

WEB_ICP3

DHARMA TEJA K

Email: dkbmy@umsystem.edu

GitHub Link: <https://github.com/Dharmateja183/Web-and-Mobile-programming-spring-2022/tree/main/Webpart/ICP3>

Avinash Reddy T

Email: atfkh@umsystem.edu

GitHub link: <https://github.com/avinashreddy3/WebDevCourse/tree/main/WebPart/ICP3>

1. Rock, Paper, Scissors Task

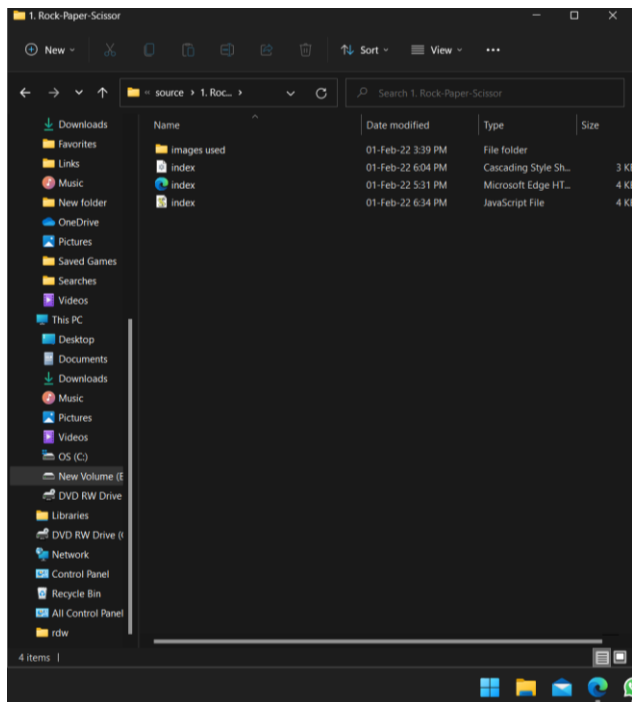
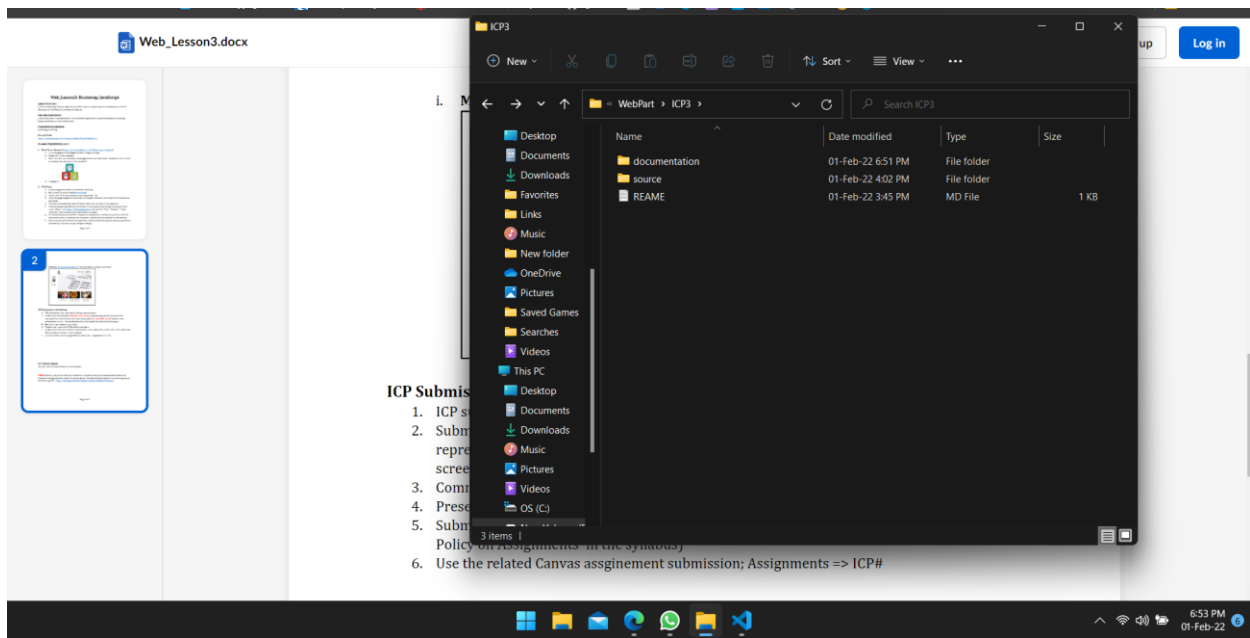
The main aim here is to create Rock, Paper, and Scissors game webpage using JavaScript.

JavaScript is a dynamic programming language that is used in web development and in developing web applications as well. It allows us to run some features dynamically on websites/webpages which is not possible to do with only CSS and HTML.

And we also use HTML, CSS coding as well to complete this task.

