MEMORY GAME

Introduction

The memory game created with App Inventor offers players a classic yet engaging experience of matching cards. The objective is simple: match 6 pairs of cards to emerge victorious. Unlike many other games, there is no time limit, allowing users to enjoy the gameplay at their own pace. Additionally, players have the flexibility to reset the game whenever they desire, providing them with endless opportunities to improve their skills and challenge themselves.

What sets this memory game apart is its integration of insightful data visualization tools. Alongside the gameplay, users can track their progress through dynamic bar and line charts. These charts provide valuable insights into key metrics such as the total time taken to match all pairs, the total number of clicks made, and the total number of wrong moves. By analyzing these metrics, players can not only enjoy the challenge of the game but also enhance their strategic thinking and memory skills over time.

In summary, the memory game made with App Inventor offers an enjoyable and customizable gaming experience. With its simple yet addictive gameplay, flexible reset option, and informative data visualization tools, it provides players with a fun and rewarding way to sharpen their memory and cognitive abilities.

Logic of Game

The logic of game goes like this -

1. In the start, we will lay out twelve buttons to the layout.
2. Create an array of these 12 buttons.
3. Upload 6 images that you want to show on the buttons.
4. Loop over the 12 buttons and assign those six images randomly to the buttons, provided one image should not get assigned to more than 2 buttons.
5. Set a global variable say flip1 to -1 to indicate none of the cards is flipped yet.
6. When a user clicks on a button, set flip1 to that card and show that image.
7. When the user clicks on the 2nd card, there are two possibilities, either it will be a match or not.
8. If it's a match we will increase the totalMatches which will be a variable initialized to 0 in start by one, and we will show the images.
9. If it's a mismatch, the cards will be shown for certain seconds, and then they will be turned back to the background image or initial stage.
10. Note that when the 2nd card is clicked all the other buttons should be disabled for a certain time until the timer resets, so the user can’t click on the third card while there are already two cards opened.

11. With this, the user will keep on matching the pairs until he matched all of them.
12. When all the six pairs are matched, a notifier will pop up, congratulating the user.
13. Also to mention, there is a reset button always under the cards so the user can start the game at any time he wants.
14. After the win, Bar chart will be shown which is populated by three parameters which are total time taken to win the game, total clicks made to win the game and total wrong clicks made.
15. Users can switch to Line graph as well by clicking on the Switch Chart button.
16. Then there is a play again button that will take the user back to the start screen and initialize the game again.
17. There is a reset logic that I have implemented that was very handy. I call it at the start while initializing the game, when the reset button is pressed or when the play again button is pressed. I have made it a procedure to minimize redundancy.
18. We can also reset the game by shaking the phone. I have used Accelerometer sensor for that.
19. The rest procedure will initialize all the variables and empty the lists, then layout the buttons again to the layout, randomly assigning images to the buttons. Toggling visibility of the layout components as required.
20. Blocks of the app are self explanatory as well, if we'll go through them, one gets a better idea of how things are working.

Code Review and testing -

My app's code review and testing was done by my classmate Suhail Thechikodan. He went through the blocks to understand logic and was easily able to grasp it. Testing was done by me and him, where he suggested adding a line graph as well with a Bar graph to represent result details. Further some layout changes were made after some discussion. Overall at the time of testing there was not any major bug found, but some minor modifications that were done.

