A

PROJECT REPORT

ON

"ARCADE GAMES PARADISE"

SUBMITED
TO

SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

FOR THE PARTIAL FULFILMENT OF

MASTER OF COMPUTER APPLICATION

(MCA-I, SEM.-I)

BY

KOKANE VISHAL DATTATRAY

UNDER THE GUIDANCE OF

PROF. YOGESH SHARMA

THROUGH

THE DIRECTOR
SINHGAD INSTITUTE OF MANAGEMENT AND COMPUTER APPLICATION
(SIMCA), NARHE, PUNE (AY. 2022-2023)

SINHGAD TECHNICAL EDUCATION SOCIETY'S
SINHGAD INSTITUTE OF
MANAGEMENT & COMPUTER APPLICATION
(Affiliated to Savitribai Phule Pune University & Approved by AICTE)
'NAAC' Accredited with 'A' Grade
S. No. 49/1, Off Westernly Bypass, Pune – Mumbai Expressway, Narhe, Pune – 411041, Tel : (020) 66831896 / 66831908 / 66831907
E-mail : director_mca_simca@sinhgad.edu Website : www.sinhgad.edu

Prof. M. N. Navale
M.E. (Elec), MIE, MBA
FOUNDER PRESIDENT

Dr. (Mrs.) Sunanda M. Navale
B.A., P.P.M., Ph.D.
FOUNDER SECRETARY

Dr. Vijaya Puranik
M.Sc., MMM,MPM, Ph.D.
DIRECTOR, SIMCA

CERTIFICATE

This is to certify that, the project entitled "***Arcade Games Paradise*",** being submitted for the partial fulfilment of the degree of **Master of Computer Application** by him to **Sinhgad Institute of Management and Computer Application affiliated to Savitribai Phule Pune University, Pune** is the result of the original work completed by ***Kokane Vishal Dattatray*** under the guidance of ***Prof. Yogesh Sharma***.

To the best of our knowledge and belief, this work has not been previously submitted by the award of any degree or diploma of Savitribai Phule Pune University or any other University.

PLACE:

DATE:

Prof. Yogesh Sharma	**Prof. Navanath Choudhari**	**Dr. Vijaya Puranik**
Internal Guide	**Project Co-Ordinator**	**Director SIMCA**

External Examiner

DECLARATION

I, the undersigned hereby declare that the project titled "**Arcade Games Paradise", being** submitted for the award of degree of **Master of Computer Application** by me to **Sinhgad Institute of Management and Computer Application (SIMCA) affiliated to Savitribai Phule Pune University is** the result of an independent work carried out under the guidance of Prof. **Yogesh Sharma,** is my original work. Further I declare that this project has not been submitted to this or any Institution for the award of any degree.

PLACE: PUNE

DATE:

Kokane Vishal Dattatray

(Student)

ACKNOWLEDGEMENT

We take this opportunity to express our gratitude towards all those who have helped us throughout the successful completion of our project Firstly, we would like to thank our Director **Dr. Vijaya Puranik**, and project coordinator **Prof. Navanath Choudhari** of Masters in Computer Application, for permitting us to complete our project and for showing faith in us and allowing us to develop **Arcade Games Paradise**. We must convey our gratitude to our guide **Prof. Yogesh Sharma** for giving us the constant guidance of inspiration and help in preparing the project, personally, correcting our work and providing encouragement throughout the project

Kokane Vishal Dattatray

Student Name

INDEX

1. Introduction

This project is aimed at developing a website for online gaming. It provides the users more pleasure and gladdening his mind by playing these retro-traditional games

Arcade Games Paradise is an online platform that allows users to play and interact with various types of games. Our project can feature a range of games such as action, strategy, adventure, puzzle, simulation, and sports games, among others.

Arcade Games Paradise can offer a personalized user experience by allowing users to create and manage their accounts and track their gaming progress.

Our goal is to provide users with a fun and engaging gaming experience that keeps them coming back for more

1.1 Abstract

Arcade Games Paradise project is a web-based application that allows users to access and play various games online. The project aims to provide an easy-to-use platform for users to enjoy games without the need to download or install any software.

The website includes a variety of games such as arcade games, puzzle games, strategy games, among others. The website's primary objective is to provide a fun and engaging experience for users of all ages.

The website will have a user registration system that allows users to create accounts and save their high sores in the games.

The project is developed using modern web development technologies and agile methodologies, and regular user feedback is used to improve the website's functionality and user experience.

1.2 Existing System and Need for System

Existing system lacks in below points:

- Website contains Ads
- Paid games
- Not all games at same place to play
- Not all games on browser
- High internet connection requires

2. Proposed System

Proposed system contains following points:

- No ads in webpage
- Free to play games
- All your favorite games at one place
- All Brower based games easy to access
- Slow internet connection is necessary

2.1 Objectives of Proposed System

The project's objectives may include the following:

- To provide a fun and engaging gaming experience that keeps users coming back for more.
- To offer a personalized user experience that allows users to create and manage their accounts and track their gaming progress.
- To offer a platform for game developers to showcase their games and gain exposure to a wider audience.
- To provide a platform for advertisers to reach a highly engaged and targeted audience.

2.2 Users Summery

The Arcade Games Paradise is a platform where users can browse, play games. The website allows users to create accounts and sign in to access their game scores and personal data.

Users can play games directly on the website, using a user-friendly interface with controls, menus, and other relevant features. The website tracks the user's progress and saves their game scores.

The website has strong security measures in place to protect users' personal and financial data and prevent unauthorized access. The website loads quickly and responds to user actions promptly, is user-friendly and intuitive, and is compatible with different web browsers and devices. The website is also accessible to users with disabilities and complies with accessibility standards. The website is available and operational 24/7, with minimal downtime for maintenance or upgrades, and has backup and recovery mechanisms in place.

2.3 Scope of the system

- Functional requirements:

  These are the features and functions that the system must have to meet its objectives. Examples include:

  User registration and authentication
  Game catalog
  Game play and progress tracking

- Non-functional requirements:

  These are the qualities and characteristics of the system that are important for its usability, performance, security, and other aspects. Examples include:

  User-friendly and intuitive user interface
  Fast loading and response times
  High availability and minimal downtime
  Compatibility with different web browsers and devices
  Compliance with accessibility standards
  Strong security measures to protect user data and prevent unauthorized access
  Reliable backup and recovery mechanisms to ensure data integrity and availability.

- Constraints:

  These are the limitations or restrictions that affect the design, development, and deployment of the system. Examples include:

  Budget and resource constraints
  Time constraints for development and testing
  Technical limitations of the hardware and software used to build the system
  Legal and regulatory requirements such as data privacy and security laws.

2.4 System requirements

2.4.1 Software Requirements

- Technology: JavaScript, HTML, CSS.
- Web Browser (Chrome)
- Visual Studio Code
- Google Firebase

2.4.2 Hardware Requirements

- Processor: Intel/AMD dual-core or above
- RAM: 4 GB RAM
- Disk: 10 GB of SSD/HDD
- Operating system: Windows 10, Linux

3. Requirement determination and Analysis

Fact Finding methods

- Interviews: Interviews involve face-to-face discussions with users, and subject matter experts to gather information about their needs and expectations. These interviews can be structured or unstructured and can be conducted in person or remotely.

- Observation: Observation involves watching users as they interact with the current system or with prototypes to gather information about their behavior, preferences, and pain points. This can be done in a lab or in the field, depending on the needs of the project.

