**Iteration 2#**

## Requirements Statement

1. Create a Single Page Application (SPA) using Vuejs and JavaScript ES2018.
2. Write a program to play a number guessing game. The program shall generate a random number between 0 and 99. The USER inputs his/her guess, and the program shall response with "COLD" if the guess is more than 40 from the target number, "COOL" if the guess is within 20-39 of the target number, “WARM” if the guess is within 10-19 of the target number, “HOT” if the guess is within 1-9 of the target number or "You got it in n trials" if the guess is correct.

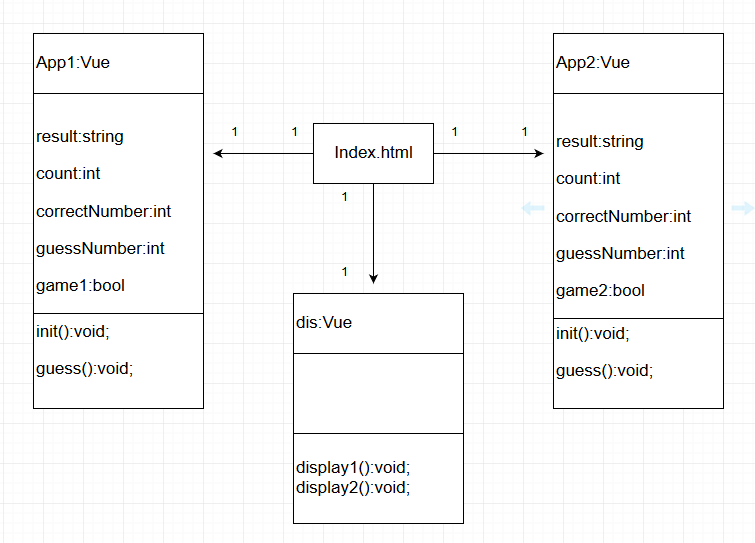
## Iteration Plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Milestone | Planned Start Date | Actual Start Date | Planned End Date | Actual End Date | Task |
| Iteration |  |  |  |  |  |
| Planning | 19/8/2019 | 19/8/2019 | 19/8/2019 | 19/8/2019 | 1.Write a programming, which can play Number Guess Game with computer.  2.Finish the second part of the game, let the computer generate a random number between 0 and 99,User guess the number, and computer shall response with "COLD", "COOL" , "WARM" ,"HOT" or "You got it in n trials" if the guess is correct.  3.Layout four games in one page. Use four buttons to choose the game. |
| Analysis | 19/8/2019 | 19/8/2019 | 19/8/2019 | 19/8/2019 | Analyze project requirements. List the required functionality for the project. |
| Design | 20/8/2019 | 20/8/2019 | 20/8/2019 | 20/8/2019 | 1.Design the Single Page Application’s website Layout.  2.Use UML dynamic diagram for the programming. |
| Coding | 21/8/2019 | 21/8/2019 | 21/8/2019 | 21/8/2019 | Coding for the second literation. |
| Testing | 23/8/2019 | 23/8/2019 | 23/8/2019 | 23/8/2019 | Make a testing plan and do the test.  Record the result of the test. |