Future Scope -

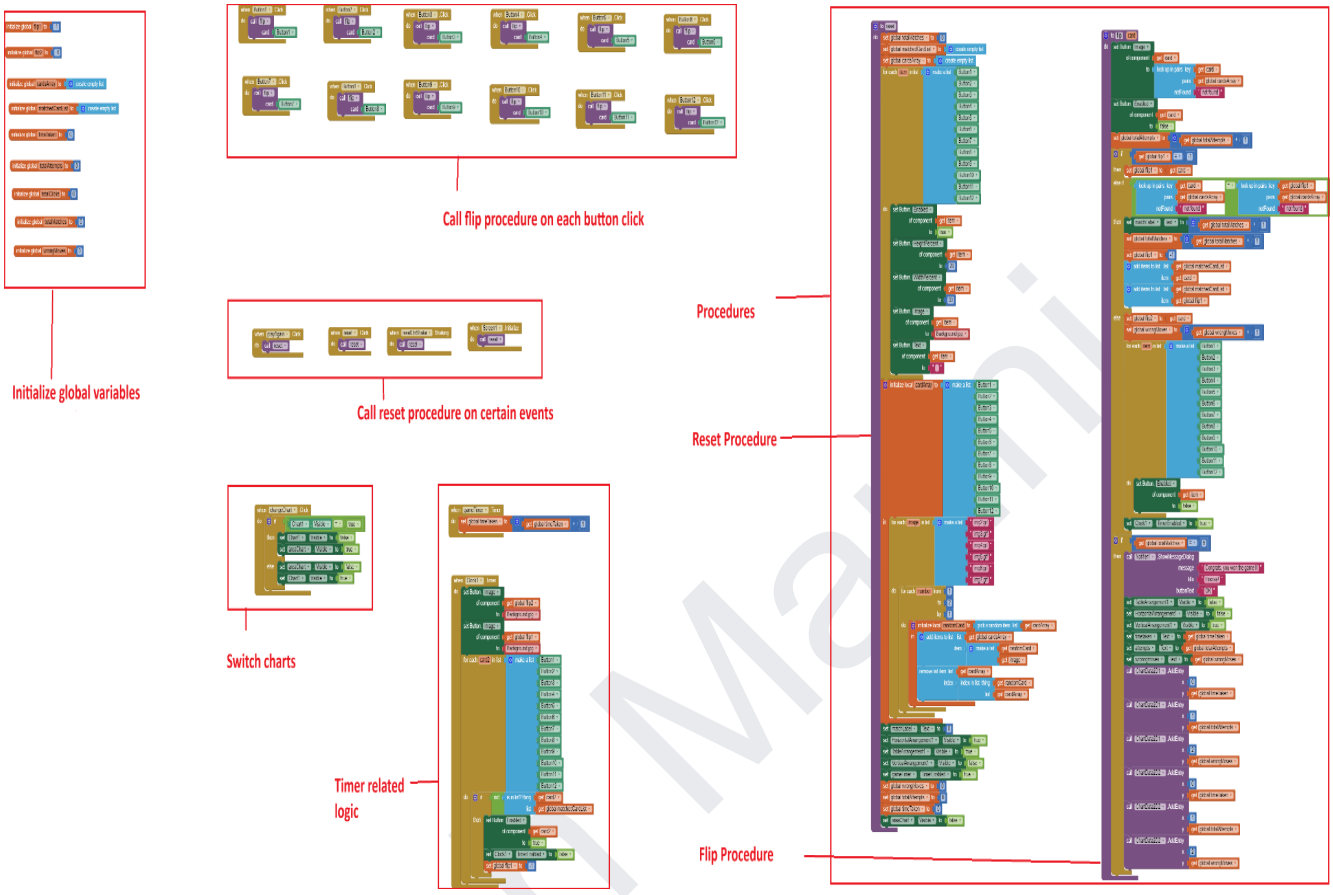
There can be plenty of features that can be added to this app such as -

1. Add a database(tiny db) to store the user's information.
2. Add leaderboard to display user's position worldwide.
3. Add difficulty levels (easy, moderate, hard) with an increasing number of cards as the difficulty increases.
4. Choose images themes, such that cards can be nature images, cartoon images or anything that we can add in game.

Moreover, there can be other features added to make the game more advanced as we want. We can always use chat gpt to suggest features or can see the advanced version of the game on the internet.

Blocks Organization -

Image below might not be clear, but we can download it to view properly and see the block's structure. I have also added the image with my submission and it's available on my github as well here - https://github.com/nic-dgl104-winter-2024/Yogesh_Manni/blob/main/blocks.png



There should be some feature or component to label our blocks or group them with some highlights as done in the image above, that will really come in handy when there are thousands of blocks. Maybe there is something that I'm not aware of, if something is there please let me know, it will be helpful in future development.

Resources used -

1. Chat gpt - Some help was taken from chat gpt in terms of logic building.

2. I have made a similar web app as a personal project long before, so the idea was referenced from there.
3. Hints were taken from this website - https://www.brainzilla.com/fun/memory-game/