

A FIELD PROJECT REPORT ON

BRAIN VAULT

A Fun and Interactive Puzzle game

Submitted

in partial fulfilment of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

by

A.BALAJI 231EA04694

N. BINDU MADHAVI 231EA04B11

CH. KOMALI PRIYA 231EA04B18

T. CHARITHA 231EA04B72

Under the Guidance of

Mr. K. Pavan Kumar

Assistant Professor, CSE



Department of COMPUTER SCIENCE AND ENGINEERING

School of ENGINEERING AND TECHNOLOGY

VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY AND RESEARCH (DEEMED TO BE UNIVERSITY)
Vignan's Foundation for Science, Technology and Research (Deemed to be University) Vadlamudi,
Guntur, Andhra Pradesh-522213, India

APRIL - 2025



(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

CERTIFICATE

DATE :- 26-04-2025

This is to certify that the field project entitled "BRAIN VAULT" being submitted by A.BALAJI(231FA04694),N.BINDU(231FA04B11),CH.KOMALIPRIYA(231FA04B18),and T.CHARITHA(231FA04B72) in partial fulfilment of Bachelor of Technology in the Department of CSE, Vignan's Foundation For Science Technology & Research (Deemed to be University), Vadlamudi, Guntur District, Andhra Pradesh, India, is a Bonafide work carried out by them under my guidance and supervision.

The work embodied in this project is original and has not been submitted for the award of any other degree, diploma, or certificate earlier. The project work is carried out in accordance with the academic regulations and guidelines set by the university.

A handwritten signature in blue ink, appearing to read "F.P. Anil".

Guide

A handwritten signature in blue ink, appearing to read "S. R. K." with a horizontal line extending from the end of the "K".

A handwritten signature in blue ink, appearing to read "O. R." with a horizontal line extending from the end of the "R".

Project Review Committee

HOD, CSE

HoD
Dept. of Computer Science & Engineering
VFSTR Deemed to be University
VADLAMUDI - 522 211
Guntur Dist., A.P., INDIA

DECLARATION

We hereby declare that the work presented in the field project titled "**BRAIN VAULT**" is the result of our own efforts and investigations.

This project is being submitted under the supervision of **Mr. K. Pavan Kumar, Assistant professor, CSE** in partial fulfilment of the requirements for the Bachelor of Technology (B.Tech.) degree in Computer Science and Engineering at Vignan's Foundation for Science, Technology and Research (Deemed to be University), Vadlamudi, Guntur, Andhra Pradesh, India.

A.BALAJI	231FA04694	Signature: <u>A.Balaji</u>
N.BINDU MADHAVI	231FA04B11	Signature: <u>N.Bindu Madhavi</u>
CH.KOMALIPRIYA	231FA04B18	Signature: <u>Ch.Komalipriya.</u>
T.CHARITHA	231FA04B72	Signature: <u>T.Charitha.</u>

Contents

S.NO	DESCRIPTION	PAGE NO
1	Abstract Introduction Problem Definition Overview Proposed System	5-6
2	System Requirements Hardware & Software Software specifications	7
3	System Design Modules Of System UML diagrams	8-10
4	Implementations Sample Code Test cases Output Screens	11-23
5	Results	24
6	Conclusion	25
7	References	26

1. ABSTRACT:

Brain Vault is an interactive web-based puzzle game designed to challenge and enhance users' critical thinking and problem-solving skills. The platform offers a collection of 50 carefully curated puzzles, where players can select how many puzzles they want to attempt based on their preference. With an engaging interface, captivating background visuals, and a user-friendly design, Brain Vault provides an immersive environment for users to test their mental agility. The project aims to combine learning and fun, encouraging users to sharpen their minds in a dynamic and intuitive setting. Built using HTML, CSS, and JavaScript, Brain Vault ensures accessibility across devices, promoting logical thinking through gamified experiences.

INTRODUCTION:

Project Overview: An interactive quiz-based web application simulating an escape room where users must answer questions to progress through levels. Encourages learning and engagement using gamified elements.

Need for the Project: Aimed at increasing engagement through learning by providing an interactive and fun environment that tests knowledge while entertaining the user.

OBJECTIVES:

- Create a user-friendly escape room game.
- Provide real-time feedback for answers.
- Display final scores.
- Ensure scalability and future backend support.

PROBLEM DEFINITION:

To design and develop an interactive, web-based escape room quiz game that overcomes the limitations of traditional quiz systems by offering real-time feedback, immersive gameplay, and a dynamic scoring mechanism that enhances user ENGAGEMENT and provides a scalable foundation for future development.

The Escape Room Game aims to address this challenge by creating a web-based interactive quiz platform that simulates the experience of an escape room. In this game, users are required to answer a series of questions correctly to "escape" or complete the challenge. The platform incorporates real-time feedback, a progressive question flow, and a score tracking system to keep users engaged throughout the experience.

OVERVIEW OF THE EXISTING SYSTEM:

Static Quiz Applications:

- Typically involve a fixed set of multiple-choice questions.
- Presented all at once or with minimal progression control.
- Often lack visual appeal and interactivity.

Offline Escape Room Games:

- Physical spaces designed with puzzles and clues.
- Require physical presence and setup.
- Not accessible to a wide audience.

Basic Web-Based Quiz Platforms:

- Some allow linear quiz progression but without escape-room logic or game-like storytelling.
- Scoring systems may be present but often without real-time feedback or progression-based unlocking of levels.

PROPOSED SYSTEM

- Web-based access
- Interactive gameplay
- Custom question count
- Real-time feedback
- Score tracking system
- User-friendly interface
- Future backend integration

2. SYSTEM REQUIREMENTS:

Basic computing device with internet access and a modern web browser.