- Document analysis: Document analysis involves reviewing existing documentation, such as user manuals and technical specifications, to identify requirements and constraints that may have been overlooked or not clearly communicated.

- Focus groups: Focus groups involve bringing together a small group of users to discuss their needs and expectations in a group setting. This can be used to gather both quantitative and qualitative data and to identify common themes and issues.

- Prototyping: Prototyping involves creating a simple, functional model of the website to test with users. This can be used to gather feedback and identify areas for improvement before the final system is developed.

- Brainstorming: Brainstorming involves generating ideas and solutions through a collaborative and creative process with subject matter experts. This can be used to identify new requirements and ideas that may not have been considered previously.

3.2 Feasibility study

- Technical Feasibility:

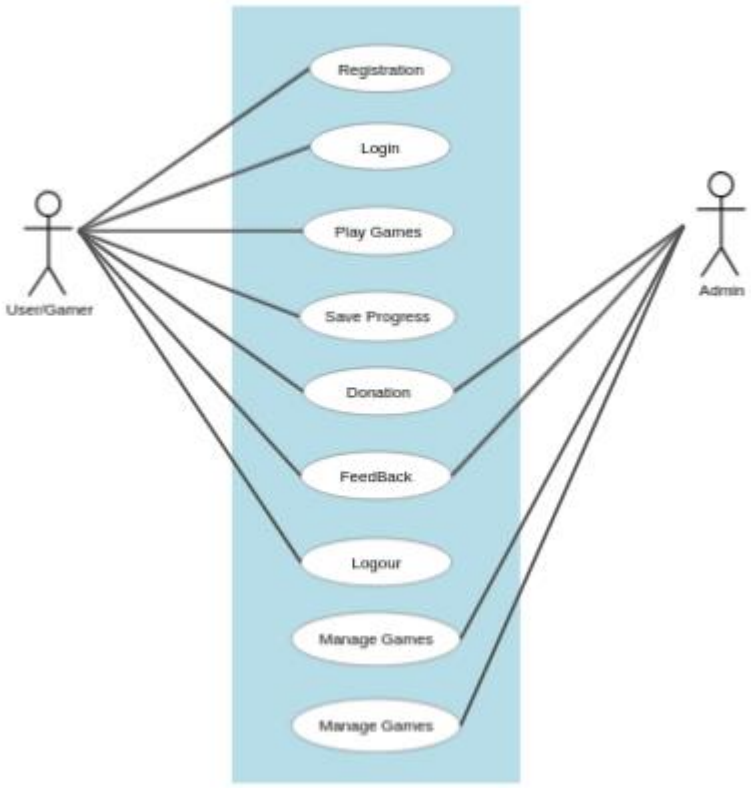
  The technical feasibility study determined that the project is technically feasible. The website can be developed using modern web development technologies such as HTML5, CSS3, and JavaScript. The website can be optimized for mobile and desktop devices, and scalable IT infrastructure can be implemented to handle traffic spikes.

- Operational Feasibility:

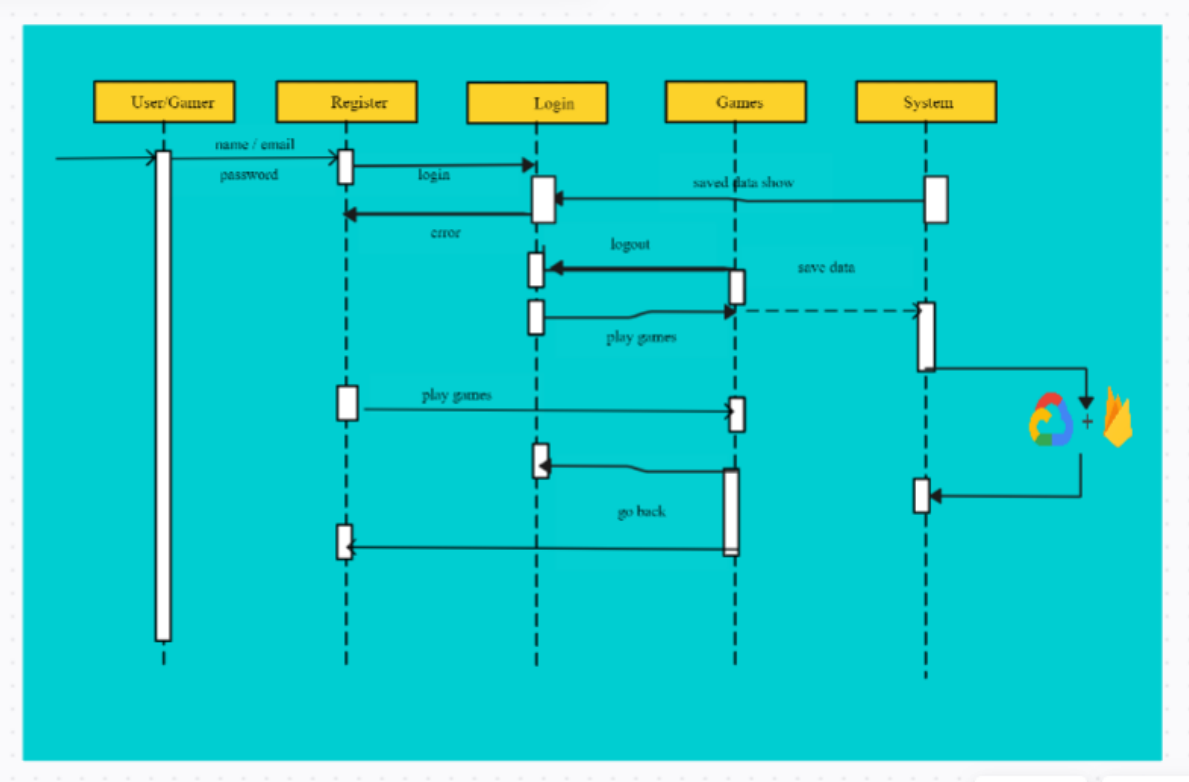
  The operational feasibility study concluded that the project is operationally feasible. The website can be developed using agile software development methodologies to ensure that it is delivered on time and within budget. User testing and feedback will be used to refine and improve the website's functionality and user experience.

