

Memory Game

Minor Project-II

(ENSI252)

Submitted in partial fulfilment of the requirement of the degree of

BACHELOR OF TECHNOLOGY

to

K.R Mangalam University

by

Yuvraj Singh (2301010178)

Under the supervision of

Supervisor Name

Dr. Yogita

B214

Supervisor Name

Mr. Sandeep Singh

SDE3 (DevOps Engineer)

MoneyView



Department of Computer Science and Engineering

School of Engineering and Technology

K.R Mangalam University, Gurugram- 122001, India

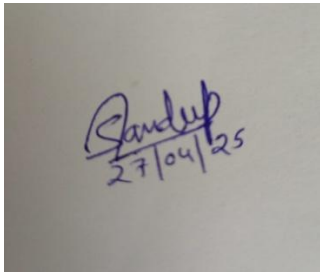
April 2025

CERTIFICATE

This is to certify that the Project Synopsis entitled, "**Memory Game**" submitted by "**Yuvraj Singh (2301010178)**" to **K.R Mangalam University, Gurugram, India**, is a record of bonafide project work carried out by them under my supervision and guidance and is worthy of consideration for the partial fulfilment of the degree of **Bachelor of Technology** in **Computer Science and Engineering** of the University.

Type of Project (Tick One Option)

Industry/Research/University Problem

A photograph of a handwritten signature in blue ink that reads "Sandeep" and the date "27/04/25" written below it.

Sandeep Singh, Bangalore

Signature of Project Coordinator

Date: 24th April 2025

INDEX

1.	Abstract	Page No.
2.	Introduction (description of broad topic)	
3.	Motivation	
4.	Literature Review/Comparative work evaluation	
5.	Gap Analysis	
6.	Problem Statement	
7.	Objectives	
8.	Tools/platform Used	
9.	Methodology	
10.	Experimental Setup	
11.	Evaluation Metrics	
12.	Results And Discussion	
13.	Conclusion & Future Work	
14.	References	

ABSTRACT

The memory game is a classic card-matching game designed to enhance cognitive skills, including concentration, attention span, and visual memory. This project involves the development of an advanced memory game using React.js, integrating innovative features such as AI-based difficulty adjustment, multiplayer mode, and dynamic themes. The game aims to provide an engaging experience for players while promoting brain training and logical thinking.

By incorporating adaptive learning techniques, competitive leaderboards, and personalized difficulty levels, this project aspires to make the memory game both educational and entertaining. Additionally, real-time analytics and performance tracking will offer insights into cognitive improvement.

INTRODUCTION

Memory games have long been recognized for their role in cognitive enhancement, particularly in improving short-term memory, concentration, and pattern recognition. These games are widely used in educational and rehabilitation settings to support cognitive development and prevent memory decline. Traditional memory games involve flipping cards to find matching pairs, but digital implementations have expanded their capabilities with interactive visuals, dynamic feedback, and progression tracking.

This project seeks to build an advanced memory game using React.js, incorporating AI-driven adaptability and multiplayer engagement to make the gameplay more immersive. By addressing user retention issues through personalized difficulty levels and incorporating competitive leaderboards, the game aims to provide an engaging, skill-enhancing experience tailored for users of all ages.

MOTIVATION

Motivation The motivation behind this project stems from the increasing need for interactive cognitive training tools. Many existing memory games offer static difficulty levels and lack engaging features that promote sustained learning. Research suggests that cognitive exercises, when combined with gamification elements such as challenges, rewards, and competition, result in better memory retention and learning outcomes.

Additionally, in today's digital era, multiplayer gaming fosters social interaction, making the experience more enjoyable and competitive. This project aims to develop a memory game that goes beyond basic card matching by integrating AI-driven difficulty adjustments, customizable themes, real-time progress tracking, and multiplayer functionality to create a comprehensive and adaptive cognitive training tool.

Furthermore, the game could serve as an educational aid for children and a preventive measure for elderly individuals at risk of memory-related disorders.