HARDWARE & SOFTWARE REQUIREMENTS

Hardware Requirements:

- Processor: Intel i3 or higher
- RAM: 4GB minimum
- Disk Space: 100MB

Software Requirements:

- Browser: Chrome/Firefox
- IDE: VS Code (for development)
- OS: Windows/Linux/Mac

SOFTWARE REQUIREMENTS SPECIFICATIONS

Functional Requirements:

- Start game
- Display questions
- Validate answers
- Score tracking

Non-Functional Requirements:

- User-friendly interface
- Fast response time
- Platform independent

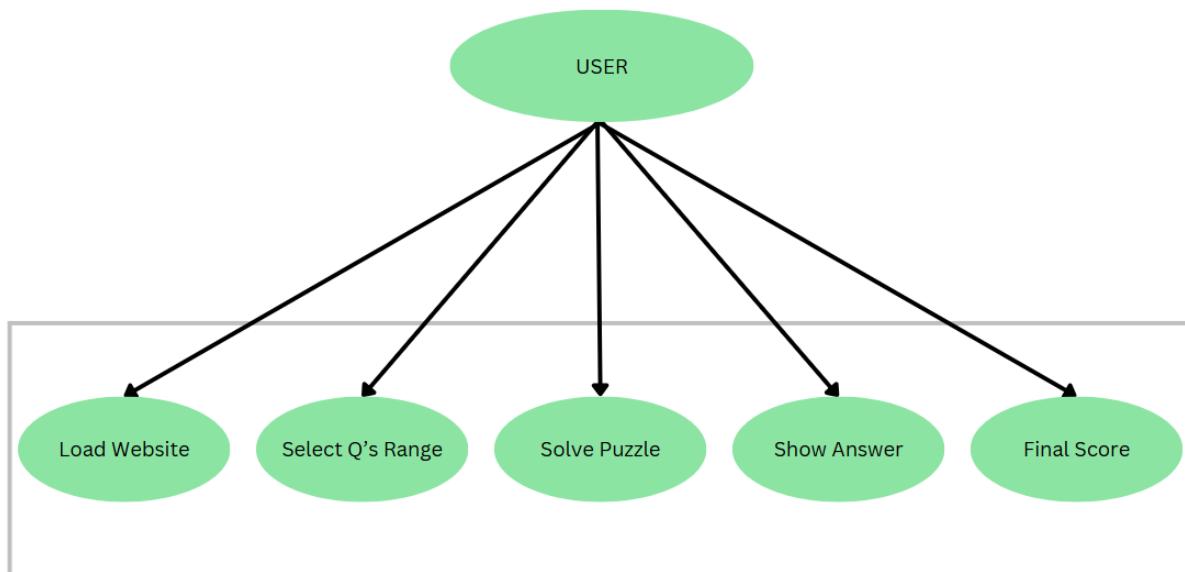
3. SYSTEM DESIGN

The system uses HTML for structure, CSS for styling, and JavaScript for interactivity. The architecture is simple and modular, enabling easy navigation and flow control.