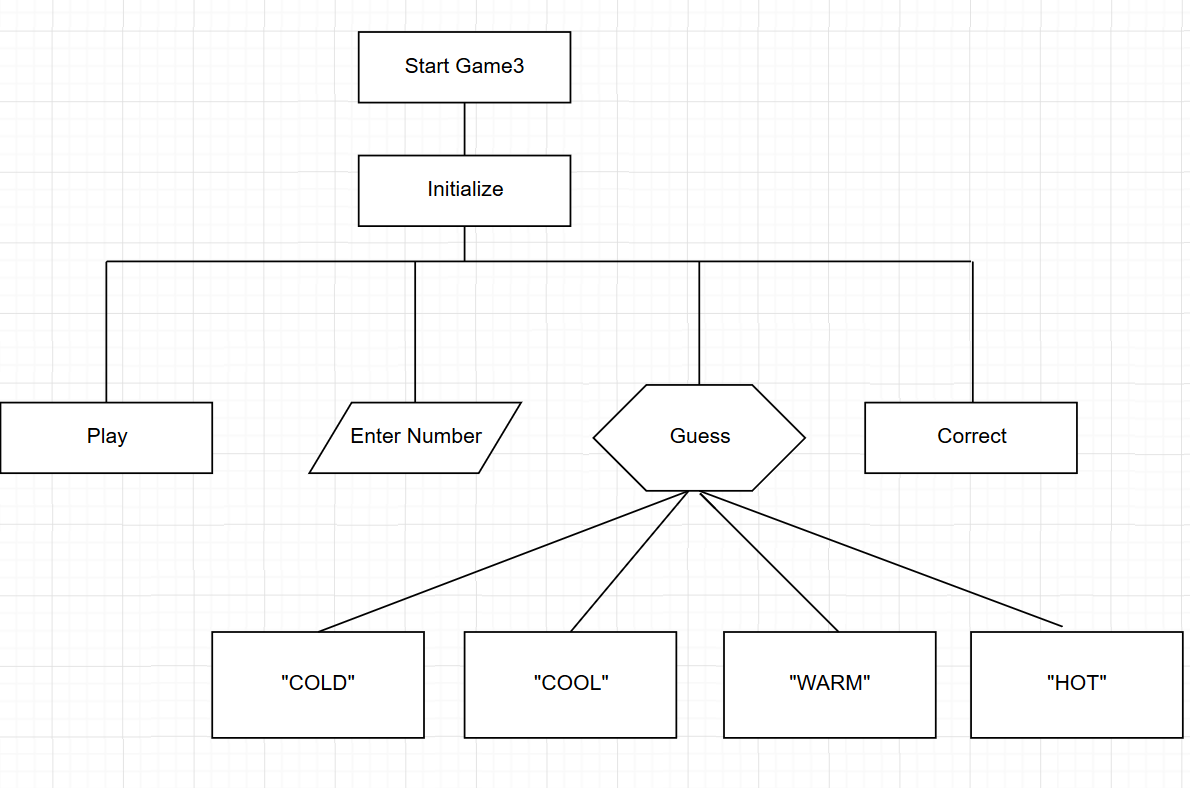
## Task Summary

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Time in Phase (hr.)** | **Plan** | **Actual** | **Start** | **Stop** | **To Date** | **To Date %** |
| Planning | 0.5hr | 0.5hr | 21:00 | 21:30 | 19/8/2019 | 10% |
| Analysis | 0.5hr | 0.5hr | 22:00 | 22:30 | 19/8/2019 | 10% |
| Design | 0.5hr | 0.5hr | 10:00 | 10:30 | 20/8/2019 | 10% |
| Coding | 1.5hr | 1.5hr | 12:00 | 13:30 | 21/8/2019 | 30% |
| Test | 1.5hr | 1.5hr | 22:00 | 23:30 | 23/8/2019 | 30% |
| Postmortem | 0.5hr | 0.5hr | 18:00 | 18:30 | 24/8/2019 | 10% |
| Total | 5hr | 5hr |  |  |  | 100% |

## Class Diagram



## Plan for how feature works



## Test Plans:

|  |  |  |
| --- | --- | --- |
|  | Test Plans | Expected Results |
| 1 | Test the function of button “Game One”, “Game Two”. | When the user clicks the button “Game One”, game one appears on the webpage, game two still hide. When the user clicks the button “Game Two”, game one disappears and game two appears on the webpage. |
| 2 | Test the function of button “Play”. | When the user clicks the button “play”, the game counter clears to zero, and the computer automatically generates an integer of 0-99, and the web page appears a string "Please guess a number, enter it, and press Guess." |
| 3 | Test the “Guess” function. | Enter test guess number, the prompt says with "COLD”, “COOL" , "WARM" ,"HOT" or "You got it in n trials" |
| 4 | Test the counter. | Play the game and record the number of guesses. Compare with the final output. |