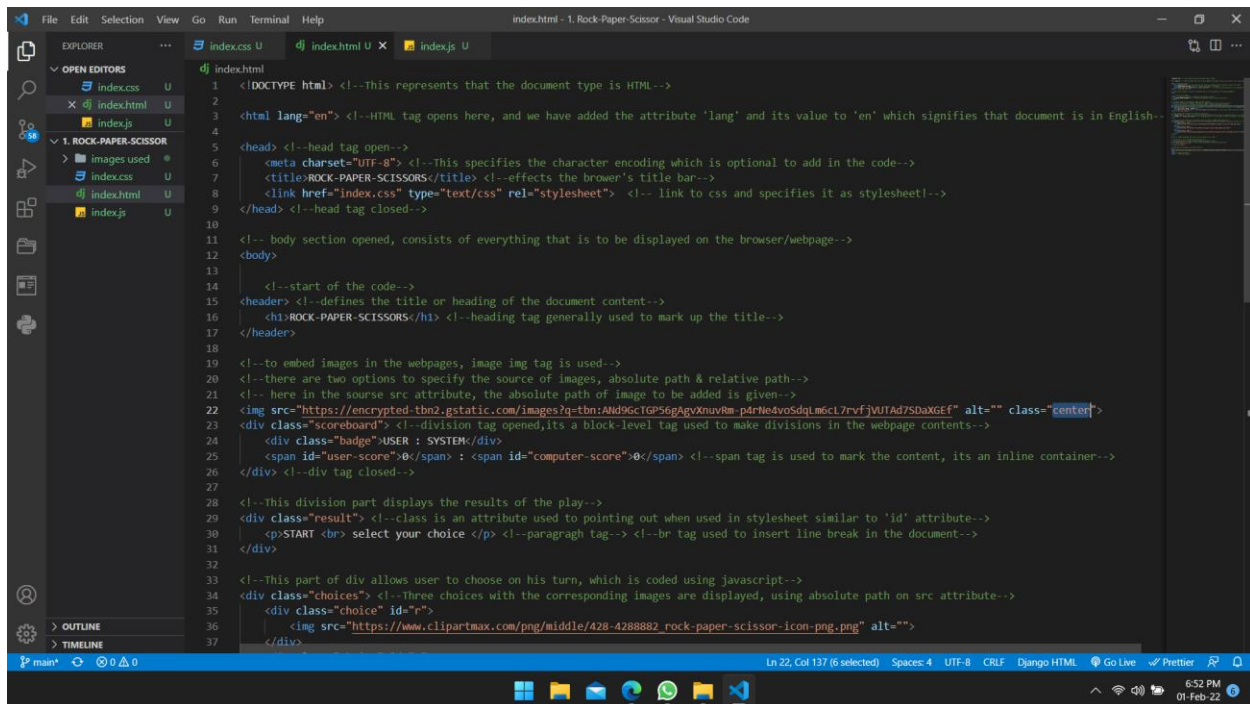
First, we create two folders named

- 1) Documentation (code snaps are stored)
- 2) Source (the actual code files are stored)

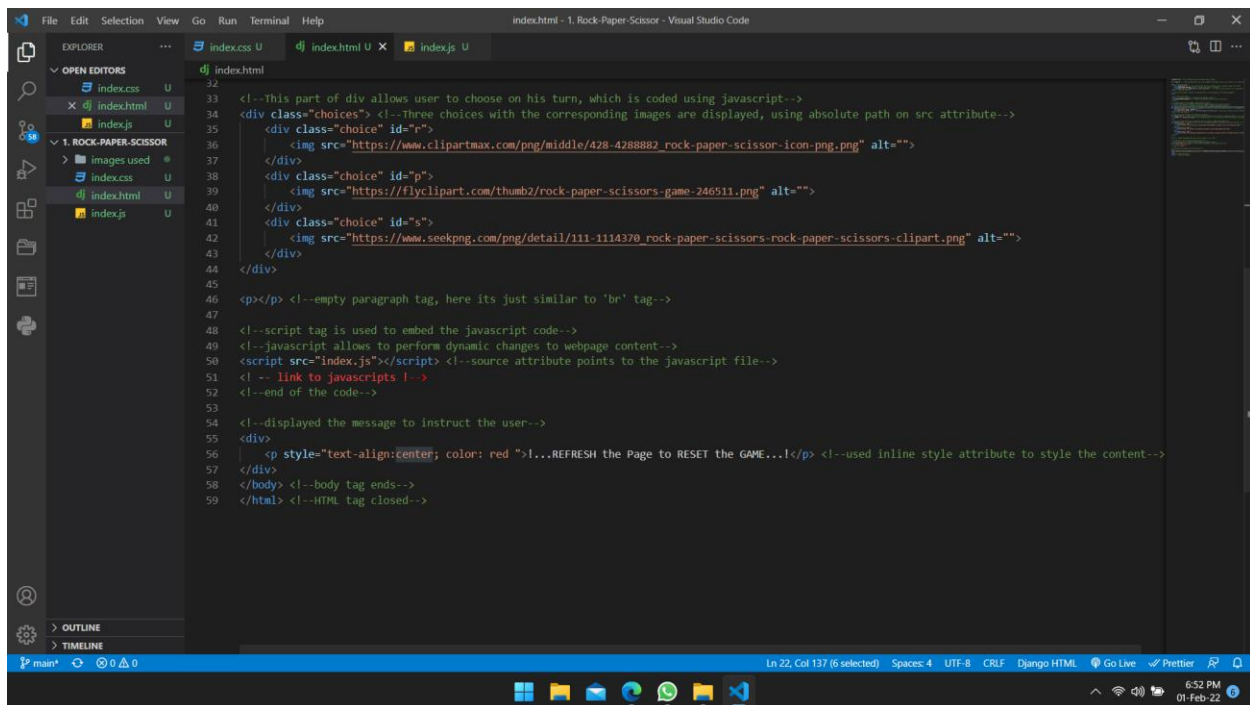


HTML code

Below are the snaps of HTML code we developed and also commented well to understand.