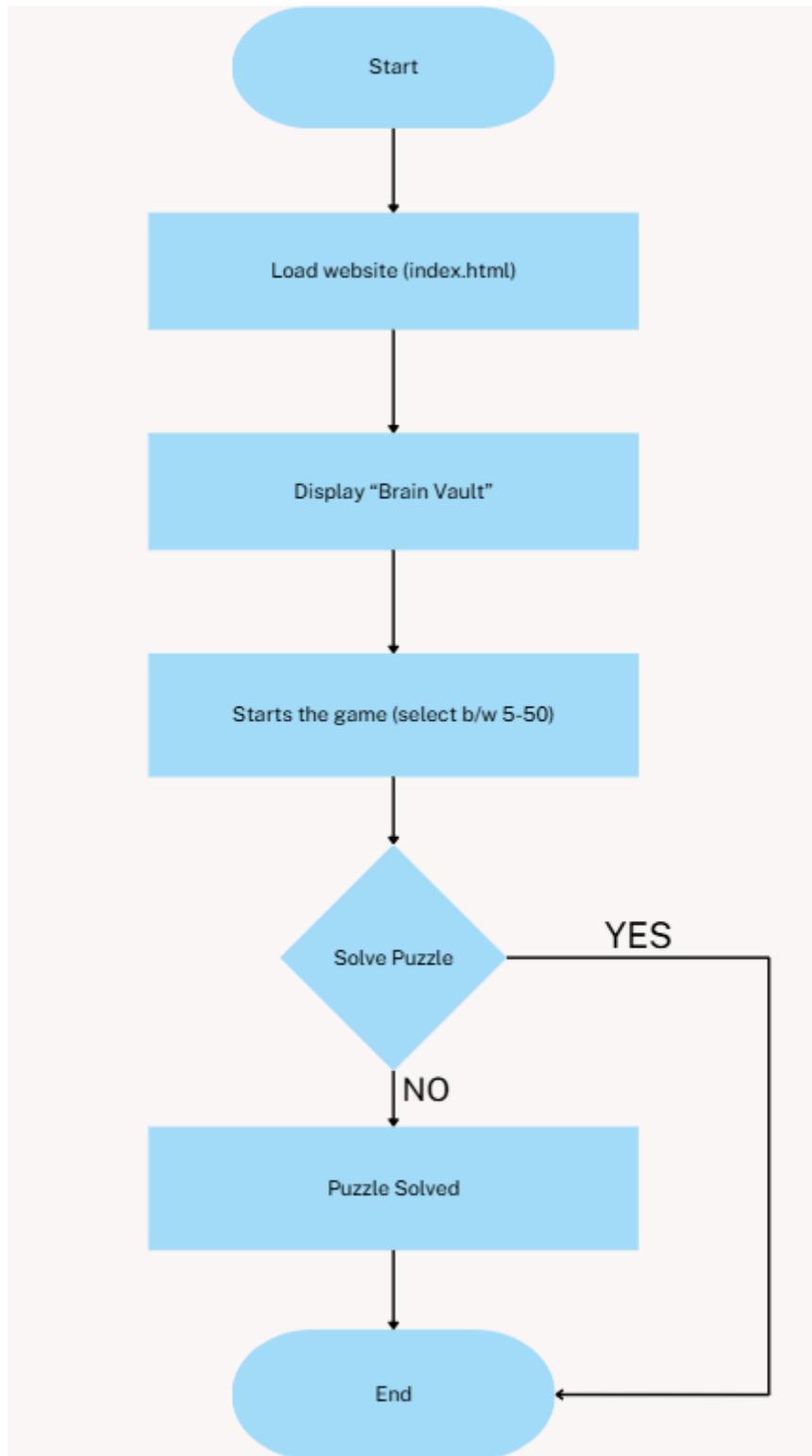
MODULES OF SYSTEM

- Welcome Module
- Question Display Module
- Answer Validation Module
- Scoring Module
- Result Module

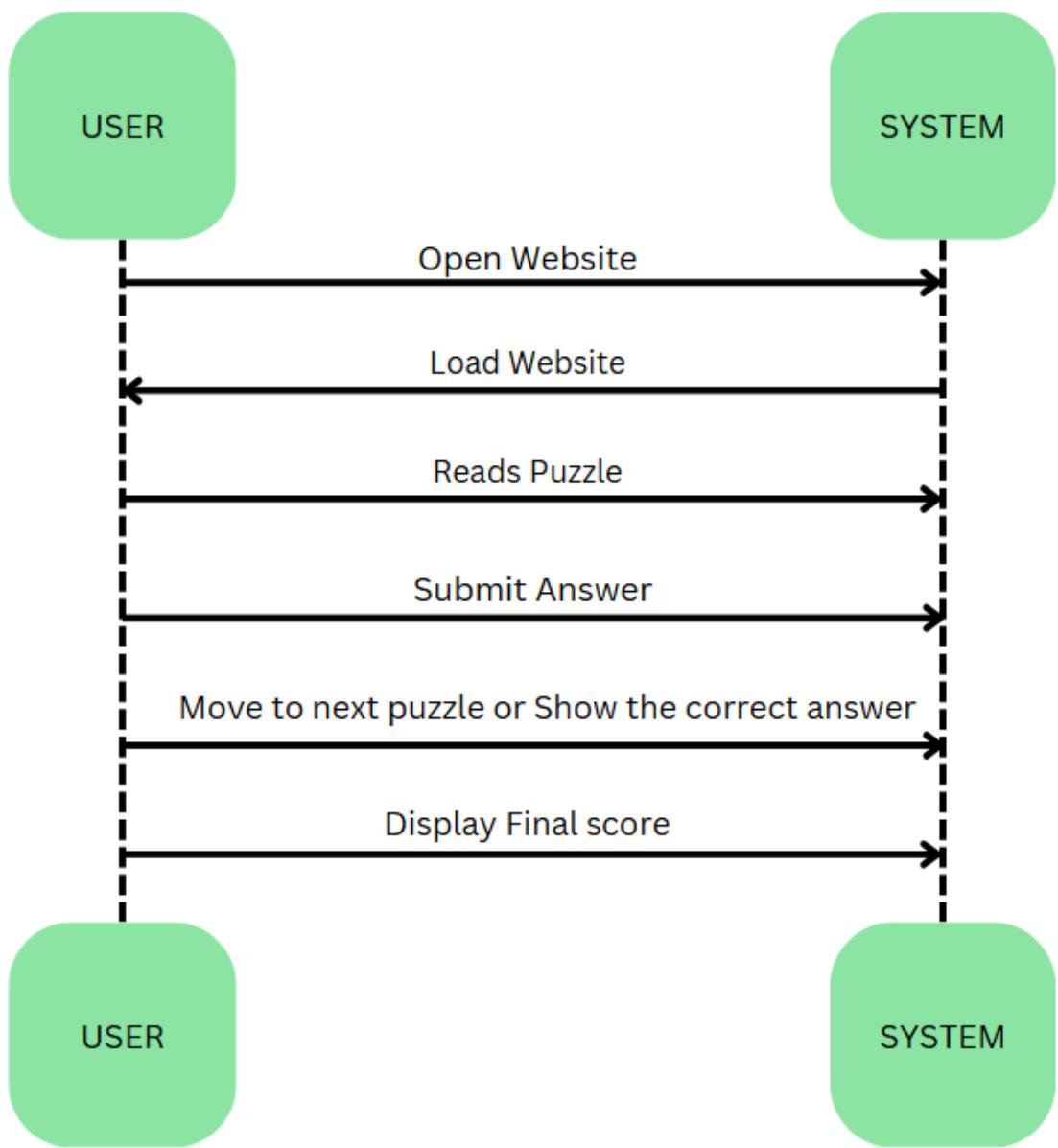
UML DIAGRAMS:



Use case diagram for Brain Vault. The Use Case Diagram highlights the main interactions between the User and the System in the Brain Vault game. It clearly defines that the user is actively involved in every part of the gameplay experience.



Activity Diagram for Brain Vault. This flow ensures that the user can smoothly navigate through the game without getting stuck at any point.



Sequence Diagram for Brain Vault. This sequence ensures that there is a smooth two-way communication between the user and the system throughout the entire game.

4. IMPLEMENTATIONS:

Frontend is developed using HTML, CSS, and JavaScript. Game logic is handled in JS while UI components are styled using CSS.