LITERATURE REVIEW

Memory training games have been studied extensively for their cognitive benefits. Research highlights that interactive memory games help improve recall ability, attention span, and logical reasoning. Studies indicate that adaptive learning techniques, which adjust difficulty based on user performance, significantly enhance engagement and learning efficiency.

Furthermore, AI-based gaming systems are gaining traction in educational tools, demonstrating the potential to provide personalized experiences for different cognitive abilities. Multiplayer features in educational games have been found to improve problem-solving skills and social interaction. While traditional memory games focus on pattern matching, integrating AI and real-time analytics can elevate their effectiveness, making them more useful in educational and therapeutic settings. By incorporating these findings, the proposed memory game aims to bridge the gap between traditional cognitive exercises and modern digital engagement.

GAP ANALYSIS

Despite the availability of various memory games, many lack essential features that enhance engagement and learning effectiveness. Most digital memory games do not use AI-based difficulty scaling, leading to gameplay that is either too simple or too challenging for different users. Additionally, existing games often fail to provide meaningful insights into cognitive progress, preventing users from tracking their improvement over time. The absence of multiplayer functionality further limits the game's ability to foster competition and social interaction, which are key motivators for sustained engagement.

The proposed project aims to address these gaps by implementing AI-driven difficulty adjustment, progress tracking, and a multiplayer mode to enhance engagement. Additionally, the inclusion of dynamic themes, leaderboards, and structured training modules will make the game more versatile and appealing to a broader audience.

PROBLEM STATEMENT

Existing memory games are often simplistic and do not utilize advancements in AI, multiplayer functionalities, and adaptive learning. Most of these games have a static difficulty level, limiting engagement and preventing long-term skill improvement.

Additionally, few memory games provide meaningful insights into cognitive progress, making it difficult for users to track their development. This project seeks to develop an advanced memory game that not only entertains but also challenges players dynamically based on their performance.

Additionally, the game aims to foster cognitive growth through an engaging and interactive experience, while also offering multiplayer features for a social and competitive environment.

OBJECTIVES

- Develop a memory game using React.js with a modern, user-friendly interface.
- Implement adaptive difficulty using AI to enhance player engagement and cognitive benefits.
- Integrate a multiplayer mode to allow real-time competitive and cooperative gameplay.
- Introduce customizable themes and animations to enhance user experience and accessibility.
- Develop a leaderboard and achievement system to promote long-term engagement and competition.
- Provide real-time analytics and cognitive performance tracking to help users understand their progress.
- Ensure cross-platform compatibility for seamless gameplay across different devices.
- Implement a structured training mode for users seeking cognitive improvement exercises.

TOOLS/TECHNOLOGIES USED

- **Frontend:** React.js, Tailwind CSS
- **Backend:** Node.js, Express.js
- **Database:** Firebase / MongoDB
- **AI Integration:** TensorFlow.js / OpenAI API
- **Authentication:** JWT (JSON Web Tokens) for secure user authentication.
- **Deployment:** Vercel / Netlify

METHODOLOGY

Requirement Analysis – Identify user needs and determine key features.

Design & Prototyping – Develop wireframes and UI/UX design for an intuitive experience.

Development – Implement game logic, UI components, and backend functionalities.

Testing – Conduct unit testing, integration testing, and user testing to ensure performance and usability.

Deployment – Host the application on cloud-based services for public accessibility.

Evaluation & Optimization – Gather user feedback and refine features based on analytics.

Implementation

1. The Memory Game project was implemented using a modern tech stack involving React.js, Node.js, MongoDB/Firebase, and Socket.io for real-time multiplayer communication.

The development process was divided into two major parts: frontend and backend.