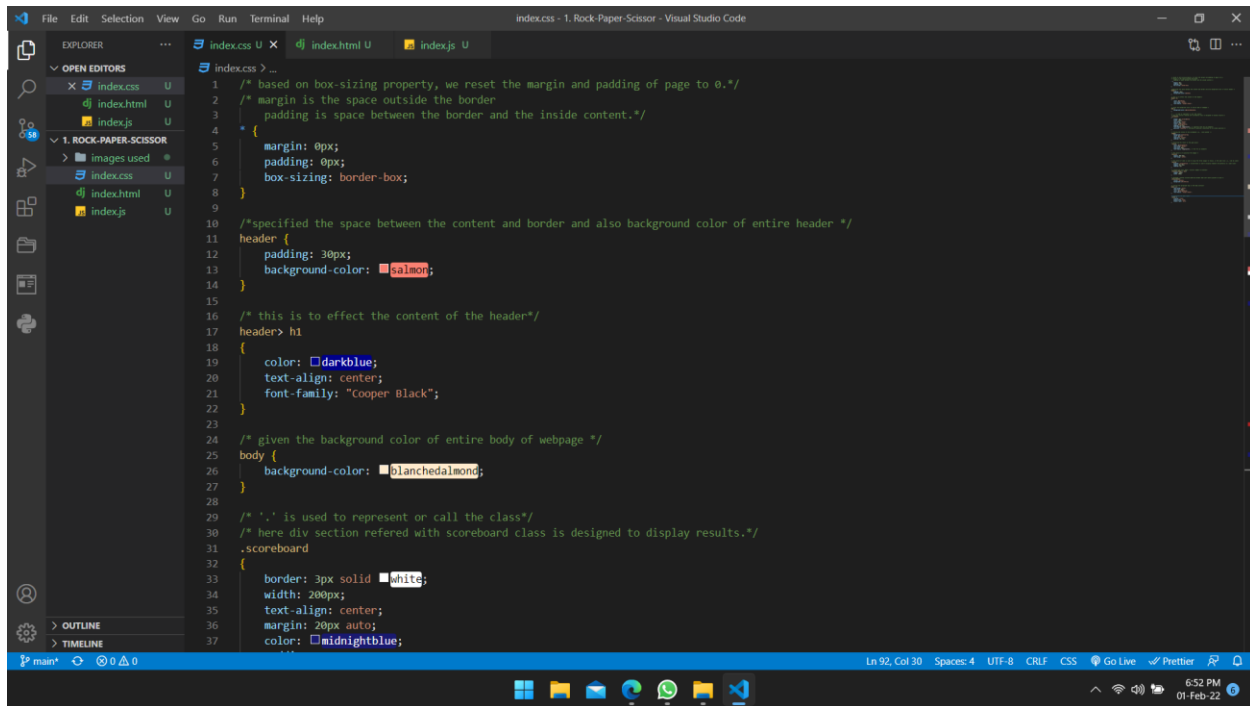
```
1 <!DOCTYPE html> <!--This represents that the document type is HTML-->
2
3 <html lang="en"> <!--HTML tag opens here, and we have added the attribute 'lang' and its value to 'en' which signifies that document is in English-->
4
5 <head> <!--head tag open-->
6   <meta charset="UTF-8"> <!--This specifies the character encoding which is optional to add in the code-->
7   <title>ROCK-PAPER-SCISSORS</title> <!--effects the browser's title bar-->
8   <link href="index.css" type="text/css" rel="stylesheet"> <!-- link to css and specifies it as stylesheet!-->
9 </head> <!--head tag closed-->
10
11 <!-- body section opened, consists of everything that is to be displayed on the browser/webpage-->
12 <body>
13
14   <!--start of the code-->
15   <header> <!--defines the title or heading of the document content-->
16     <h1>ROCK-PAPER-SCISSORS</h1> <!--heading tag generally used to mark up the title-->
17   </header>
18
19   <!--to embed images in the webpages, image img tag is used-->
20   <!--there are two options to specify the source of images, absolute path & relative path-->
21   <!-- here in the source src attribute, the absolute path of image to be added is given-->
22   
23   <div class="scoreboard"> <!--division tag opened, its a block-level tag used to make divisions in the webpage contents-->
24     <div class="badge">USER : SYSTEM</div>
25     <span id="user-score">0</span> : <span id="computer-score">0</span> <!--span tag is used to mark the content, its an inline container-->
26   </div> <!--div tag closed-->
27
28   <!--This division part displays the results of the play-->
29   <div class="result"> <!--class is an attribute used to pointing out when used in stylesheet similar to 'id' attribute-->
30     <p>START <br> select your choice </p> <!--paragraph tag--> <!--br tag used to insert line break in the document-->
31   </div>
32
33   <!--This part of div allows user to choose on his turn, which is coded using javascript-->
34   <div class="choices"> <!--Three choices with the corresponding images are displayed, using absolute path on src attribute-->
35     <div class="choice" id="r">
36       
37     </div>
```



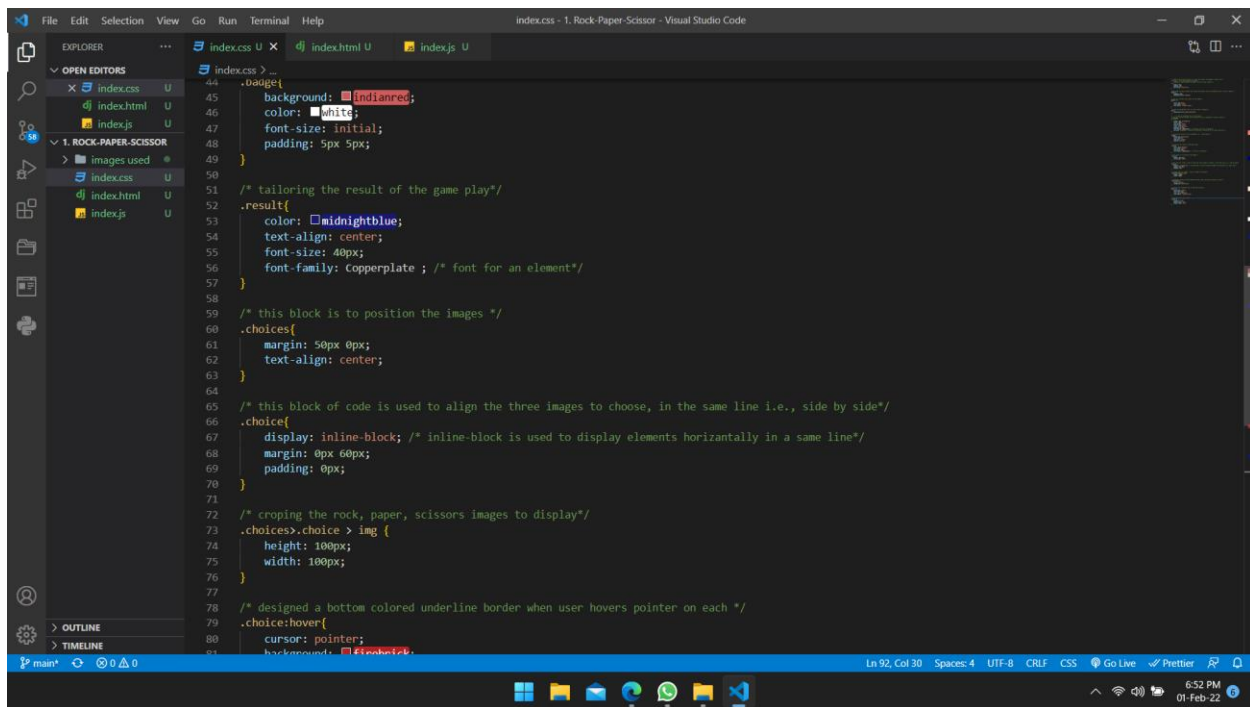
```
38   <div class="choice" id="p">
39     
40   </div>
41   <div class="choice" id="s">
42     
43   </div>
44 </div>
45
46 <p></p> <!--empty paragraph tag, here its just similar to 'br' tag-->
47
48 <!--script tag is used to embed the javascript code-->
49 <!--javascript allows to perform dynamic changes to webpage content-->
50 <script src="index.js"></script> <!--source attribute points to the javascript file-->
51 <!-- link to javascripts !-->
52 <!--end of the code-->
53
54 <!--displayed the message to instruct the user-->
55 <div>
56   <p style="text-align:center; color: red ">!...REFRESH the Page to RESET the GAME...!</p> <!--used inline style attribute to style the content-->
57 </div>
58 </body> <!--body tag ends-->
59 </html> <!--HTML tag closed-->
```