SAMPLE CODE

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0"/>

<title>Brain Vault</title>

<style>

* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: 'Arial', sans-serif;

}

body {

background: url('background.jpeg') no-repeat centercenter fixed;

background-size: cover;

background-color: #000;

display: flex;

align-items: center;

justify-content: center;

height: 100vh;

text-align: center;

}
```

```
.container {  
background-color: #E0FFFF;  
padding: 2rem;  
border-radius: 12px;  
width: 90%;  
max-width: 600px;  
box-shadow: 0px 0px 20px rgba(186, 85, 211, 0.6);  
}  
  
h1 {  
font-size: 2rem;  
margin-bottom: 1.5rem;  
color: #000;  
text-shadow: 0px 0px 10px rgba(186, 85, 211, 1), 0px 0px 20px rgba(186, 85, 211, 0.8), 0px 0px 40px  
rgba(186, 85, 211, 0.6);  
}  
  
input[type="number"], input[type="text"] {  
width: 90%;  
padding: 0.8rem;  
margin: 1.5rem 0;  
border-radius: 8px;  
border: none;  
background: #000;  
color: white;  
text-align: center;  
font-size: 1rem;  
box-shadow: 0px 0px 10px rgba(186, 85, 211, 0.5);
```

```
}
```

```
button {  
    padding: 0.8rem 1.5rem;  
    background: #000;  
    color: white;  
    border: none;  
    border-radius: 8px;  
    cursor: pointer;  
    font-size: 1rem;  
    transition: 0.3s;  
    box-shadow: 0px 0px 10px rgba(186, 85, 211, 0.6);  
}  
  
}
```

```
button:hover {  
    background: #111;  
    transform: scale(1.05);  
    box-shadow: 0px 0px 15px rgba(186, 85, 211, 1);  
}  
  
}
```

```
.feedback {  
    margin-top: 1rem;  
    font-weight: bold;  
    font-size: 1.2rem;  
    color: #9370DB;  
}  
  
}
```

```
h2, p {
```

```

color: #000;

}

</style>

</head>

<body>

<div class="container">

<h1>Brain Vault</h1>

<div id="startPage">

<p>We have 50 fun and tricky puzzles! Choose how many you want to solve:</p>

<input type="number" id="numQuestions" min="5" max="50" step="5" placeholder="Choose between 5 and 50">

<button onclick="startGame()">Start Game</button>

</div>

<div id="gamePage" style="display:none;">

<h2 id="questionArea"></h2>

<input type="text" id="answerInput" placeholder="Your answer here">

<button id="submitButton" onclick="submitAnswer()">Submit</button>

<button id="nextButton" onclick="nextQuestion()" style="display:none;">Next</button>

<div id="feedback" class="feedback"></div>

</div>

<div id="endPage" style="display:none;">

<h2>Game Over!</h2>

<p id="finalScore"></p>

<button onclick="restartGame()">Play Again</button>

</div>

```

</div>

<script>

```
const questions = [
```

```
    { q: "What has to be broken before you can use it?", a: ["egg"] },  
    { q: "I'm tall when I'm young, and I'm short when I'm old. What am I?", a: ["candle"] },  
    { q: "What month of the year has 28 days?", a: ["all", "every month"] },  
    { q: "What is full of holes but still holds water?", a: ["sponge"] },  
    { q: "What gets wet while drying?", a: ["towel"] },  
    { q: "What has a neck but no head?", a: ["bottle"] },  
    { q: "What comes down but never goes up?", a: ["rain"] },  
    { q: "I speak without a mouth and hear without ears. What am I?", a: ["echo"] },  
    { q: "What has hands but can't clap?", a: ["clock"] },  
    { q: "What has many keys but can't open a single lock?", a: ["piano"] },  
    { q: "The more you take, the more you leave behind. What am I?", a: ["footsteps"] },  
    { q: "What is always in front of you but can't be seen?", a: ["future"] },  
    { q: "There's a one-story house where everything is yellow. What color are the stairs?", a: ["no stairs"] },  
    { q: "What can you break, even if you never pick it up or touch it?", a: ["promise"] },  
    { q: "What goes up but never comes down?", a: ["age"] },  
    { q: "A man who was outside in the rain without an umbrella didn't get a single hair on his head wet. Why?", a: ["bald"] },  
    { q: "What can you keep after giving to someone?", a: ["word"] },  
    { q: "What has a head, a tail, but no body?", a: ["coin"] },  
    { q: "What can travel around the world while staying in a corner?", a: ["stamp"] },  
    { q: "What has legs but doesn't walk?", a: ["table"] },  
    { q: "What has an eye but can't see?", a: ["needle"] },  
    { q: "What kind of room has no doors or windows?", a: ["mushroom"] },  
    { q: "What is black when it's clean and white when it's dirty?", a: ["chalkboard"] },  
    { q: "What has words but never speaks?", a: ["book"] },
```

```

{ q: "What begins with T, ends with T, and has T in it?", a: ["teapot"] },
{ q: "What has a face and two hands but no arms or legs?", a: ["clock"] },
{ q: "What has four wheels and flies?", a: ["garbage truck"] },
{ q: "What has a bottom at the top?", a: ["leg"] },
{ q: "What can run but never walks, has a bed but never sleeps?", a: ["river"] },
{ q: "What has roots as nobody sees, is taller than trees?", a: ["mountain"] },
{ q: "What comes once in a minute, twice in a moment, but never in a thousand years?", a: ["m"] },
{ q: "What is always coming but never arrives?", a: ["tomorrow"] },
{ q: "What flies without wings?", a: ["time"] },
{ q: "What can be cracked, made, told, and played?", a: ["joke"] },
{ q: "What has 13 hearts but no other organs?", a: ["deck of cards"] },
{ q: "What has teeth but can't bite?", a: ["comb"] },
{ q: "What is easy to lift but hard to throw?", a: ["feather"] },
{ q: "What can't be put in a saucepan?", a: ["lid"] },
{ q: "What kind of band never plays music?", a: ["rubber band"] },
{ q: "What has cities but no houses?", a: ["map"] },
{ q: "What has oceans but no water?", a: ["map"] },
{ q: "What has mountains but no rocks?", a: ["map"] },
{ q: "What gets bigger the more you take away?", a: ["hole"] },
{ q: "What goes through towns and over hills but never moves?", a: ["road"] },
{ q: "What can you hold without touching it?", a: ["breath"] },
{ q: "What comes at the end of everything?", a: ["g"] }