4. System Analysis and Design

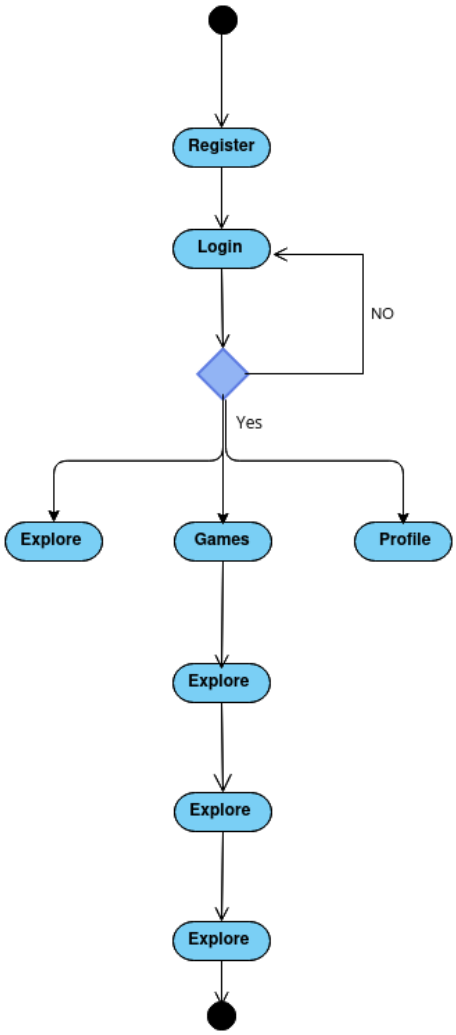
4.1 Use Case Diagram



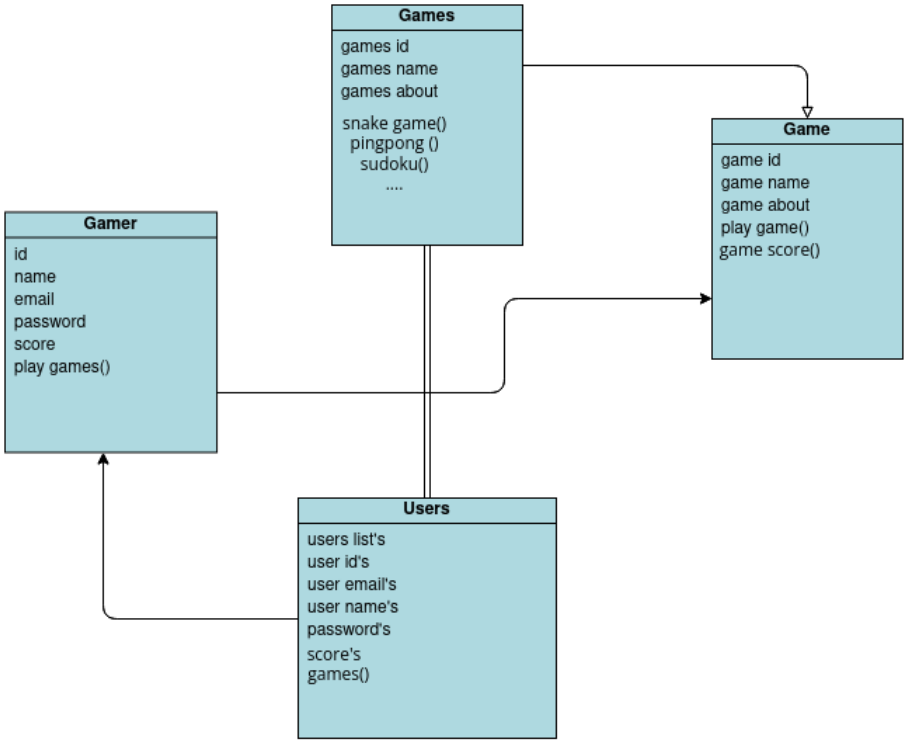
4.2 Sequence Diagram



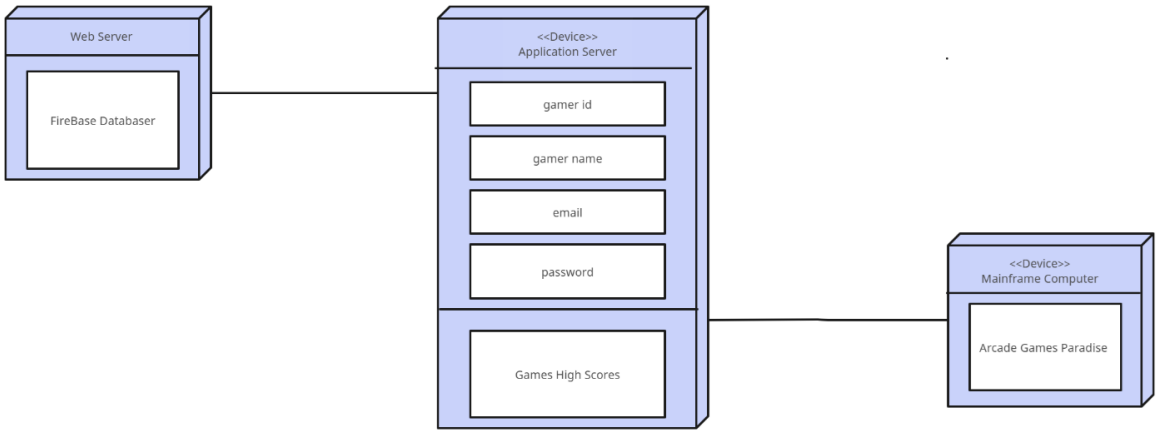