CSS code

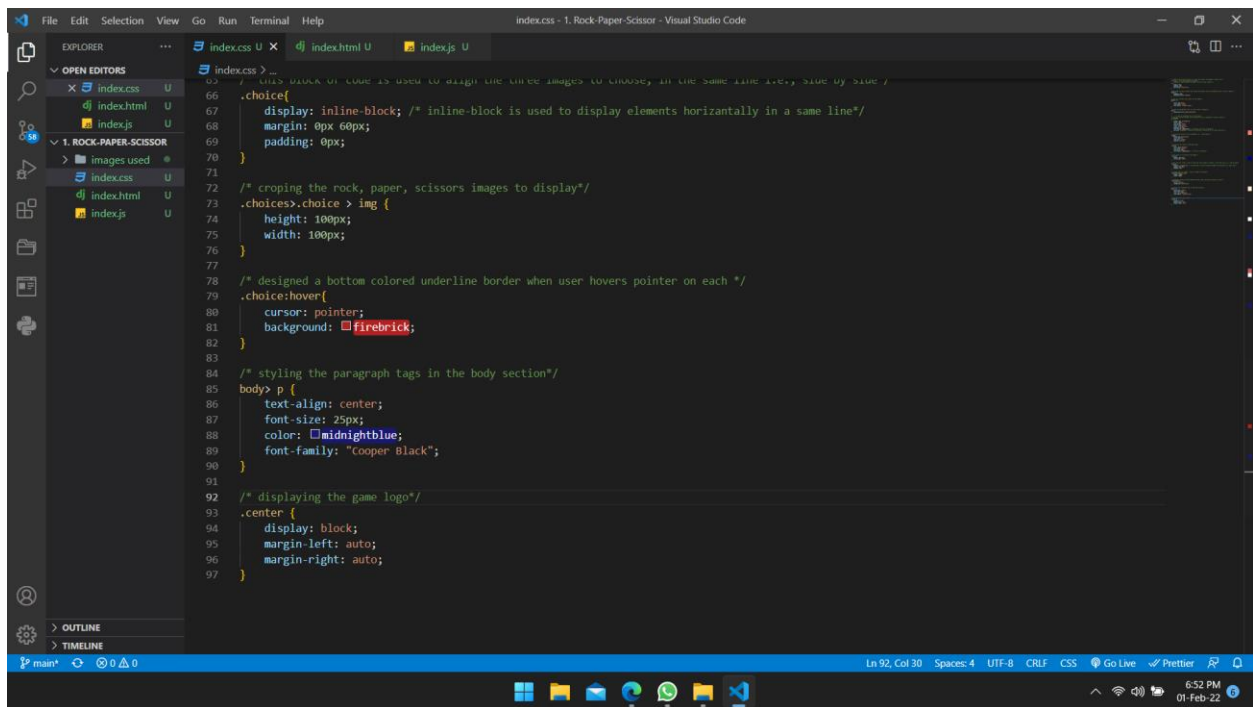
Below are the screenshots of CSS code we developed and well commented as well for understanding.



```
1  /* based on box-sizing property, we reset the margin and padding of page to 0.*/
2  /* margin is the space outside the border
3  /* padding is space between the border and the inside content.*/
4
5  {
6      margin: 0px;
7      padding: 0px;
8      box-sizing: border-box;
9  }
10
11 /*specified the space between the content and border and also background color of entire header */
12 header {
13     padding: 30px;
14     background-color: #salmon;
15 }
16
17 /* this is to effect the content of the header*/
18 header> h1
19 {
20     color: #darkblue;
21     text-align: center;
22     font-family: "Cooper Black";
23 }
24
25 /* given the background color of entire body of webpage */
26 body {
27     background-color: #blanchedalmond;
28 }
29
30 /* '.' is used to represent or call the class*/
31 /* here div section referred with scoreboard class is designed to display results.*/
32 .scoreboard
33 {
34     border: 3px solid #white;
35     width: 200px;
36     text-align: center;
37     margin: 20px auto;
38     color: #midnightblue;
```