];

```

```
let selectedQuestions = [];
```

```
let currentQuestionIndex = 0;
```

```
let totalQuestions = 0;
```

```
let correctAnswers = 0;
```

```

function startGame() {

constnum = parseInt(document.getElementById('numQuestions').value);

if (num>= 5 && num<= 50 && num % 5 === 0) {

selectedQuestions = questions.sort(() => 0.5 - Math.random()).slice(0, num);

totalQuestions = num;

currentQuestionIndex = 0;

correctAnswers = 0;

document.getElementById('startPage').style.display = 'none';

document.getElementById('gamePage').style.display = 'block';

showQuestion();

} else {

alert("Please choose a number between 5 and 50 in increments of 5.");

}

}

function showQuestion() {

document.getElementById('answerInput').value = "";

document.getElementById('questionArea').textContent = selectedQuestions[currentQuestionIndex].q;

document.getElementById('feedback').textContent = "";

document.getElementById('submitButton').style.display = 'inline';

document.getElementById('nextButton').style.display = 'none';

}

```

```

function submitAnswer() {

const userAnswer = document.getElementById('answerInput').value.trim().toLowerCase();

const correctAnswersList = selectedQuestions[currentQuestionIndex].a;

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

if (correctAnswersList.includes(userAnswer)) {

  feedback.innerHTML = " ✅ Correct!";

  correctAnswers++;

} else {

  feedback.innerHTML = ` ❌ Incorrect! The correct answer was: <b>${correctAnswersList[0]}</b>`;

}

document.getElementById('submitButton').style.display = 'none';

document.getElementById('nextButton').style.display = 'inline';

}

function nextQuestion() {

currentQuestionIndex++;

if (currentQuestionIndex<totalQuestions) {

showQuestion();

} else {

  document.getElementById('gamePage').style.display = 'none';

  document.getElementById('endPage').style.display = 'block';

  document.getElementById('finalScore').textContent = `You answered ${correctAnswers} out of ${totalQuestions} correctly!`;

}

}

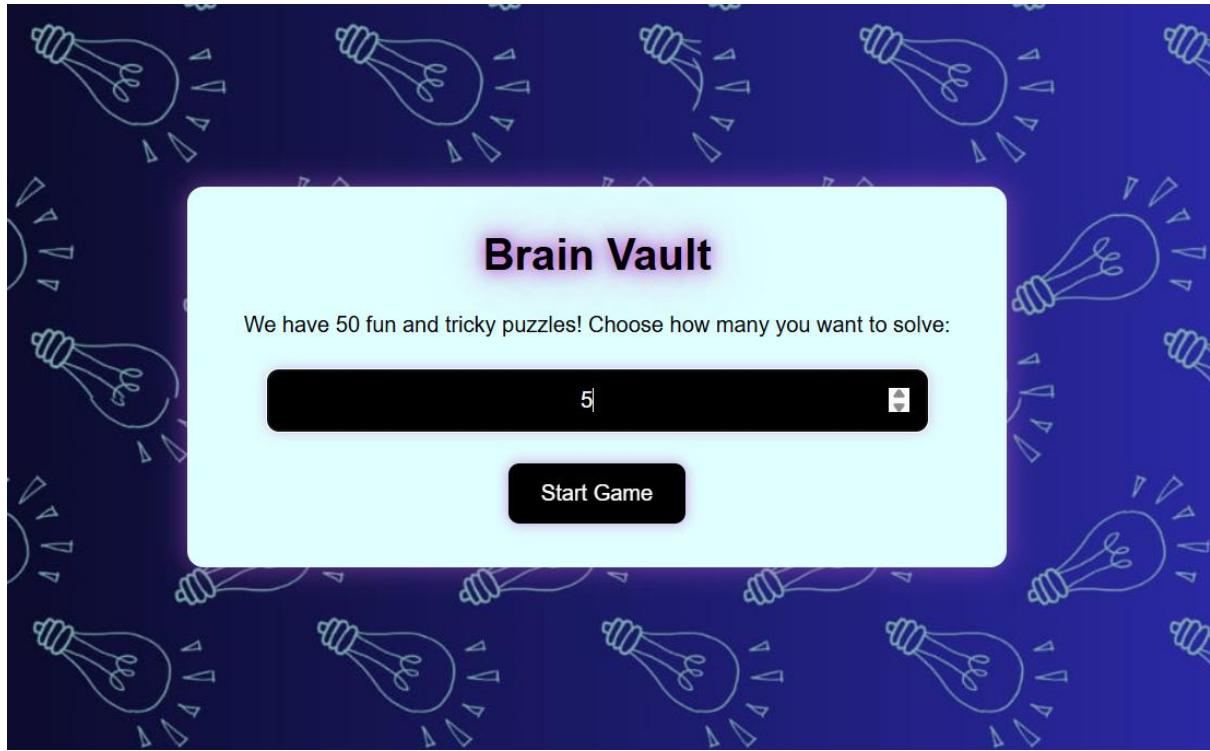
```

```
function restartGame() {  
  
document.getElementById('endPage').style.display = 'none';  
  
document.getElementById('startPage').style.display = 'block';  
  