4.3 Activity Diagram ERD



4.4 Class Diagram

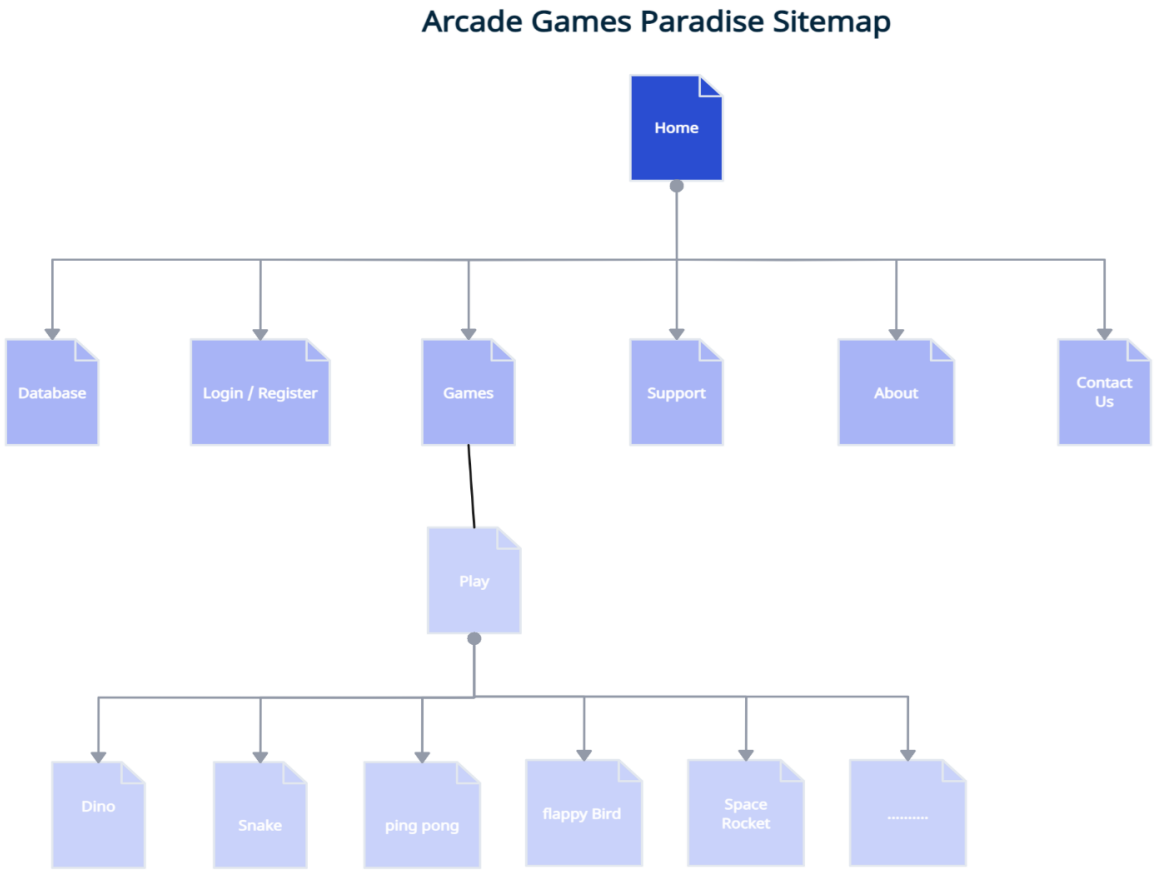


4.5 Deployment Diagram



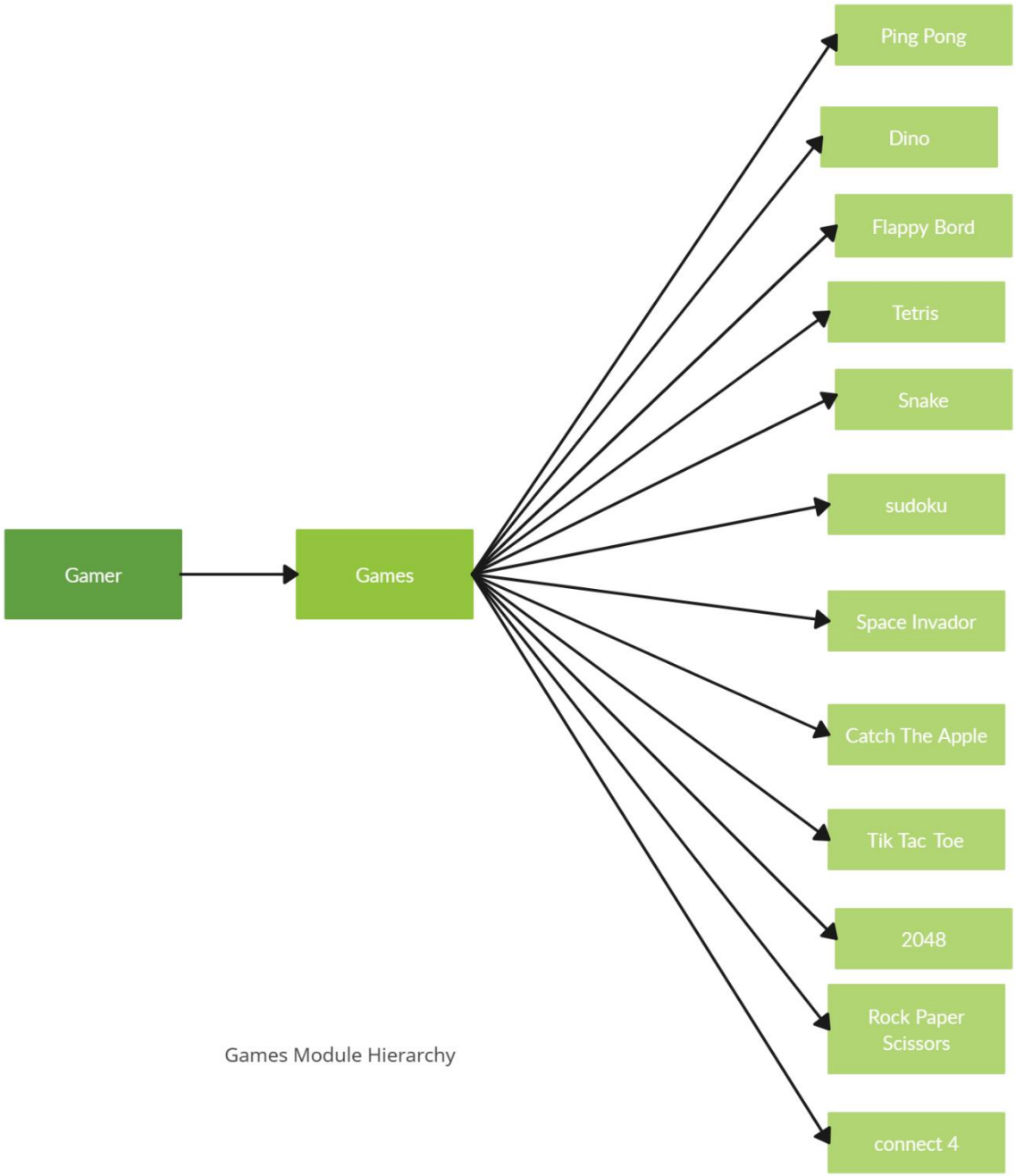
Deployment Diagram Arcade Games Paradise