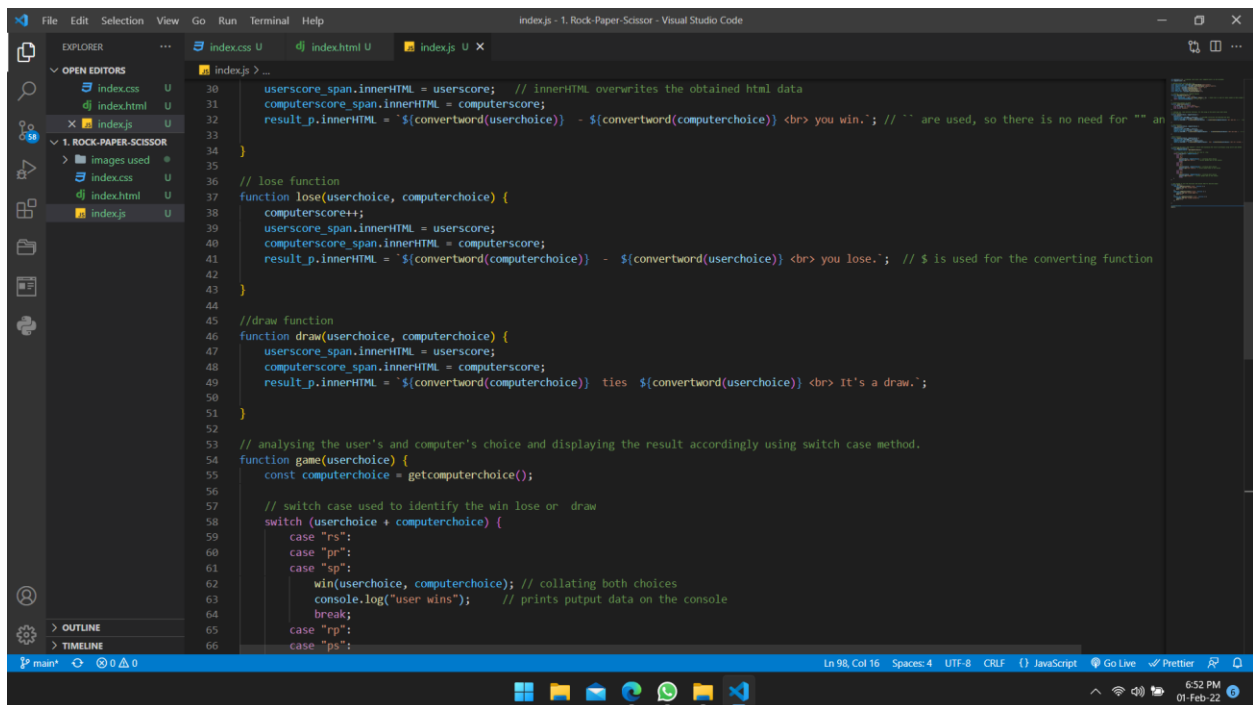


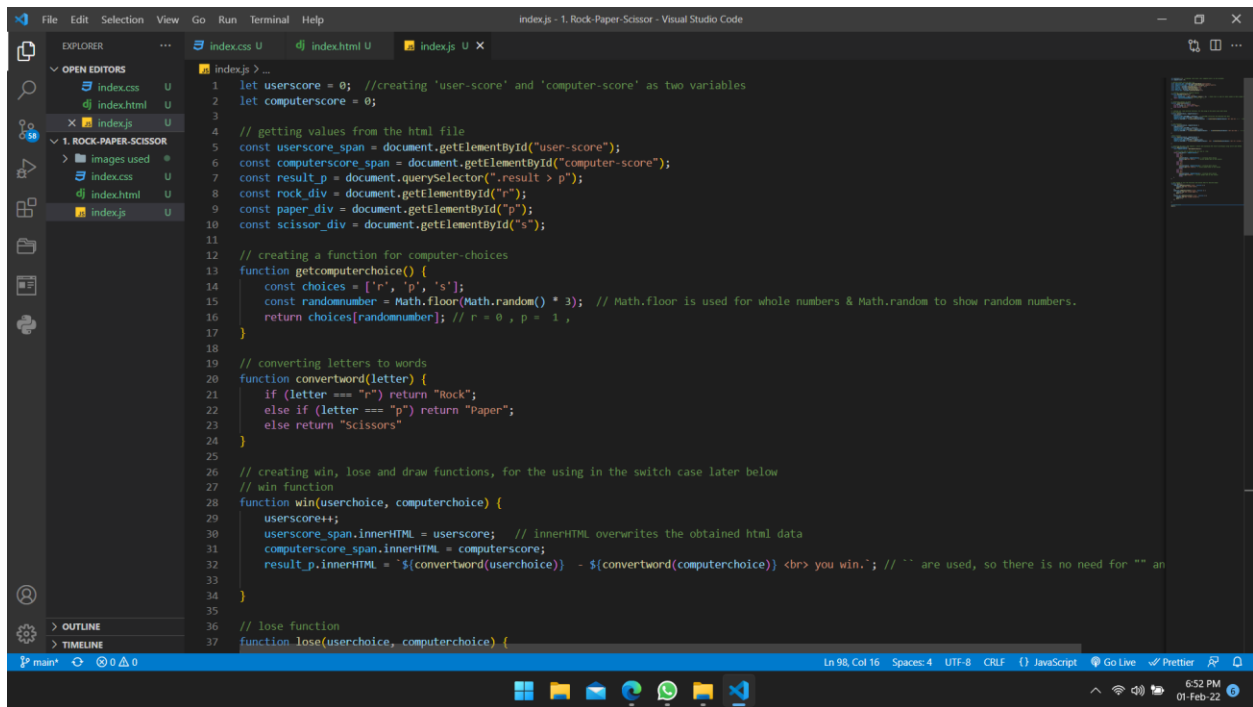
```
44 .budget
45 {
46     background: #indianred;
47     color: #white;
48     font-size: initial;
49     padding: 5px 5px;
50 }
51
52 /* tailoring the result of the game play*/
53 .result{
54     color: #midnightblue;
55     text-align: center;
56     font-size: 40px;
57     font-family: Copperplate ; /* font for an element*/
58 }
59
60 /* this block is to position the images */
61 .choices{
62     margin: 50px 0px;
63     text-align: center;
64 }
65
66 /* this block of code is used to align the three images to choose, in the same line i.e., side by side*/
67 .choice{
68     display: inline-block; /* inline-block is used to display elements horizontally in a same line*/
69     margin: 0px 60px;
70     padding: 0px;
71 }
72
73 /* cropping the rock, paper, scissors images to display*/
74 .choices>.choice>img {
75     height: 100px;
76     width: 100px;
77 }
78
79 /* designed a bottom colored underline border when user hovers pointer on each */
80 .choice:hover{
81     cursor: pointer;
82     background: #firebrick;
```