}  
  
</script>  
  
  
</body>  
  
</html>
```

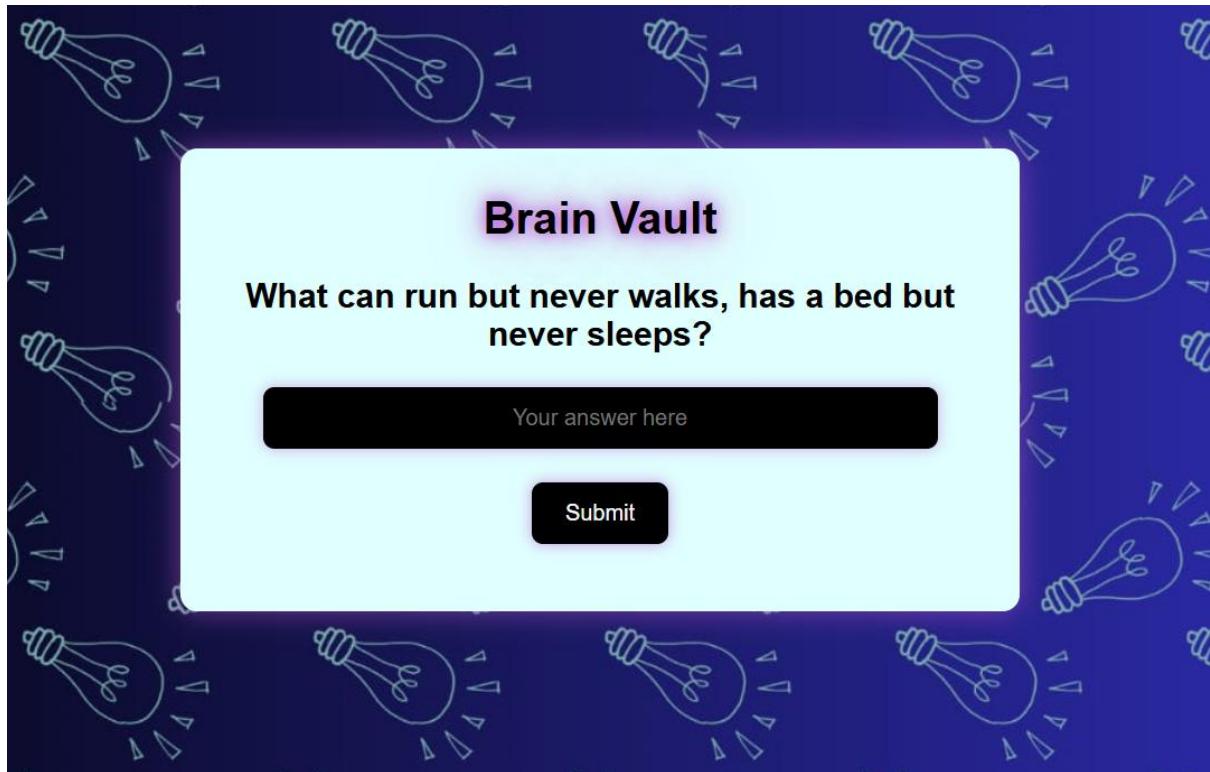
TEST CASES:

Test Case ID	Test Scenario	Test Steps	Expected Result	Actual Result	Status
TC01	Load Webpage	Open the index.html file in browser	Home page loads showing "Brain Vault" title	Home page displayed	<input checked="" type="checkbox"/> Pass
TC02	Start the Game	Click on the "Start" button	First puzzle appears	Puzzle displayed	<input checked="" type="checkbox"/> Pass
TC03	Submit Correct Answer	Enter correct answer and click "Submit"	"Correct!" message shown and next puzzle loads	Message displayed	<input checked="" type="checkbox"/> Pass
TC04	Submit Wrong Answer	Enter wrong answer and click "Submit"	"Wrong! Correct answer is..." message shown	Message displayed	<input checked="" type="checkbox"/> Pass
TC05	Empty Answer Submission	Click "Submit" without entering anything	Show validation message or prevent submission	Validation shown	<input checked="" type="checkbox"/> Pass
TC06	Answer Case Sensitivity Check	Enter answer in different cases (e.g., lowercase, uppercase)	System accepts correct answer irrespective of case	Case handled properly	<input checked="" type="checkbox"/> Pass
TC07	Score Update After Correct Answer	Submit correct answers	Score increments by 1	Score incremented	<input checked="" type="checkbox"/> Pass
TC08	Score Update After Wrong Answer	Submit wrong answers	Score remains the same	Score unchanged	<input checked="" type="checkbox"/> Pass
TC09	Completion of All Puzzles	Answer all available puzzles	Final score is displayed	Score displayed	<input checked="" type="checkbox"/> Pass
TC10	Browser Refresh During Quiz	Refresh page in the middle of quiz	Quiz restarts from beginning	Quiz restarted	<input checked="" type="checkbox"/> Pass
TC11	Responsive Layout Test (Mobile View)	Open project on a mobile browser	Content adjusts properly (responsive design)	Mobile friendly	<input checked="" type="checkbox"/> Pass
TC12	Image Loading	Check if any image (logo or background) is displayed	Images load without broken links	Images displayed	<input checked="" type="checkbox"/> Pass