4.6 Web Site Map Diagram



WEBSITE MAP of Arcade Games Paradise

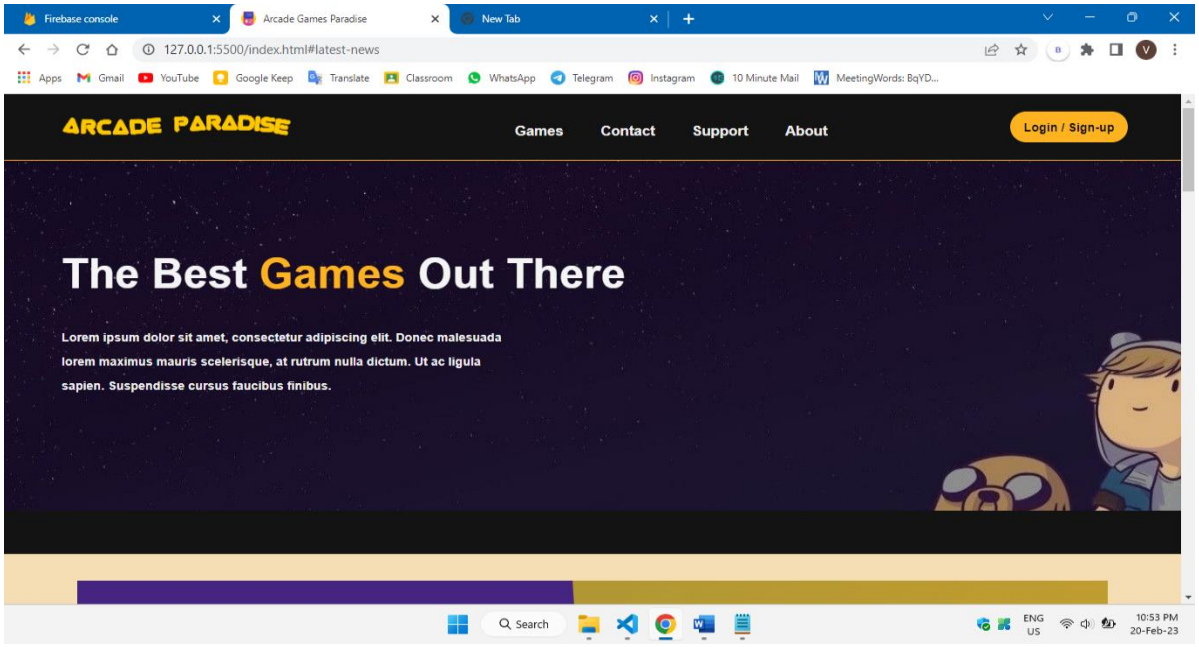
4.7 Games modules hierarchy



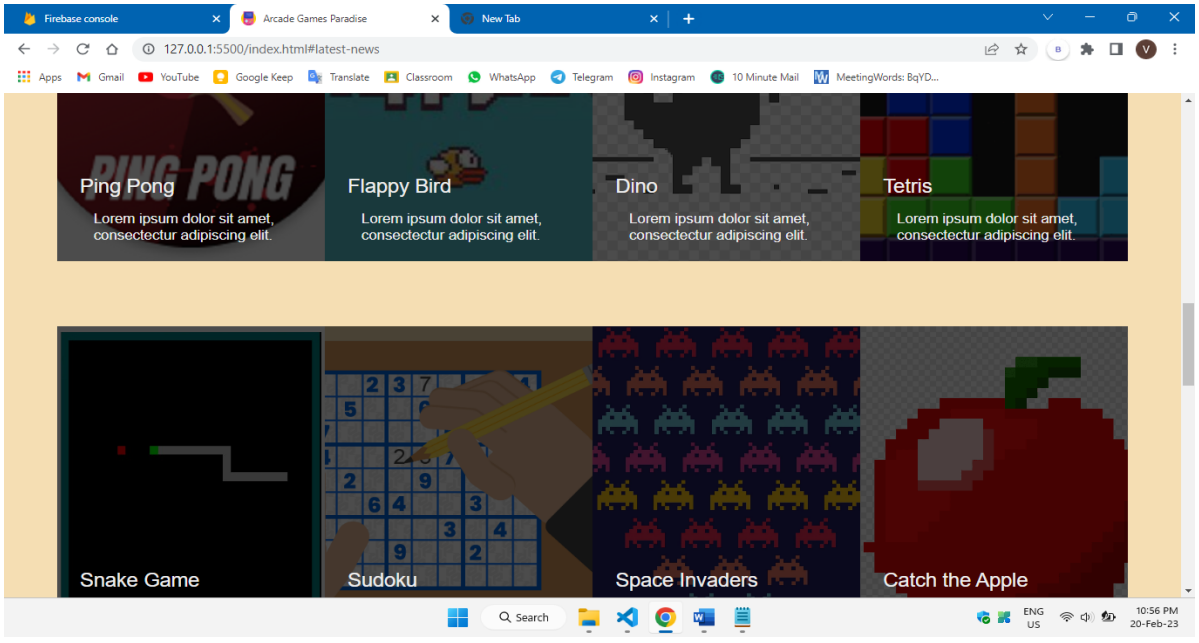
Games Module Hierarchy

4.8 User Interface Design (Screens)

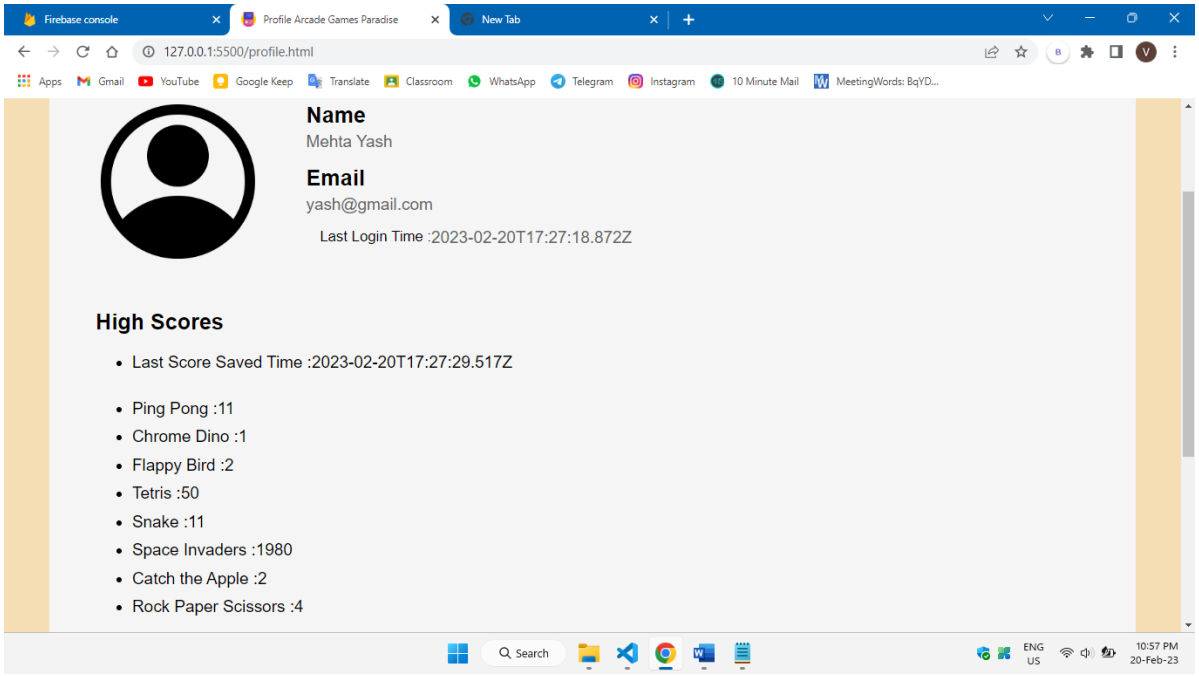
//Home Page



//Games

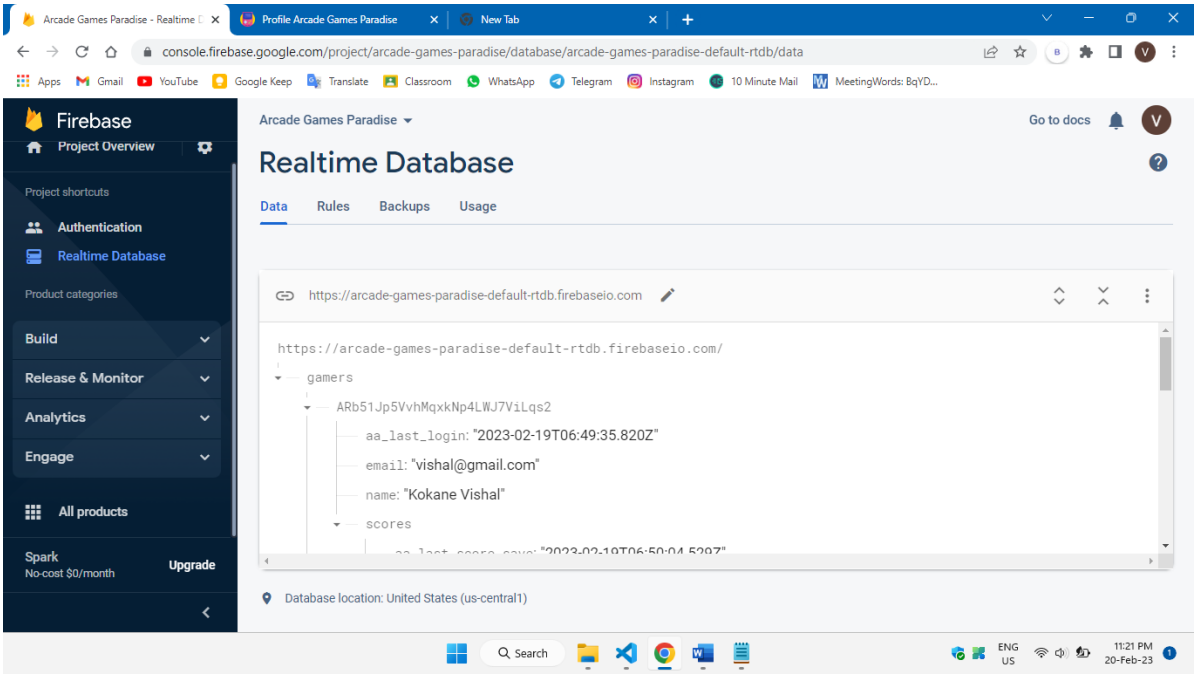


//Profile Page

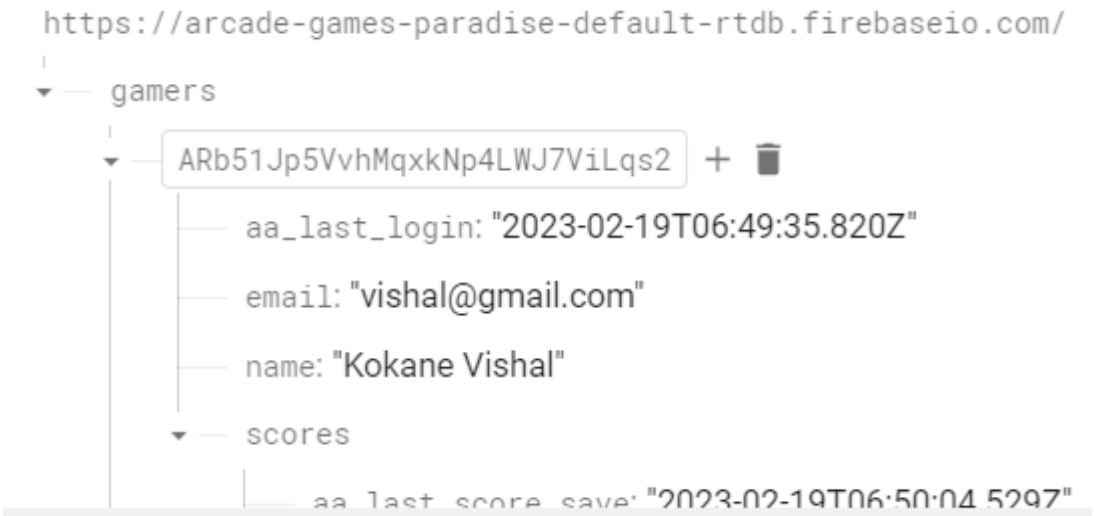


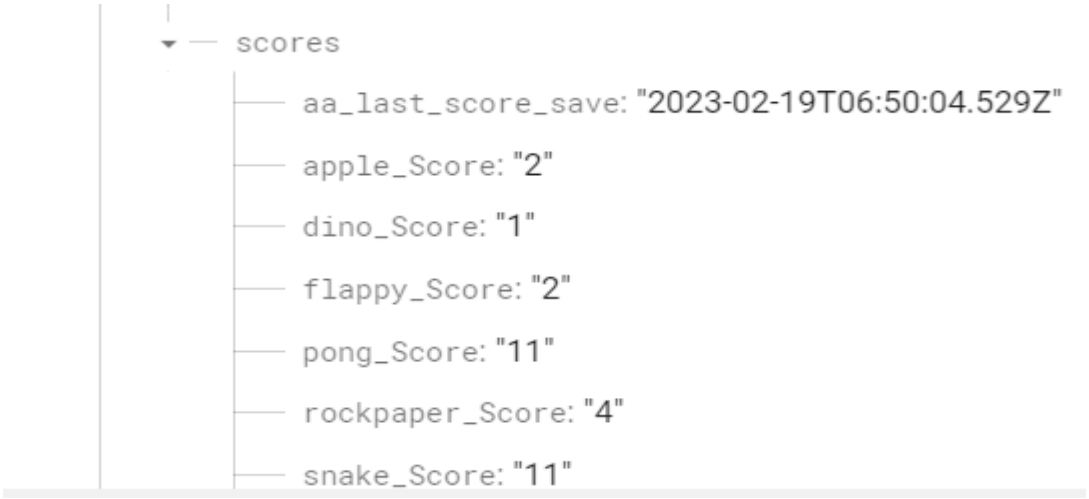
4.9 Table structure

//Database

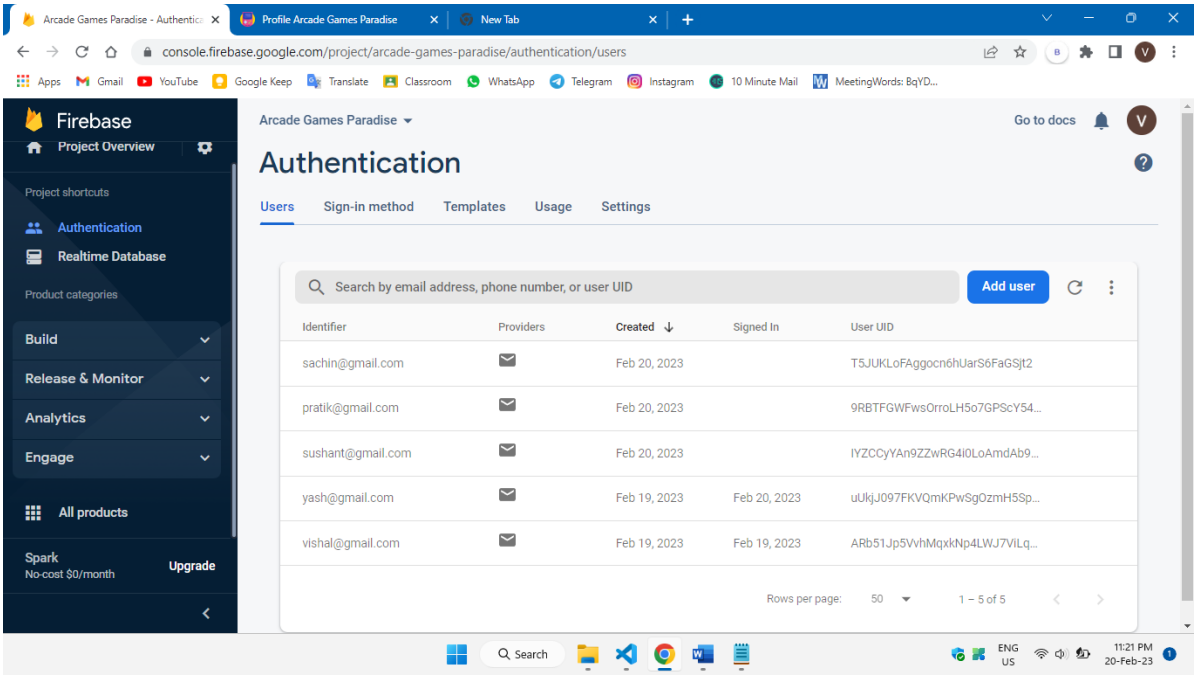


//Database Structure





//User table (authentication)



5. Coding

Arcade Games Paradise Project Coding phase

- Front-end:
  Front-end coding involves coding the user interface using HTML, CSS, and JavaScript. The goal is to create an attractive and intuitive user interface that is responsive and easy to use.

- User management:
  user authentication, authorization, and profile coding include coding the login, registration, and profile pages. It also involves implementing a user profile system that allows users to view their personal information and game scores. Additionally, it involves implementing authorization mechanisms to restrict access to certain features or content based on user roles and permissions.