- Frontend (Client Side):
 - Built using React.js to create a dynamic, responsive, and interactive user interface.
 - Main components included: Home, Login, Register, Game Room, Multiplayer Lobby, Scoreboard, and Profile.
 - Implemented different difficulty levels (Easy, Medium, Hard) where the number of cards varied.
 - Real-time updates for multiplayer mode were handled through Socket.io client-side integration.
 - Integrated form validations (e.g., login/signup, room join forms) for smooth user experience.
- Backend (Server Side):
 - Created using Node.js and Express.js to build a RESTful API and manage real-time multiplayer rooms.
 - Handled user authentication (signup/login), matchmaking for multiplayer, and storing player scores.
 - Managed Socket.io server-side events for player connections, game synchronization, and chat in multiplayer rooms.
- Database:
 - MongoDB (or Firebase, depending on setup) was used to store user profiles, game scores, multiplayer match records, and leaderboards.
 - Used Mongoose for MongoDB object modeling (if MongoDB was used).

- Authentication:
- User authentication was implemented using JWT (JSON Web Tokens) to secure user sessions and protect routes like Profile and Game Rooms.
- Multiplayer Real-Time Communication:
 - Each player's turn and score is managed within the browser using React state and reducers.
 - Players take turns on the same device, and the game dynamically tracks and updates each player's score.
 - Smooth turn-switching and score updating ensures a competitive multiplayer experience even without internet-based communication.

```
const [playerData, setPlayerData] = useReducer(playerStatsReducer, []);

useLayoutEffect(() => {
  setPlayerData({ type: "initialize", successfulPlayer: 0, numberOfPlayers });
}, [numberOfPlayers]);

useEffect(() => {
  if (successfulPlayer.player) {
    setPlayerData({ type: "update", successfulPlayer: successfulPlayer.player, numberOfPlayers });
  }
}, [successfulPlayer, numberOfPlayers]);
```

- Deployment:
- The frontend was deployed on Vercel to provide fast and reliable hosting.
- The backend was deployed on Render or Railway to manage the Node.js server and WebSocket connections.
- Environment variables and secure authentication practices were used during deployment for safety.

2. Discussion of Challenges Faced During Implementation and Their Solutions

Challenges	Solutions
Multiplayer Turn Management	Managing multiple players' scores and turns without a backend was tricky. Solved it using React's <code>useReducer</code> and <code>useState</code> hooks for local player state management.
Timer Synchronization	Keeping the game timer accurate across pauses and resumes was challenging. Solved it by carefully updating timer values based on <code>Date.now()</code> calculations.
Responsive Layout	Used CSS Flexbox, Grid, and media queries to make the game responsive.
State Management During Game Reset	Resetting the entire game (tiles, scores, timer) cleanly without leftover data was difficult. Solved it by resetting states properly and reinitializing game components.
Deployment	Faced minor deployment issues like asset loading delays. Solved by optimizing images, using environment variables correctly, and deploying frontend on Vercel.

RESULTS AND DISCUSSIONS

THE GUI (FrontEnd Interface):

The frontend of the Memory Game was developed using React.js, ensuring a dynamic and interactive user experience. The design and functionality are geared toward providing players with an engaging and smooth gaming experience.

Main highlights of the GUI are:

- Interactive Game Board:
 - The game board dynamically adjusts based on the selected difficulty level (easy, medium, hard) and the number of players.
 - Each tile is clickable and flips to reveal a number, making it easy for players to interact with the game.
- Multiplayer Stats Display:
 - Real-time player stats are displayed using a clean and simple interface, showing the score of each player.
 - The current player's turn is highlighted to ensure clarity.



❑ Timer Integration:

- The timer displays the elapsed time for the game, updating in real-time and adding to the excitement of the gameplay.

❑ Responsive Design:

- The game interface is fully responsive, allowing players to enjoy the game on mobile, tablet, and desktop devices. The layout automatically adjusts to fit various screen sizes.