TC13	Script Loading Error	Disable JavaScript and load page	Puzzle functionality does not work (as expected)	JS dependency confirmed	<input checked="" type="checkbox"/> Pass
TC14	Submit Answer After Quiz Ends	Try submitting answer after final puzzle	Submission disabled or ignored	Handled properly	<input checked="" type="checkbox"/> Pass
TC15	Final Score Accuracy	Play entire quiz intentionally submitting a mix of correct/wrong answers	Final score accurately reflects the number of correct answers	Score accurate	<input checked="" type="checkbox"/> Pass

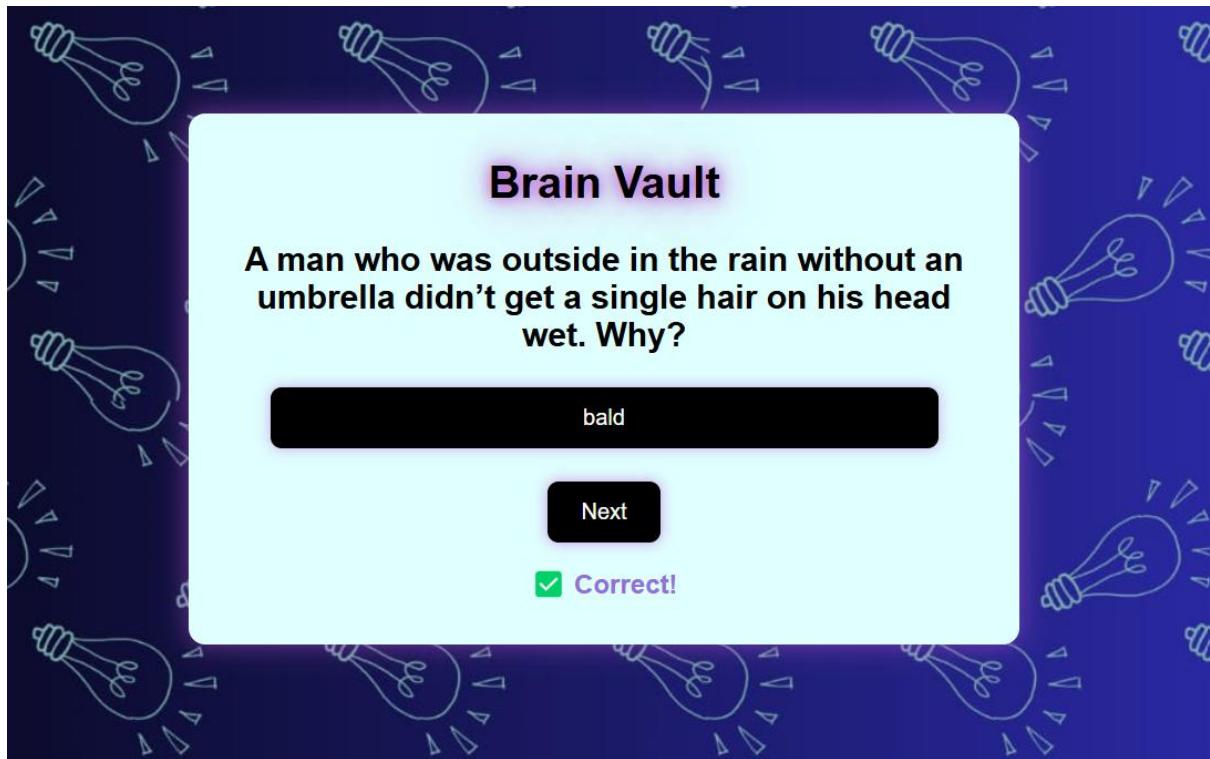
OUTPUT SCREENS:



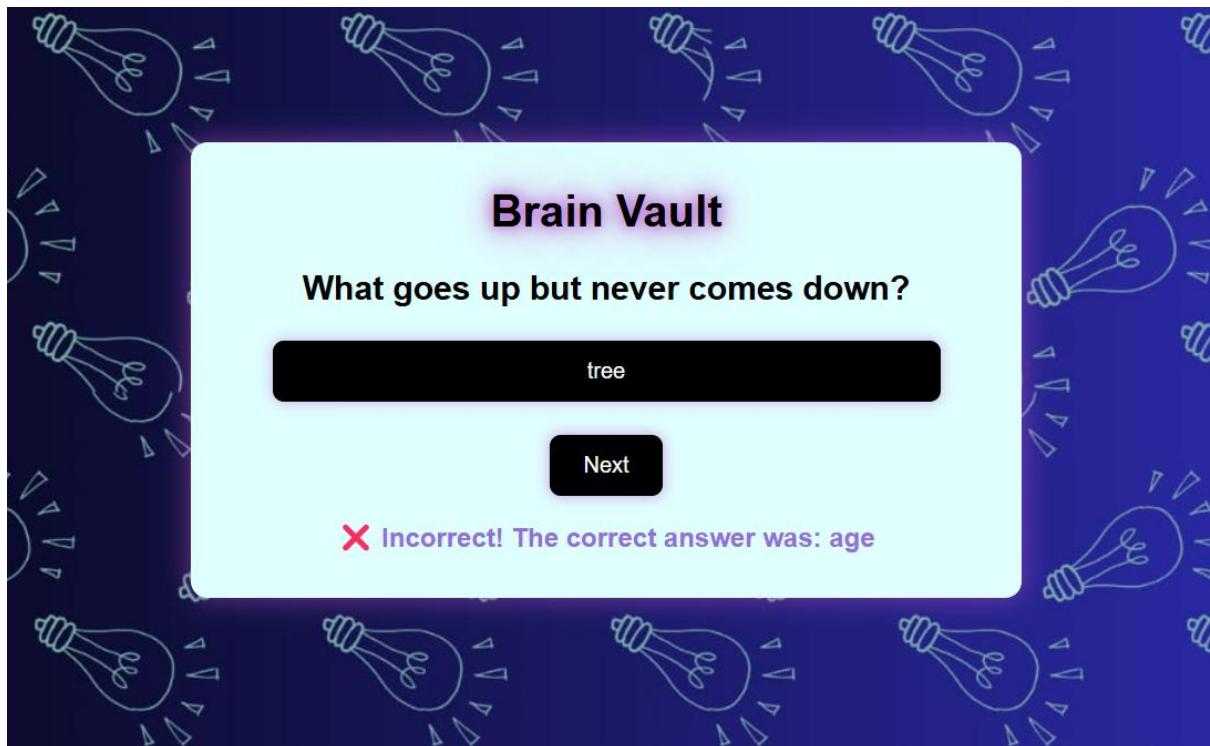
The user selects the number of questions as per their like. The questions range from 5 to 50. The user starts the game by clicking “Start Game” button.



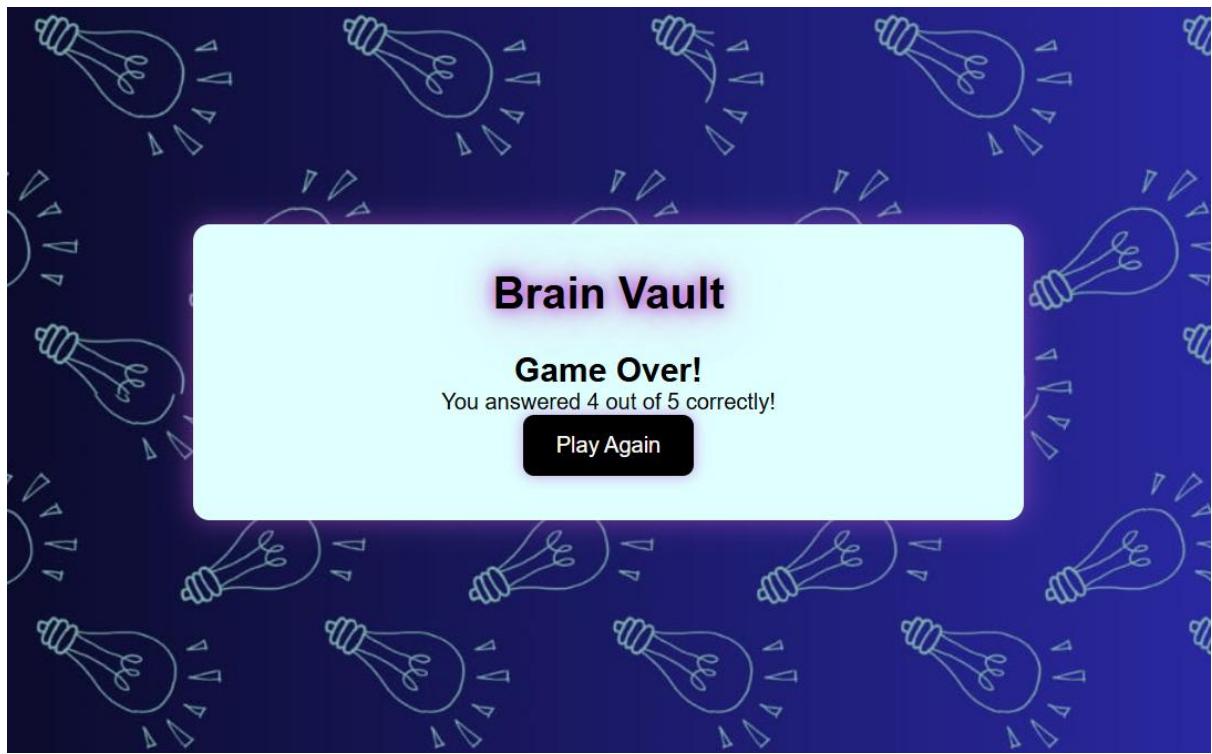
When user starts the game, answers will be taken by clicking “Submit” button.



After user submits the answer, it checks the answer is correct or not. It shows “✓ correct!” when the answer is correct.



It shows “✗ Incorrect!” when the answer is wrong. For the next question “Next” button is clicked.



At end of the game, the scores will be displayed to the user.

5. RESULTS:

Feature	Expected Result	Actual Result	Status
Webpage loads	"Brain Vault" homepage displays successfully	Homepage loaded properly with logo and title	<input checked="" type="checkbox"/> Success
Start Puzzle	User can start answering puzzles	Puzzles appear one after another	<input checked="" type="checkbox"/> Success
Submit Answer	User can submit an answer for each puzzle	Submission button works correctly	<input checked="" type="checkbox"/> Success
Check Answer	System validates answer and displays correctness	Correct/Incorrect feedback shown	<input checked="" type="checkbox"/> Success
Score Calculation	System keeps track of the score	Score updates after each question	<input checked="" type="checkbox"/> Success
Final Score Display	Show final result at the end of the quiz	Final score displayed clearly	<input checked="" type="checkbox"/> Success