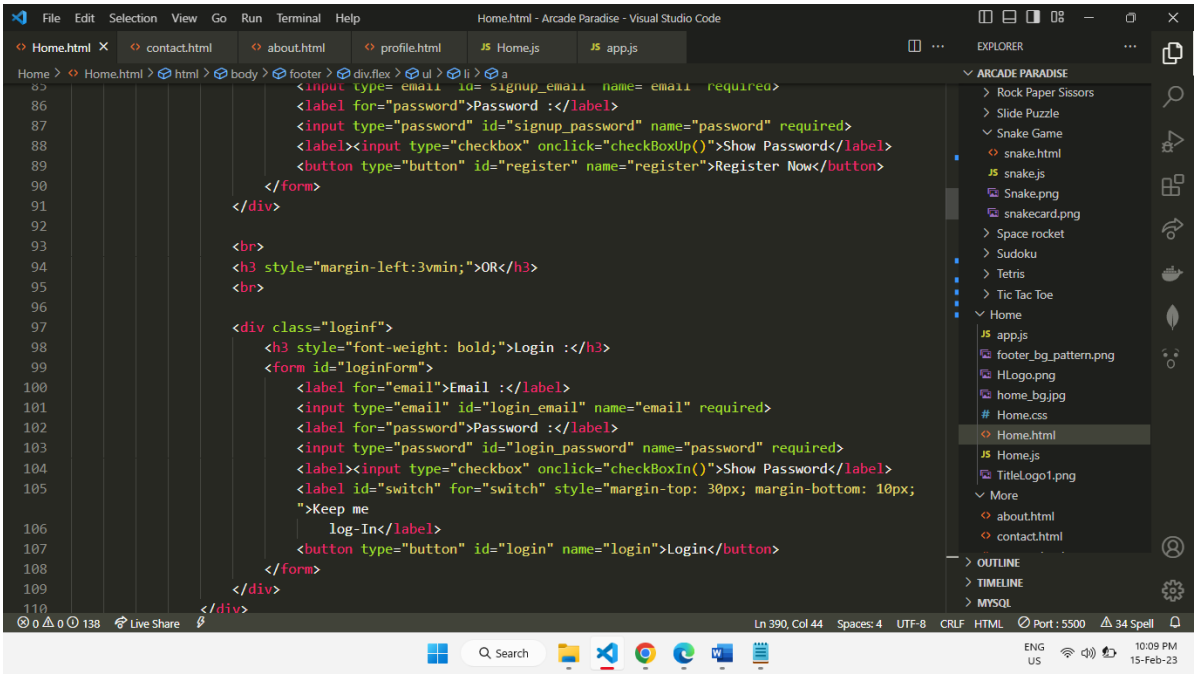
- Game coding:
  Game coding involves implementing the game mechanics for each game, including rules, scoring, and levels, using JavaScript and its frameworks. The goal is to create engaging and immersive game experiences that keep users coming back to the website.

- Back-end
  Back-end coding involves coding the server-side logic and database connections using JavaScript and firebase. The goal is to handle the business logic, user authentication, data storage, and other server-side functions necessary for the website to function properly.

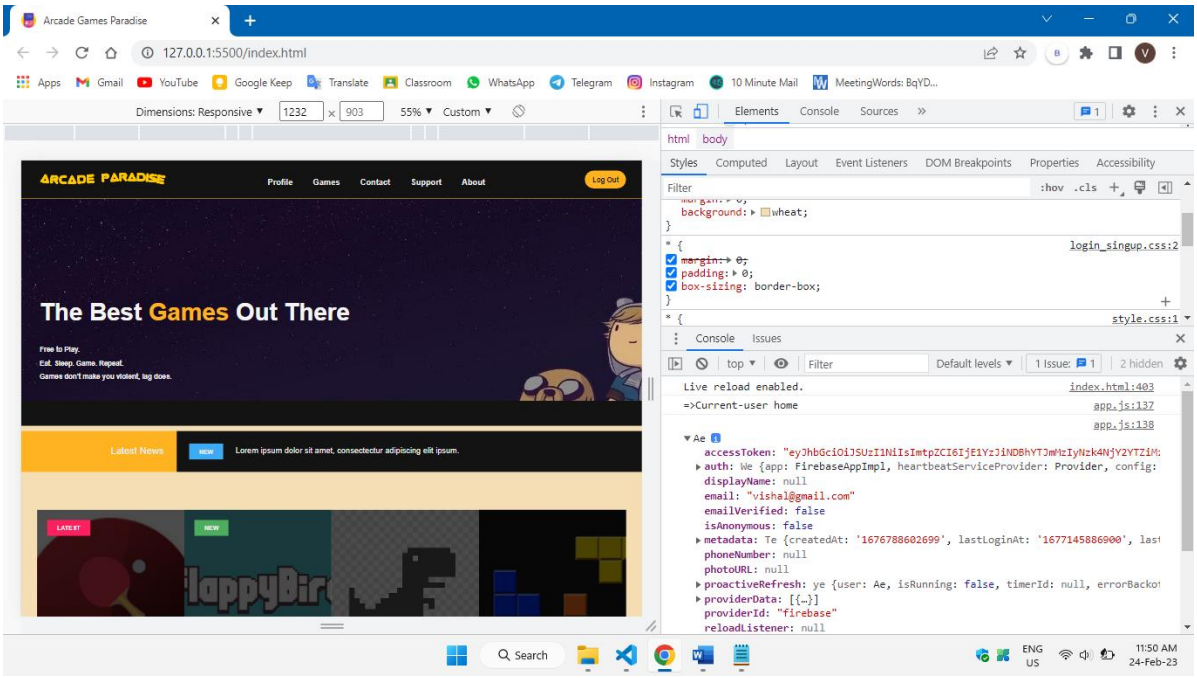
- Game database management
  Game database management involves coding the game database schema using a database management system such as Firebase and implementing CRUD operations for game data. It also includes implementing saving game scores in the database and displaying on user profile.

- Codding tool
  The coding tools involves Visual Studio Code is used for coding front-end and back-end logic. The VS code IDE come with features like syntax highlighting, code completion, and debugging capabilities.
  We also used some JavaScript frameworks and libraries like kaboom js. It can be used to create games. And also google firebase for Database storage and management.

- Sample Coding Screen



- Browser console



6. Testing

Arcade Games Paradise Project Testing phase

- Functionality Testing:
  Verify that all games can be played without issues, including the gameplay mechanics, scoring, and user interface.
  Test all user registration and account management features to ensure they work correctly.

- Compatibility Testing:
  Test the website on different browsers, including Chrome, Firefox, and Safari, to ensure that it works correctly.
  Verify that the website is responsive on different laptops and desktops.
  Test the website's performance on different internet connections, including 3G, 4G, and Wi-Fi.

- Performance Testing:
  Perform load testing to ensure that the website can handle a high volume of traffic and requests.
  Test the website's response time to ensure that it is quick and responsive.
  Verify that the website can handle multiple users playing games simultaneously.

- Security Testing:
  Test the website's user authentication and authorization process to ensure it is secure.
  Test the website's payment system to ensure that user payment details are secure.
  Verify that the website is protected against common attacks, such as SQL injection and cross-site scripting.

- User Acceptance Testing:
  Test the website's user interface to ensure that it is easy to navigate and understand.
  Perform user acceptance testing on all games to ensure that they meet user expectations.
  Verify that the website meets accessibility requirements, including screen readers and keyboard navigation.

Testing Techniques:

- Black-box testing: This involves testing the game website without knowing the internal workings of the code, by treating it as a black box. Testers can test the input/output and expected behavior of the website without having access to the codebase.

- Gray-box testing: This involves testing the game website with partial knowledge of the code, by treating it as a gray box. Testers can test the website with partial understanding of the code and may use debugging tools to analyze the code.

- Exploratory testing: This involves testing the game website by exploring its various features and functionalities. Testers can identify defects or issues by actively using the website and analyzing its behavior.

- Regression testing: This involves testing the game website after making changes to the code to ensure that previously working features are still functional.

- Usability testing: This involves testing the game website to ensure that it is user-friendly and easy to use. Testers can observe users interacting with the website and provide feedback on its user interface, functionality, and overall experience.

- Unit testing: This involves testing individual functions or components of the game code to ensure that they work as expected. Unit testing frameworks like Jest or Mocha can be used to automate this process.