JAVASCRIPT code

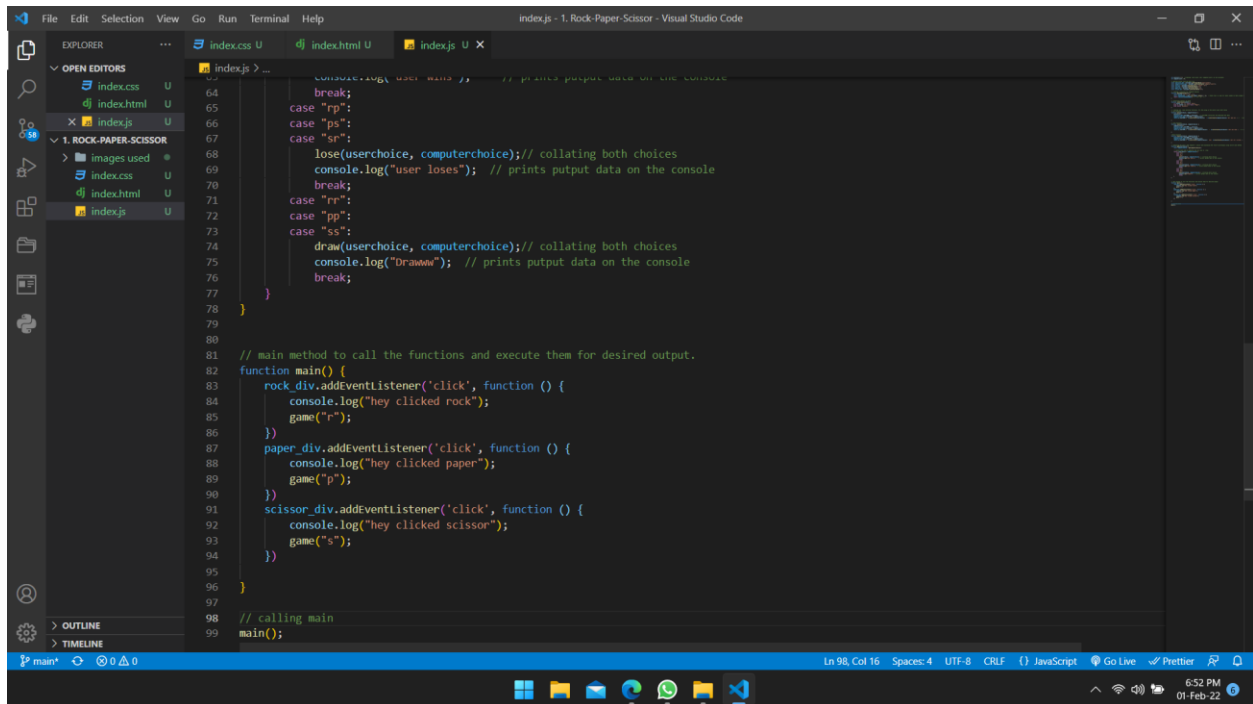
Below are the snippets for JS code we have written and shared the explanation in the form of comments where required, which helps in understanding the functions, methods used.





```
1 let userscore = 0; //creating 'user-score' and 'computer-score' as two variables
2 let computerscore = 0;
3
4 // getting values from the html file
5 const userscore_span = document.getElementById("user-score");
6 const computerscore_span = document.getElementById("computer-score");
7 const result_p = document.querySelector(".result > p");
8 const rock_div = document.getElementById("r");
9 const paper_div = document.getElementById("p");
10 const scissor_div = document.getElementById("s");
11
12 // creating a function for computer-choices
13 function getcomputerchoice() {
14     const choices = ['r', 'p', 's'];
15     const randomnumber = Math.floor(Math.random() * 3); // Math.floor is used for whole numbers & Math.random to show random numbers.
16     return choices[randomnumber]; // r = 0, p = 1, s = 2
17 }
18
19 // converting letters to words
20 function convertword(letter) {
21     if (letter === "r") return "Rock";
22     else if (letter === "p") return "Paper";
23     else return "Scissors"
24 }
25
26 // creating win, lose and draw functions, for the using in the switch case later below
27 // win function
28 function win(userchoice, computerchoice) {
29     userscore++;
30     userscore_span.innerHTML = userscore; // innerHTML overwrites the obtained html data
31     computerscore_span.innerHTML = computerscore;
32     result_p.innerHTML = `${convertword(userchoice)} - ${convertword(computerchoice)} <br> you win.`; // `` are used, so there is no need for "" an
33 }
34
35 // lose function
36 function lose(userchoice, computerchoice) {
37     computerscore++;
38     computerscore_span.innerHTML = computerscore;
39     result_p.innerHTML = `${convertword(userchoice)} - ${convertword(computerchoice)} <br> you lose.`;
40 }
41
42 // draw function
43 function draw(userchoice, computerchoice) {
44     result_p.innerHTML = `${convertword(userchoice)} - ${convertword(computerchoice)} <br> Draw!`;
45 }
46
47 // main method to call the functions and execute them for desired output.
48 function main() {
49     rock_div.addEventListener('click', function () {
50         console.log("hey clicked rock");
51         game("r");
52     });
53     paper_div.addEventListener('click', function () {
54         console.log("hey clicked paper");
55         game("p");
56     });
57     scissor_div.addEventListener('click', function () {
58         console.log("hey clicked scissor");
59         game("s");
60     });
61 }
62
63 // calling main
64 main();
```

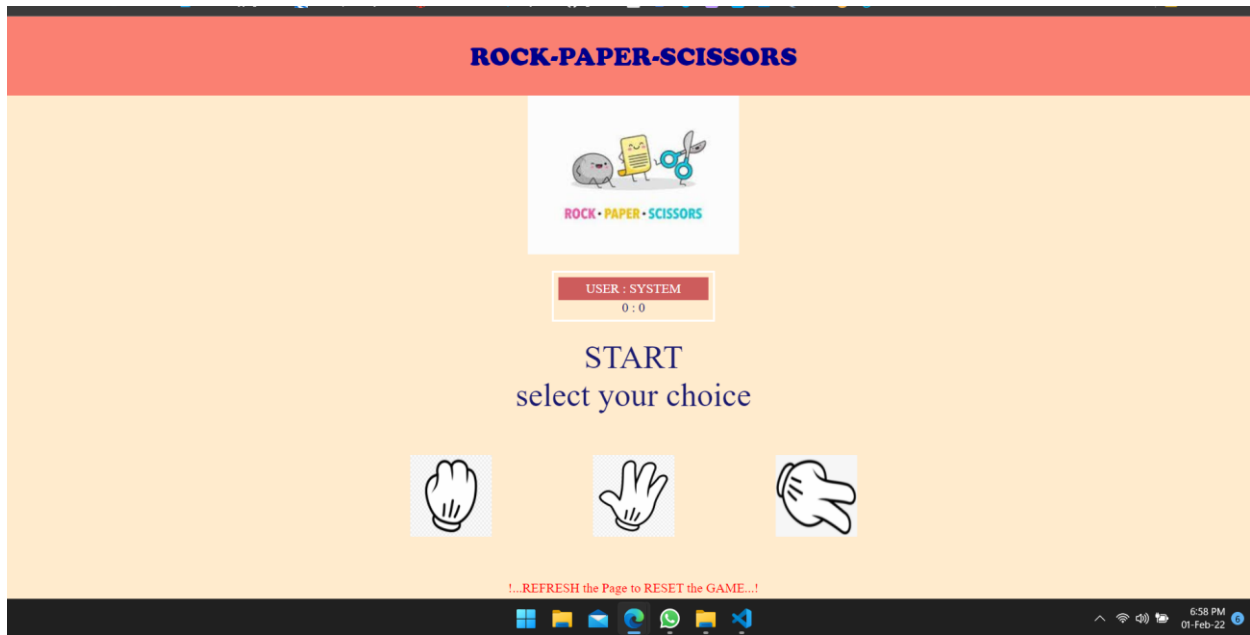
We declare the main (), and we call it for execution.



