Web Game Reflection: Witchy Pairings

Team: Sejal Bansal, Varshitha Bonguluru, Carolina Liskey

Game Objective:

The objective of the Halloween-themed Matching Blocks game is to create a captivating and immersive gaming experience that seamlessly integrates the spirit of Halloween into the gameplay. By incorporating visually stunning graphics, a user-friendly interface (UI), and high-quality sound effects, the goal is to engage players of all ages in a thrilling and addictive puzzle-solving adventure. This game aims to evoke the essence of Halloween through its thematic elements while providing a seamless, intuitive, and enjoyable user experience. Through the strategic use of sound effects and a well-designed UI, the objective is to enhance player engagement, ensuring that they are not only visually delighted by the Halloween theme but also entranced by the game's audio-visual atmosphere. Ultimately, the objective is to create a game that not only entertains but also immerses players in the enchanting world of Halloween, making it a must-play during the festive season.

How to play:

1. Click on the level options given as Easy, Medium and Hard.
2. Click "Start" to begin the game.
3. Click on the blocks to reveal the image behind it. Do it to the remaining blocks and find the
   matching blocks. Every matching block will gain you points.
4. Continue this till you flip all the matching blocks.
5. You can reset the game using "Reset" button.
6. Check the time with the help of "Timer" at the top of the game.

Technology Stack:

1. JavaScript
2.HTML
3. CSS

Setup and deployment instructions

Setup Instructions for Local:

1. Clone the repository
   Git clone git@github.com:sejal-bansal/8710-project-game.git
2. Change your current directory to the game's directory.
3. Open the index.html file using a web browser to play the game locally.

Deployment:

1. Install netlify using cli
2. Enter the command netlify login on terminal
3. It will take you to the netlify account, login using your github credentials
4. Select the repository to be deployed
5. Click on Deploy Site, Netlify will then pull the latest code from the selected branch and deploy it
6. After deployment, a unique Netlify URL will be provided, such as https://witchy-pairings.netlify.app/.

7. Click on the link to access the deployed game

Credits:

1. [W3C schools](https://www.w3schools.com/)
2. [Mozilla Developer Networks](https://developer.mozilla.org/en-US/)
3. [Flaticon.com](https://www.flaticon.com/) for the all images used in the project
4. Kris DeBruine Media - [How to shuffle an array](https://www.youtube.com/watch?v=79AWYPyPEdU)
5. Adam Khoury - [Visualising the Fisher-Yates shuffle method](https://www.youtube.com/watch?v=tLxBwSL3lPQ&t=423s)
6. FreeCodeCamp.org for tutorials regarding breaking down Memory game logic
7. [Sandra Israel's Memory game process for inspiration](https://scotch.io/tutorials/how-to-build-a-memory-matching-game-in-javascript#toc-what-is-the-memory-game)
8. Flip Cards based on [W3C flip card tutorial](https://www.w3schools.com/howto/howto_css_flip_card.asp)
9. [SVG Backgrounds](https://www.svgbackgrounds.com/) for free customisable SVG Backgrounds.
10. [CSS gradient generator](https://cssgradient.io/)
11. [Game reference](https://github.com/motazabdou/MS2-MemoryGame)

Reflection:

Creating the Matching Blocks game was a challenging yet rewarding experience that provided valuable insights into the world of game design and development. Throughout the process, I encountered various challenges, discovered what worked effectively, identified areas for improvement, and gained essential lessons that significantly enhanced my skills and understanding of game development.

We managed our project using Agile methodology. Since not all team members are in the same physical location, we first established several channels of communications. Most of our meetings were dedicated to choosing the project and discussing design changes for the memory game we selected.

We used to have our stand up meetings every Monday at 6:00pm, where we discussed the progress of the game, the tasks assigned, any blocker issues, etc.

Challenges faced:

- Selecting a game which is both easy to create and manage is our first challenge.
- Balancing complexity and simplicity posed another challenge. Striking the right balance was crucial to ensure the game was easy to understand for players while offering a satisfying level of challenge.
- We all live in different places which made it impossible to meet in-person and to discuss regularly.
- Technical challenges also emerged during the development process. Implementing smooth animations, responsive touch controls, and efficient collision detection required careful coding and constant iteration.
- Compatibility issues across different devices and operating systems demanded extensive testing and optimization, adding complexity to the development process.

What worked:

- A key success factor was the iterative development approach. Starting with a basic prototype allowed for rapid experimentation and testing of various gameplay ideas. Feedback from playtesting sessions played a vital role in refining the mechanics and overall game experience. Iterative testing and feedback helped in identifying and rectifying issues early, ensuring a polished end product.
- Conducting regular zoom calls to discuss the ideas and to implement them made it easier for everyone to follow the flow. Documenting every stage of the project and creating user stories gave us clarity regarding the work which has been done.
- Additionally, establishing a clear art style and consistent visual design contributed significantly to the game's appeal. Creating visually appealing and cohesive graphics enhanced the overall player experience, making the game more engaging and immersive. The sound effects make the user feel the game while playing.

What didn't work:

- One of the notable challenges was managing scope. Initially, the project aimed for a wide range of features, but it became apparent that some elements needed to be scaled back or postponed to ensure a timely release.
- Learning to prioritize features and focus on the core gameplay mechanics would have streamlined the development process.
- Our game is compatible across Google Chrome and Safari browser only.

Lessons learned:

- The development of the Matching Blocks game taught me several valuable lessons. Firstly, the importance of simplicity cannot be overstated. A straightforward and intuitive gameplay mechanic often leads to the most enjoyable games. Complexity should arise from mastering simple rules, providing depth without overwhelming players.
- Effective planning and prioritization are essential. Establishing a clear roadmap, setting achievable milestones, and regularly reevaluating the project's scope are crucial for successful development. Moreover, incorporating player feedback early and often significantly improves the game. Players provide valuable insights that are vital for refining gameplay and addressing potential issues.
- Lastly, the significance of adaptability and resilience cannot be overlooked. Unexpected challenges will inevitably arise during the development process. Adapting to unforeseen circumstances, being open to feedback, and embracing a willingness to iterate and improve are fundamental qualities that contribute to a successful game development journey.