Part I

For this project, I created a Chinese/Mandarin language learning web game. The website would aim to teach basic Chinese words through an interactive matching game where the user matches Chinese pinyin and characters to their English equivalent. This website conveys Chinese characters and the respective pinyin about numbers, colors, and animals; it also includes its English equivalent and an image to match. Since this project idea is a game, I think it is imperative that it is interactive and engaging. It is interesting because this project is a game that allows the user to engage by matching terms together. The game also offers feedback when the user gets a match correctly or incorrectly. Finally, the game has a stopwatch and a score tracker. At the end when the user completes all the matches, there is a final screen that summarizes the user's time spent and final score. All of these features are gamification and hopefully, make the website exciting and engaging. The target audience is preschool kids that are starting to learn basic Chinese. My website is responsive for screen sizes 1340 x 890 and 1513 x 838

Part II

- Selecting a game mode
- Click on either "numbers", "colors", or "animals" card buttons on the homepage

- Clicking the help button/Getting the instructions to the game
- Click on the circle with a "?" in it that is located at the top right corner of the homepage

- Closing the instructions
- Click on "X" located at the top right corner of the "How to Play" popup on the homepage

- Matching the cards
- After clicking on a certain game, click a card with Chinese characters and click a card with the English equivalent on that game page

- Playing a new game
- After matching all the cards, click on the "New Game" button located at the bottom center of the "Congratulations" pop-up that appears

- Returning to the home page before the game ends
- Click the green back arrow located at the top left corner of the game page

Part III

- jQuery
- There was a lot of logic involved in making my game fully functional which meant I had to write a lot of JavaScript. Thus, I chose to use jQuery because it made it easier to use JavaScript. Instead of writing many lines of JavaScript, I could just call methods using jQuery using a single line. This saved me time and cleaned up my JavaScript code.
- I mainly used it to simplify my code when it came to selecting HTML elements by using '$'. I also used it for event methods such as hover(), DOM manipulation for my radio button cards, and manipulating CSS through methods such as addClass(). How you used it? (2-4 sentences max)
- One of the main things it added was my game over popup where I used addClass() to manipulate whether the pop-up was hidden or visible. It also played a crucial role in allowing me to populate my game cards easily. I used a game template and then manipulated the card DOM in order to simplify my JavaScript.

- Bootstrap
- I chose to use Bootstrap to help make my website responsive. It generally helped to save me time when it came to writing CSS code. Specifically, it had grid systems and form elements that I wanted to use.
- I used the form element radio buttons in order to create the game cards and the grid system to organize the game cards. Combining the radio buttons and grid system, I was able to easily control the fact that the player would only be able to select one Chinese vocab card and one English vocab card. These features also helped me easily make my website responsive as the grid and font in radio buttons adapt automatically to the screen size.
- Overall, it added clear organization to my website. Without Bootstrap, it would have taken significantly more CSS in order to properly create a responsive game layout and

card design. Furthermore, it would have taken a lot more thought in order to come up with a way to allow the player to only choose one Chinese vocab card and one English vocab card.

Part IV

The critiques and user testing I conducted helped me iterate through my prototypes. In my first iteration, I had the correctly matched cards flip instead of disappearing, and I got feedback that it made the game layout look cluttered. I also added more feedback for players throughout my iterations: no feedback was given immediately if a correct match was made in my initial prototype, but by my final prototype, the cards had drop shadows that would change to green whenever a correct match was made. I made some changes to the timer and score features of the game. The functionality is the same, however; I changed where the features were located on the page. Furthermore, I changed the images that were to be displayed on the English Vocabulary cards. Besides some minor aesthetic changes, the functionalities all remain the same.

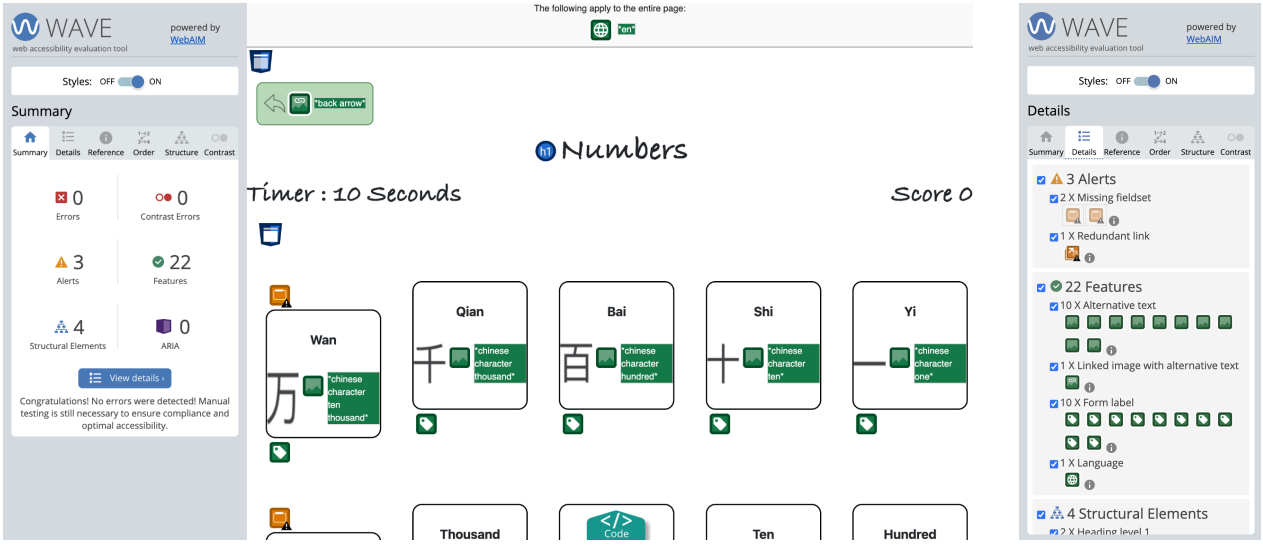
Part V

This class was my first exposure to HTML, CSS, and JavaScript, so working through the logic for JavaScript was a big challenge. Specifically, one of the main challenges I experienced came from getting the logic for checking a pair of selected cards and making them disappear if correct. Another challenge I faced was regarding alt text; I created a game template so all the game images were only listed in JavaScript, so I had to figure out how to add alt text to images using JavaScript.