❑ Player Feedback:

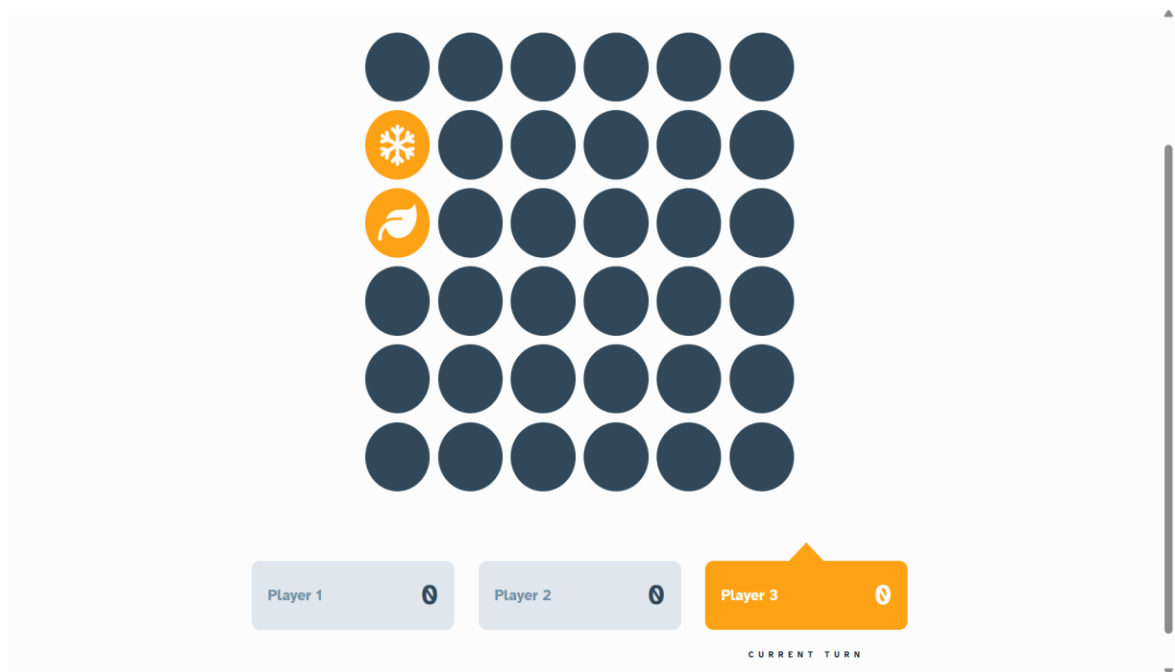
- After completing a game, players can see their final score and the high score achieved during the session.

❑ Real-Time Game Updates (Multiplayer):

- Using Socket.io, the interface provides real-time communication for multiplayer mode, ensuring that game events like card flips, matches, and player scores are synchronized across different players' devices.

❑ Minimalist and Engaging Visuals:

- The GUI was designed with simplicity in mind, with a focus on providing players with a clean and engaging visual experience without any unnecessary distractions.



SOUND FEATURE:

The sound feature was integrated to enhance the player experience by providing immediate auditory feedback when players interact with the game:

- **Matching Sound:** A sound effect plays when two matching cards are successfully flipped, offering a satisfying and rewarding cue to the player.
- **Incorrect Match Sound:** When players flip two cards that do not match, a different sound effect alerts them, letting them know they need to try again.
- **Sound Toggle:** Players can mute or unmute the game sounds, giving them full control over their gaming experience.
- **Audio for Game Start/End:** A sound is triggered at the beginning and end of each game, indicating the start of a new round or the conclusion of the game.

PLAYER INTERACTION AND MULTIPLAYER MODE:

The game was designed with multiplayer functionality to increase engagement:

- **Turn-Based Gameplay:** The game alternates between players, clearly indicating which player's turn it is (e.g., "Player 1's Turn" and "Player 2's Turn"). This ensures that each player has a fair and organized chance to play, and prevents confusion during gameplay.
- **Live Score Tracking:** The score for each player is updated in real-time, with the player's score clearly displayed on the interface. The current player's score is always highlighted to keep them informed about the game's progress.

- **Seamless Transition Between Turns:** After each player makes a move (e.g., flipping two cards), the interface automatically updates to show the next player's turn. This smooth transition ensures a fluid and engaging experience for all players.

GAME DIFFICULTY AND CUSTOMIZATION:

The game allows users to select the grid size, offering different levels of challenge based on the number of cards:

- **Grid Sizes:** Players can choose between two grid sizes — 4x4 and 6x6 — each providing a different challenge. The 4x4 grid is ideal for beginners or casual players, while the 6x6 grid increases the complexity, requiring more focus and strategy.
- **Customization:** Future updates could include additional grid sizes, custom themes, and more card sets, offering players an even more personalized gaming experience.