- Integration testing: This involves testing how different parts of the game code work together to ensure that they integrate properly. Integration testing frameworks like Cypress or Selenium can be used to automate this process.

- Accessibility testing: This involves testing the game website's accessibility to ensure that it is accessible to users with disabilities. Tools like Google Lighthouse or Axe can be used to automate this process.

- Testing Table

Sr.No.	Data input	Excepted Output	Actual Output	Pass/Fail
1	All fields are empty	Error: *All fields are Compulsory*	Error: *All fields are Compulsory*	Pass
2	Username	Error: *Username is compulsory*	Error message: *Invalid Username *	Pass
3	Email	Error: *Email is compulsory*	Error message: *Invalid Email Address *	Pass
4	Password	Success	Error message *Password should be at least 6 characters *	pass
5	Register	Error: *User already Exist use different Email*	Register Successful Account Created Go back to login page	Pass
6	Login	Error: *Email or password is wrong*	Login Successful Go to games	Pass

7. Limitations and Enhancements

7.1 Limitations

- High Competition: The gaming industry is highly competitive, with many established players and new entrants. It can be challenging to stand out and attract users to a new game's website, especially with so many free alternatives available.
- Technical Complexity: Developing a games website involves creating complex algorithms and programming code. It requires highly skilled developers and can be expensive to maintain and update.
- User Retention: User retention can be a significant challenge for a game's website. It is crucial to keep the website's content fresh and exciting to retain users and prevent churn.
- Legal and Regulatory Compliance: Games websites must comply with various legal and regulatory requirements, such as data privacy and security laws, consumer protection laws, and intellectual property laws. Failure to comply with these regulations can lead to legal and financial consequences.

7.2 Future Enhancements

- Improved user experience: Enhancing the user experience can be accomplished by making the website more intuitive, visually appealing, and faster to load.
- Mobile optimization: With the increasing popularity of mobile gaming, optimizing a game website for mobile devices can help attract a wider audience.
- Social features: Adding social features such as chat rooms, leaderboards, and friend lists can enhance the sense of community and provide players with more ways to interact.
- Gamification: Incorporating gamification elements such as achievements, badges, and points systems can add a layer of fun and motivation for players.
- Innovation: Developing multiplayer games requires creativity and innovation, which can attract new players and set a game website apart from competitors.

8. Conclusion

In conclusion, The Arcade Games paradise project has several advantages and disadvantages. This project can be great opportunity to showcase your creativity and programming skills while providing entertainment to user by Creating user friendly website with a selection of high-quality games and you can attract a large audience gaming and feedback.

However, building and maintaining a successful game website can be technically challenging, resource-intensive, and competitive. Therefore, careful planning, investment, and ongoing effort are necessary to make a games website project successful.

Overall, a game website project can be a rewarding venture for those with a passion for game design and the resources and commitment to build and maintain a high-quality website.

9. Bibliography

- Google
  https://www.google.com
- YouTube
  https://www.youtube.com
- Firebase database
  https://firebase.google.com
- Kaboom js
  https://kaboomjs.com
- Stack overflow
  https://stackoverflow.com
- w3schools
  https://www.w3schools.com

10.Annexures

- Sample Code of Tic Tac Toe game

```
<!DOCTYPE html>
<html lang="en">

<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <link rel="website icon" type="png" href="/images/title_logo.png">
   <title>Tic Tac Toe</title>
</head>

<body>
   <h2>Tic Tac Toe</h2>
   <hr width="500px">
   <br>
   <div id="board"></div>
   <style>
      body {
         font-family: Arial, Helvetica, sans-serif;
         text-align: center;
      }

      #board {
         width: 450px;
         height: 450px;
         margin: 0 auto;
         display: flex;
         flex-wrap: wrap;
      }

      .tile {
         width: 147px;
         height: 147px;
         font-size: 150px;
```

```
            display: flex;
            justify-content: center;
            align-items: center;
        }

        .winner {
            background-color: lightgray;
            color: red;
        }

        .horizontal-line {
            border-bottom: 3px solid black;
        }

        .vertical-line {
            border-right: 3px solid black;
        }
    </style>
</body>
<script>

    // Show Winner
    var board;
    var playerO = "O";
    var playerX = "X";
    var currPlayer = playerO;
    let winplayer = "";
    var gameOver = false;

    window.onload = function () {
        setGame();
    }
    function setGame() {
        board = [
            [' ', ' ', ' '],
            [' ', ' ', ' '],
            [' ', ' ', ' ']
        ]
```

```
    for (let r = 0; r < 3; r++) {
        for (let c = 0; c < 3; c++) {
            //<div id="0-0;"></div>
            let tile = document.createElement("div");
            tile.id = r.toString() + "-" + c.toString();
            tile.classList.add("tile");
            if (r == 0 || r == 1) {
                tile.classList.add("horizontal-line");
            }
            if (c == 0 || c == 1) {
                tile.classList.add("vertical-line");
            }
            tile.innerText = "";
            tile.addEventListener("click", setTile);
            document.getElementById("board").appendChild(tile);
        }
    }
}
function setTile() {
    if (gameOver) {

        if (currPlayer == playerO) {
            winplayer = "X"
        } else {
            winplayer = "O"
        }

         if (confirm("----- Winner is " + winplayer + " -----" + "\n" + "Press 'OK' to Restart
game")) { window.location = ""; }
        return;
    }
    //"1-2" -> ["1", "2""]
    let coords = this.id.split("-");
    let r = parseInt(coords[0]);
    let c = parseInt(coords[1]);

    if (board[r][c] != ' ') {
        //already taken spot
        return;
```

```
    }
    board[r][c] = currPlayer; //mark the board
    this.innerText = currPlayer; //mark the board on html
    //change players
    if (currPlayer == playerO) {
        currPlayer = playerX;
    }
    else {
        currPlayer = playerO;
    }

    //check winner
    checkWinner();
}
function checkWinner() {
    //horizontally
    for (let r = 0; r < 3; r++) {
        if (board[r][0] == board[r][1] && board[r][1] == board[r][2] && board[r][0] != ' ') {
            //if we found the winning row
            //apply the winner style to that row
            for (let i = 0; i < 3; i++) {
                let tile = document.getElementById(r.toString() + "-" + i.toString());
                tile.classList.add("winner");
            }
            gameOver = true;
            return;
        }
    }
    //vertically
    for (let c = 0; c < 3; c++) {
        if (board[0][c] == board[1][c] && board[1][c] == board[2][c] && board[0][c] != ' ') {
            //if we found the winning col
            //apply the winner style to that col
            for (let i = 0; i < 3; i++) {
                let tile = document.getElementById(i.toString() + "-" + c.toString());
                tile.classList.add("winner");
            }
            gameOver = true;
            return;
```

```
            }
        }
        //diagonally
        if (board[0][0] == board[1][1] && board[1][1] == board[2][2] && board[0][0] != ' ') {
            for (let i = 0; i < 3; i++) {
                let tile = document.getElementById(i.toString() + "-" + i.toString());
                tile.classList.add("winner");
            }
            gameOver = true;
            return;
        }
        //anti-diagonally
        if (board[0][2] == board[1][1] && board[1][1] == board[2][0] && board[0][2] != ' ') {
            //0-2
            let tile = document.getElementById("0-2");
            tile.classList.add("winner");

            //1-1
            tile = document.getElementById("1-1");
            tile.classList.add("winner");

            //2-0
            tile = document.getElementById("2-0");
            tile.classList.add("winner");
            gameOver = true;
            return;
        }
    }
</script>
</html>
```