6. CONCLUSION:

The Brain Vault project successfully achieved its primary goal of creating an interactive, engaging, and educational web-based puzzle game. Developed using HTML, CSS, and JavaScript, the project demonstrates a smooth and responsive user experience, allowing users to participate in a series of puzzles, submit their answers, and receive instant feedback.

Throughout the development process, careful attention was given to user interface design, puzzle validation, and score calculation, ensuring the application functions accurately and intuitively. Comprehensive testing confirmed that all functional requirements were met, with the application performing reliably across different browsers and devices.

By blending creativity with technical skills, Brain Vault not only provides an enjoyable experience for users but also showcases fundamental web development principles like event handling, dynamic content update, and simple data management.

In conclusion, the project was a success, meeting all its objectives while laying a strong foundation for future enhancements, such as adding more puzzles, implementing difficulty levels, maintaining user progress, or integrating a leaderboard.

7. REFERENCES:

W3Schools

- Website: <https://www.w3schools.com/>
- Used for learning and referencing HTML, CSS, and JavaScript syntax, DOM manipulation, and event handling.

MDN Web Docs (Mozilla Developer Network)

- Website: <https://developer.mozilla.org/>
- Referenced for detailed explanations of web technologies, JavaScript events, and best practices.

Stack Overflow

- Website: <https://stackoverflow.com/>
- Used to resolve minor coding issues, bug fixes, and understanding common JavaScript errors.

GitHub Documentation

- Website: <https://docs.github.com/>
- Used for managing version control and understanding .git configuration used in the project.

Freepik (for images, if used)

- Website: <https://www.freepik.com/>
- If any stock images (like the images.jpg) were taken from free sources.