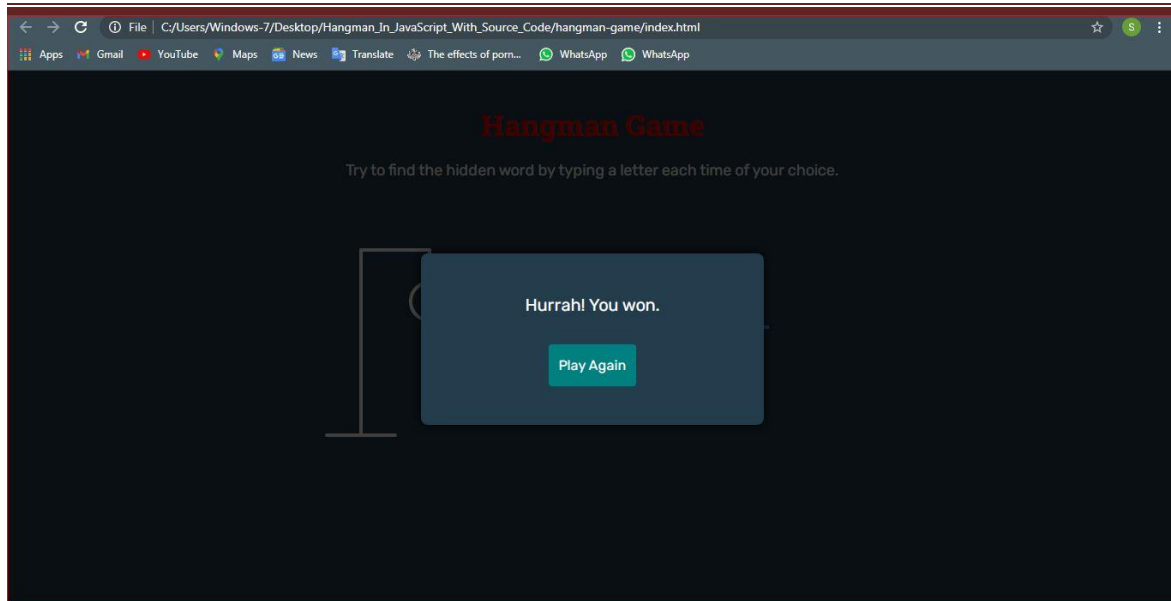


Figure 6.3: Lose Dialogue Box



**Figure 6.4: Won Dialogue Box**

## CONCLUSION

With the theoretical inclination of our syllabus it becomes very essential to take the at most advantage of any opportunity of gaining practical experience that comes along. The building blocks of this Major Project "HANGMAN GAME" was one of these opportunities. It gave us the requisite practical knowledge to supplement the already taught theoretical concepts thus making us more competent as a computer engineer. The project from a personal point of view also helped us in understanding the following aspects of project development:

- The planning that goes into implementing a project.
- The importance of proper planning and an organized methodology.
- The key element of team spirit and co-ordination in a successful project.



## BIBLIOGRAPHY

- [1]. <https://www.w3schools.com>
- [2]. Randy Connolly, Ricardo Hoar, **“Fundamentals of Web Development”**, 1<sup>st</sup> Edition, Pearson Education India.
- [3]. Robin Nixon, **“Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5”**, 4<sup>th</sup> Edition, O’Reilly Publications, 2015.