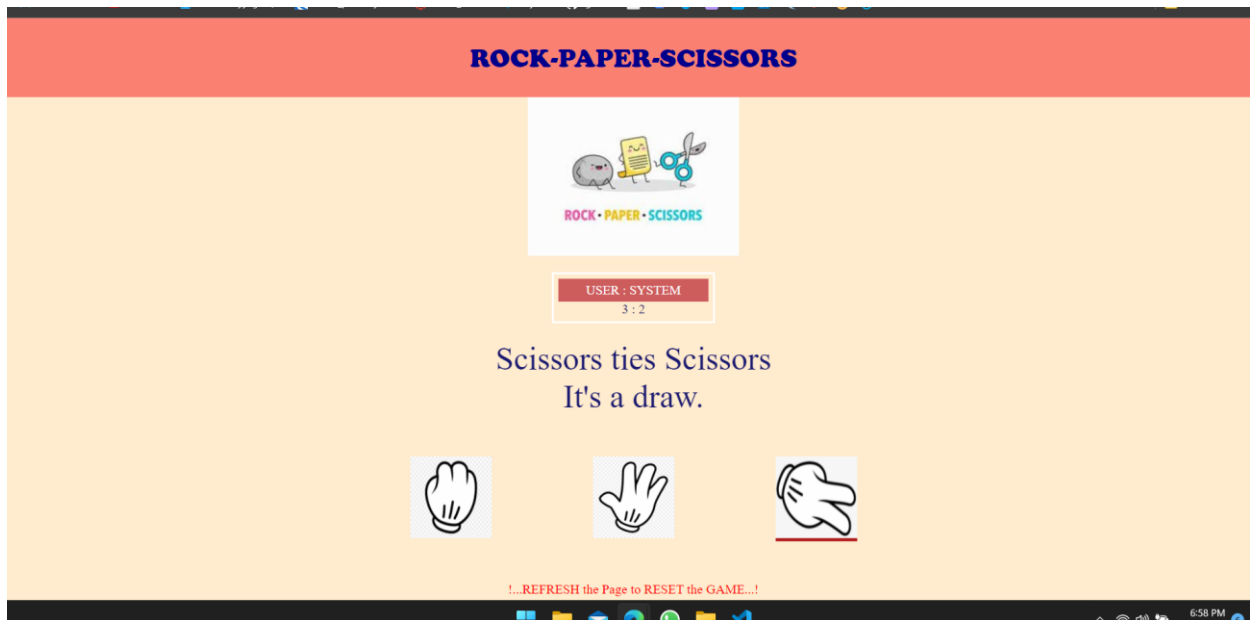
```
64 console.log("hey clicked rock"); // prints output data on the console
65 break;
66 case "rp":
67 case "ps":
68 case "sr":
69     lose(userchoice, computerchoice); // collating both choices
70     console.log("user loses"); // prints output data on the console
71     break;
72 case "rr":
73 case "pp":
74 case "ss":
75     draw(userchoice, computerchoice); // collating both choices
76     console.log("Drawww"); // prints output data on the console
77     break;
78 }
79 }
80
81 // main method to call the functions and execute them for desired output.
82 function main() {
83     rock_div.addEventListener('click', function () {
84         console.log("hey clicked rock");
85         game("r");
86     });
87     paper_div.addEventListener('click', function () {
88         console.log("hey clicked paper");
89         game("p");
90     });
91     scissor_div.addEventListener('click', function () {
92         console.log("hey clicked scissor");
93         game("s");
94     });
95 }
96
97 // calling main
98 main();
```

