THE GAME OF SET

1] Basic Rules:

Set is a card game in which players find three cards with four different features.
Features include: color, symbol, number, and symbol. Each feature has three values. There are a total of 81 cards (3^4). On the table, 12 cards are dealt in a 3 by 4 layout. A group of three cards is a set if the values are either all the same or all different with the three cards. If the card meets the phrase “Two are X and one is not,” then it is not a set. If the three cards the player selects form a set, then a player is rewarded a point. If it is not a set, the player is penalized a point. The game is over when there are no more cards left in the deck to deal, or there are no more sets remaining in play. The player who gets the most points is declared the winner. If there is a tie, then all players with the highest score are declared as the winner.

Each card consists of 4 attributes:

- Color (red, green, or purple)
- Shape (diamond, squiggle, or heart)
- Shading (solid, empty, or striped)
- Number (one, two, or three)

2] Requirements:

1. Open a new terminal window and navigate 'SetGame' to the project directory.
2. Before starting the game you are required to install electron if not yet installed by running in your command line run: “npm install electron” locally, when finished run the command: “npm start” which helps to start the desktop app.
3. Start the game by running the electron.

3] Guidelines of the game

- The browser would show a popup window prompting the user to enter the names of the players with an empty string to stop. No nonsense names should be entered.
- The browser allows the user to also select if they want to play against the CPU and select the difficulty level of the CPU.
- Player selects three cards that are on the table. If he/she waits too long then depending on the CPU difficulty it automatically selects the first three cards in a certain time limit.
- If the three cards form a set, a point is rewarded with bonuses for consecutively identifying correct sets. If not, the player is penalized a point.
- When a set is identified by a player, the cards are removed and three new cards are dealt if the deck is not empty.
- If a player is stuck, they may ask for hints by clicking the ‘hint’ button.
- If there is no set on the table and the user clicks the ‘hint’ button then they are prompted to click the reshuffle button.

- The pregame lobby will have options for you to select the difficulty of the CPU, play the game and display stats.

4] How our implementation works

- card.js-This file consists of the card class with four features of the card represented by instance variables. There was a function to check whether the three cards form a set or not. It also contains a to_string function which returns the string representation of the card.

- deck.js-This file consists of the deck class which is used to make a well shuffled deck of 81 cards. This class also contains functions to draw the next card in the deck and check if a set occurs.

- game.js-This file consists of the game class which used to implement some important features to be used in the game. This class has a constructor which initializes a deck of cards, an array of players, an array of cards on the table and a variable playable that checks if the game is still playable. There is a function that deals the 12 cards, deals three new cards if a set is found, finds a set on board, a set inspector, printing the scores, a hint pusher, etc.

- player.js-This file consists of the player class which tracks the sets found and the points.

- scores.js-This file consists of the scores class which tracks the list of all time scores for the game. The class contains functions to add a new player to the list of scores, and display the list of all-time top scores.

- stats.js-This file consists of the interesting statistics in the game, such as the accuracy of the player.

- main.js- This file is the main game engine to be executed to play the game.

- index.js- This file helps to run the index.html file for the game.

- index.html- This is the html file that allows the users to play the game.

- We used the “electron” desktop app to start the web application instead of a web browser for easier access to the local files. This reduced the amount of pain in the browser development.

- To install electron, we used $ npm install electron

- Once installed successfully, we typed $ npm start on the command line to start the application.