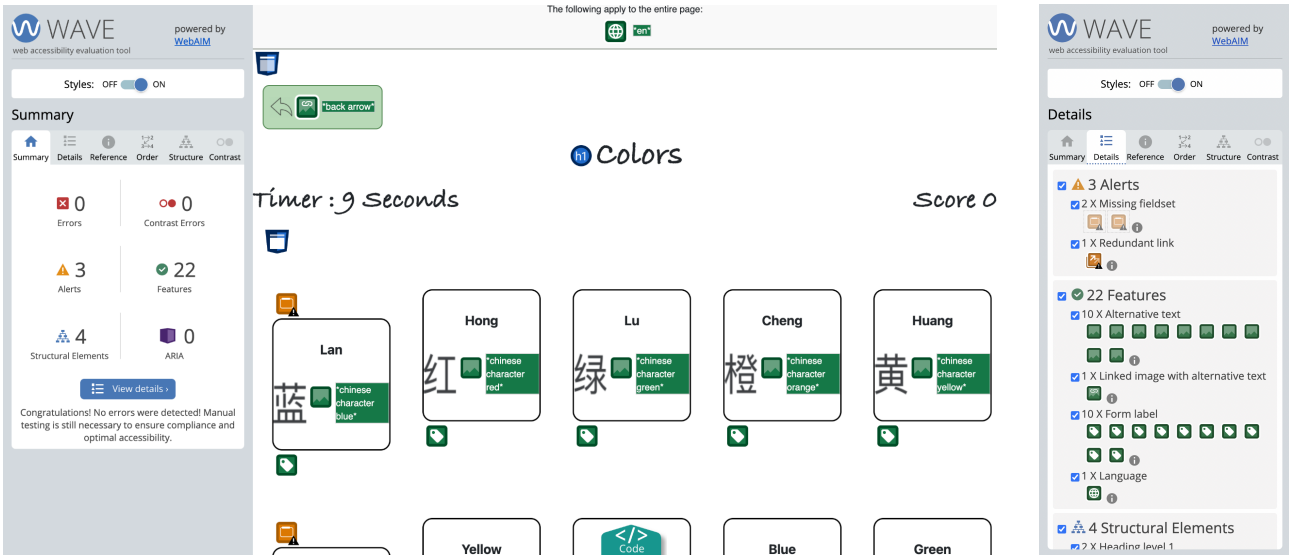
Appendix: Accessibility (WAVE) Screenshots

Homepage

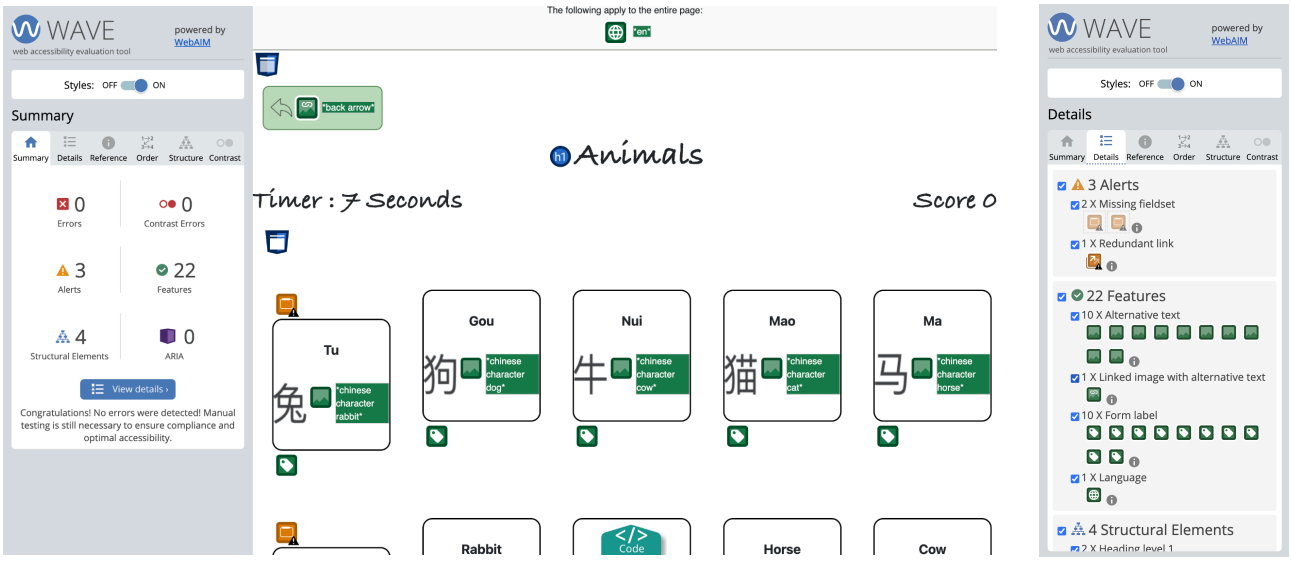
Numbers Game Page



Colors Game Page



Animals Game Page



Game Over Page/Pop Up