## Test Report Template

|  |  |
| --- | --- |
| Test Name/Number | 1 |
| Test Objective | Test the function of button “Game One”, “Game Two”. |
| Test Description | Click button “Game One”, game one appears on the webpage, game two still hide. When the user clicks the button “Game Two”, game one disappears and game two appears on the webpage. |
| Test Conditions | Two buttons to choose. |
| Expected Results | Click button “Game One”, game one appears on the webpage, game two still hide. When the user clicks the button “Game Two”, game one disappears and game two appears on the webpage. |
| Actual Results |  |
| Test Name/Number | 2 |
| Test Objective | Test the function of button “Play”. |
| Test Description | Execute NumberGuessGameV3.html on browser. Enter F12, press button “play”, input vb. count; |
|  | Execute NumberGuessGameV3.html on browser. Enter F12, press button “play”, input vb. correctNumber; |
|  | Execute NumberGuessGameV3.html on browser. |
| Test Conditions | No options are given, default options apply. |
| Expected Results | vb. Count is 0; |
|  | vb. correctNumbe’s value in the range of 0 to 99; |
|  | "Please guess a number, enter it, and press Guess." appear. |
| Actual Results |  |
| Test Name/Number | 3 |
| Test Objective | Test the “Guess” function. |
| Test Description | 1. Execute NumberGuessGameV3.html on browser. Play the game and enter number 99. Use develop tools to find out the correctNumbe, enter testing number which is within 1-9 of the target number. |
|  | 2. Enter testing number which is within 10-19 of the target number. |
|  | 3. Enter testing number which is within 20-39 of the target number. |
|  | 4. Enter testing number which is more than 40 of the target number. |
|  | 5. Enter testing number which is equal to the target number. |
| Test Conditions | No options are given, default options apply. |
| Expected Results | 1. Output “HOT”. |
|  | 2. Output “WARM”. |
|  | 3. Output “COOL”. |
|  | 4. Output “COLD”. |
|  | 5. Output “You got it in n trials”. |
| Actual Results |  |
|  |  |
|  |  |
|  |  |
|  |  |
| Test Name/Number | 4 |
| Test Objective | Test the counter. |
| Test Description | Play the game and record the number of guesses. Compare with the final output. |
| Test Conditions | No options are given, default options apply. |
| Expected Results | The record number is equal to final output. |
| Actual Results |  |

## Error Logs

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date** | **No.** | **Type** | **Inject** | **Remove** | **Fix Time(10min/block)** | **Fix Ref** | **Description** |
| **21/8/2019** | 1 | 20 | Code | Compile | 1 | trouble shooting | Game two cannot display on html page. Wrong id name. Fixed. |
| **21/8/2019** | 2 | 20 | code | Compile | 1 | trouble shooting | Add four buttons, but the buttons cannot display. Wrong html code of button type. Fixed. |
| **21/8/2019** | 3 | 20 | code | Compile | 1 | trouble shooting | Add four buttons, but the two games cannot be hide. Add two new Vue methods call “display”. Fixed. |
| **23/8/2019** | 4 | 20 | code | Compile | 1 | trouble shooting | Try to separate the two games JS code from HTML code. Created two separate JS documents but run in conflict with each other., Wrong position of JS insert statement. Fixed. |
| **23/8/2019** | 5 | 20 | code | Compile | 1 | trouble shooting | Output always be “COOL”, Error syntax(20<=abs<=39). Fixed. |
| **23/8/2019** | 6 | 20 | code | Compile | 1 | trouble shooting | Output always be “WARM”, Error syntax(10<=abs<=19). Fixed. |

## PSP Process Improvement Proposal (PIP)

|  |
| --- |
| Problem Description |
| Briefly describe the problems that you encountered. |
| Game two cannot display on html page. |
| Try to separate the html code and JS code and each game has its separate JS document. When run the program the two JS document in conflict with each other. |
| Try to add four buttons to control the four games but the buttons 1. At first the buttons cannot appear 2. Then cannot work. |
| Error Syntax. |
| Proposal Description |
| Briefly describe the process improvements that you propose. |
| Trouble shooting, fixed the ID name of the second game. |
| Trouble shooting, change the position of script. |
| Trouble shooting, fixed the type of button, add a new Vue method. |
| Fixed the wrong syntax. |
| Other Notes and Comments |
| Note any other comments or observations that describe your experiences or improvement ideas. |
| 1. When the four games are complete, all the JS code should be merged into one JS document. |
| 1. The game interface should be improved. |
| 1. Game functions are improved according to user experience. |