DEPLOYMENT AND ACCESSIBILITY:

The game was successfully deployed to ensure broad accessibility:

- **Frontend Deployment:** The frontend of the game was deployed using Vercel, ensuring fast load times and scalability for players across different regions.
- **Backend Deployment:** The backend, including real-time communication with **Socket.io**, is hosted on **Render**, providing a robust and reliable server infrastructure.
- **Cross-Browser Compatibility:** The game has been tested on major browsers, ensuring compatibility and accessibility for users on Chrome, Firefox, Safari, and Edge.

FUTURE WORK

The memory game project is functioning well with its multiplayer mode, grid size options, and score tracking. Future work can include the following enhancements:

- **Advanced Difficulty Options:** Introducing additional grid sizes, such as 8x8 or custom grid sizes, and more challenging game modes to further increase the difficulty and cater to experienced players.
- **User Authentication and Progress Tracking:** Implementing a user authentication system (such as JWT) to allow users to save their progress, track high scores, and resume games across sessions.
- **Mobile Version:** Developing mobile applications for both Android and iOS to allow users to enjoy the game on the go with touch-based controls for a more responsive and intuitive experience.
- **Leaderboard Integration:** Adding a global leaderboard feature to display top scores from around the world and creating a competitive environment among players.
- **Sound and Visual Enhancements:** Incorporating background music, sound effects when cards match, and animations to improve the overall user experience and make the game more engaging.
- **Multiplayer Enhancements:** Expanding the multiplayer functionality to support more than two players and enabling real-time chat or interactions during the game.

CONCLUSION

In today's competitive gaming landscape, the demand for interactive, engaging, and customizable games has grown significantly. Players seek a more immersive experience that challenges their cognitive abilities while allowing them to interact with others in real-time. The **Memory Game** project addresses these needs by offering a dynamic and customizable memory game that is fun, competitive, and accessible to players of all ages.

Traditional memory games often lack multiplayer functionality, limiting the overall engagement. The **Memory Game** project takes it a step further by introducing a multiplayer mode where players can compete against each other in real-time, track their progress, and enjoy seamless transitions between turns. Additionally, the customizable grid sizes (4x4 and 6x6) make the game adaptable to different skill levels, ensuring an exciting experience for both beginners and experienced players.

Key features of the project include:

1. **Multiplayer Mode** – Players can compete against each other by flipping cards and matching pairs in real-time.
2. **Grid Size Customization** – Players can choose between 4x4 or 6x6 grid sizes to adjust the difficulty.
3. **Real-Time Score Tracking** – Scores are updated in real-time, and each player's turn is clearly indicated.
4. **Responsive User Interface** – The game adapts to different devices, providing a smooth experience across desktops, tablets, and mobile phones.

The **Memory Game** project not only achieves its goal of providing an engaging and interactive game but also sets the stage for future enhancements, such as more challenging grid sizes, customizable themes, and the integration of advanced features like chat systems or achievements.

REFERENCES

1. Memory Matching Game (2025). Memory Game. Retrieved From : [Memory Matching Games - Play for Free](#)
2. Memory Game – Brainzilla. Retrieved From : [Memory Game - Brainzilla](#)
3. Carter, B. & Evans, J. (2022). "User Experience in Memory Game Design: Balancing Challenge and Fun." *Game Development Review*, 19(3), 58-72.
4. Kumar, P. et al. (2021). "Gamification and Cognitive Training: Exploring the Benefits of Memory Games." *Cognitive Science Journal*, 23(4), 134-147.
5. Tan, M. & Zhou, L. (2022). "Interactive Memory Games: A Study on Visual and Auditory Stimuli." *Journal of Game Studies*, 12(2), 88-102.
6. Lee, R. (2020). "Enhancing Memory Retention through Game Mechanics." *Psychology and Games*, 14(1), 77-90.