OUTPUT

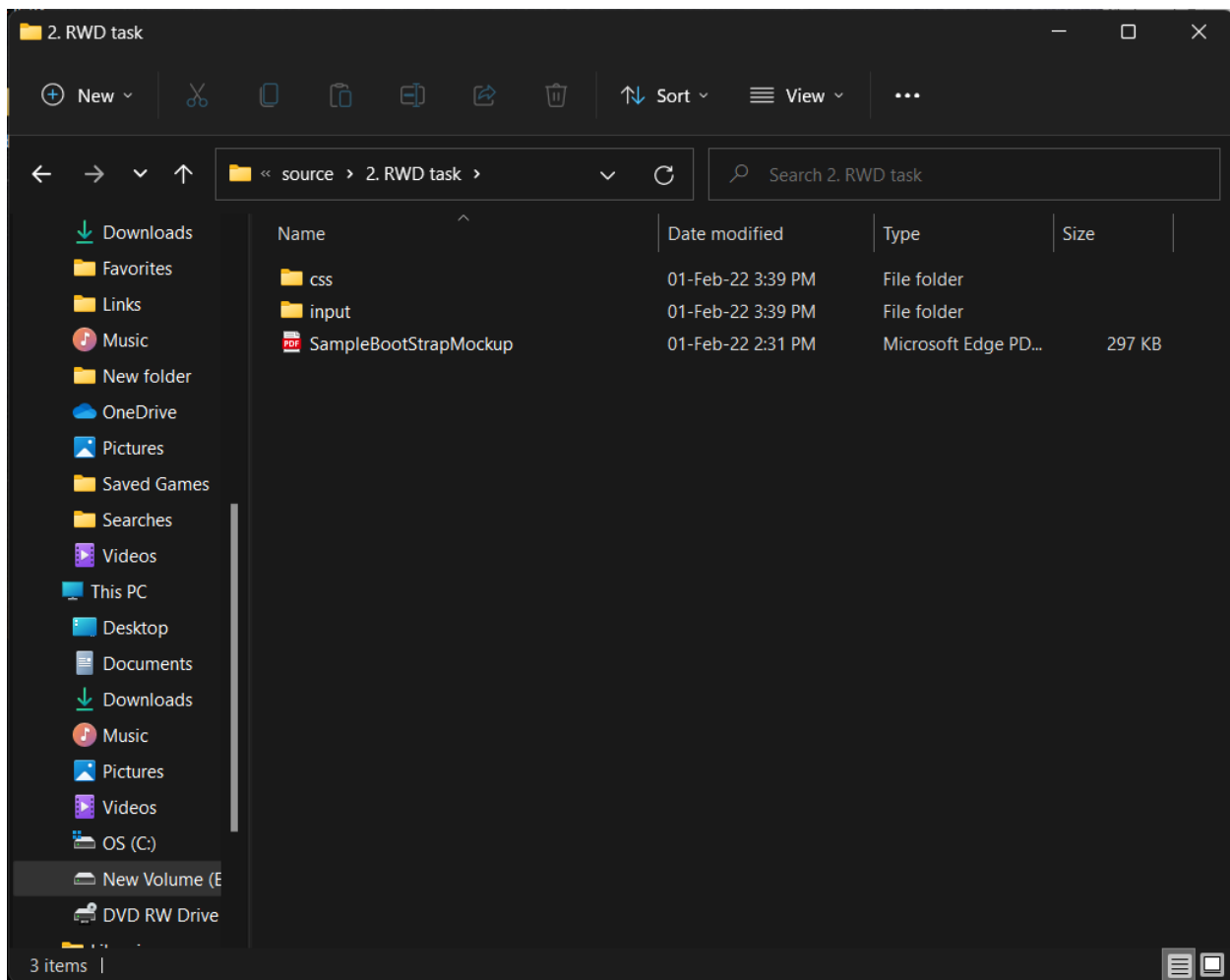
This is the output of the game webpage that was developed using HTML, CSS, and JAVASCRIPT.



Below shows the bottom colored underlined border when user hovers the pointer on any the three images/choices.



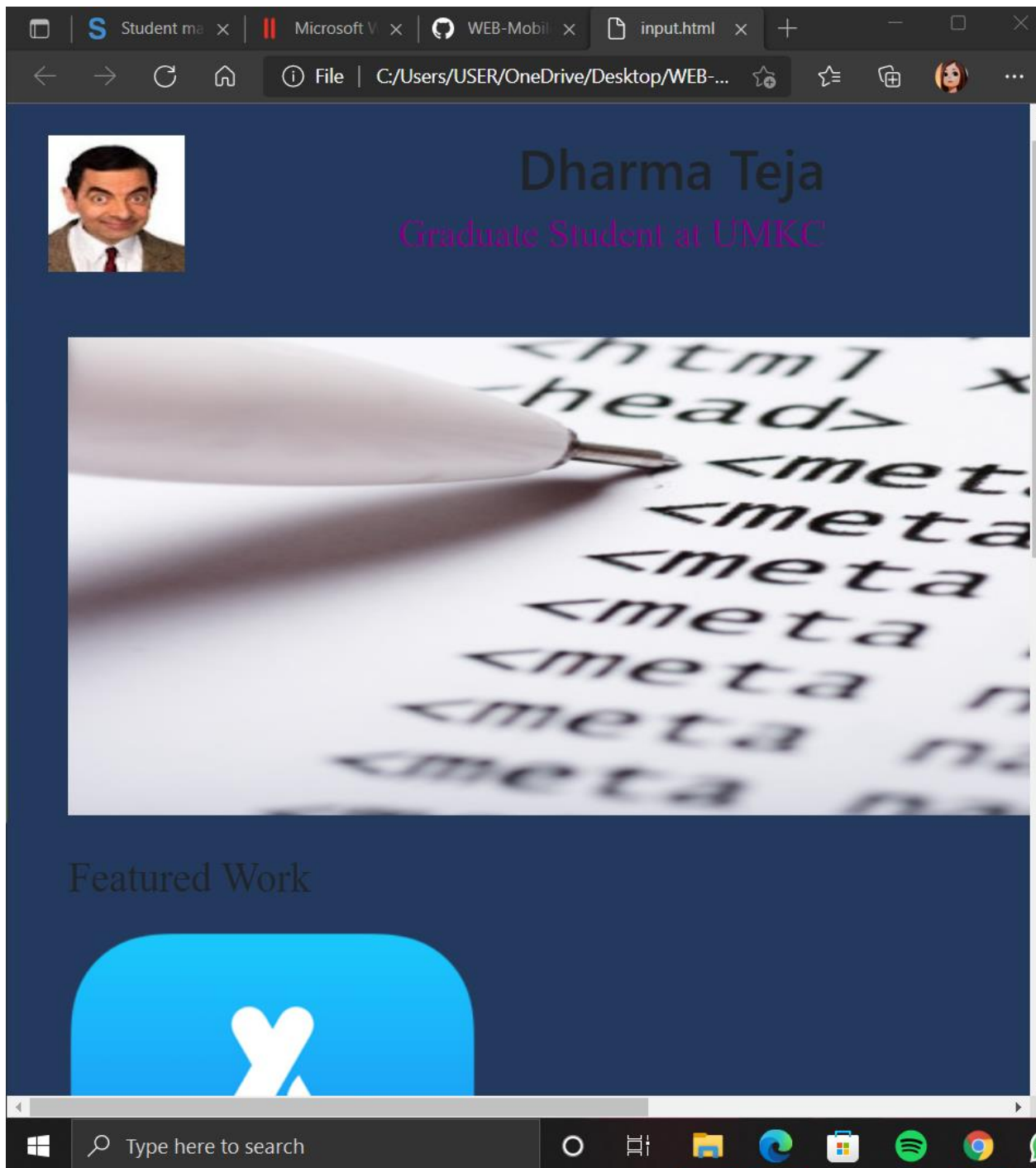
2. RWD Task

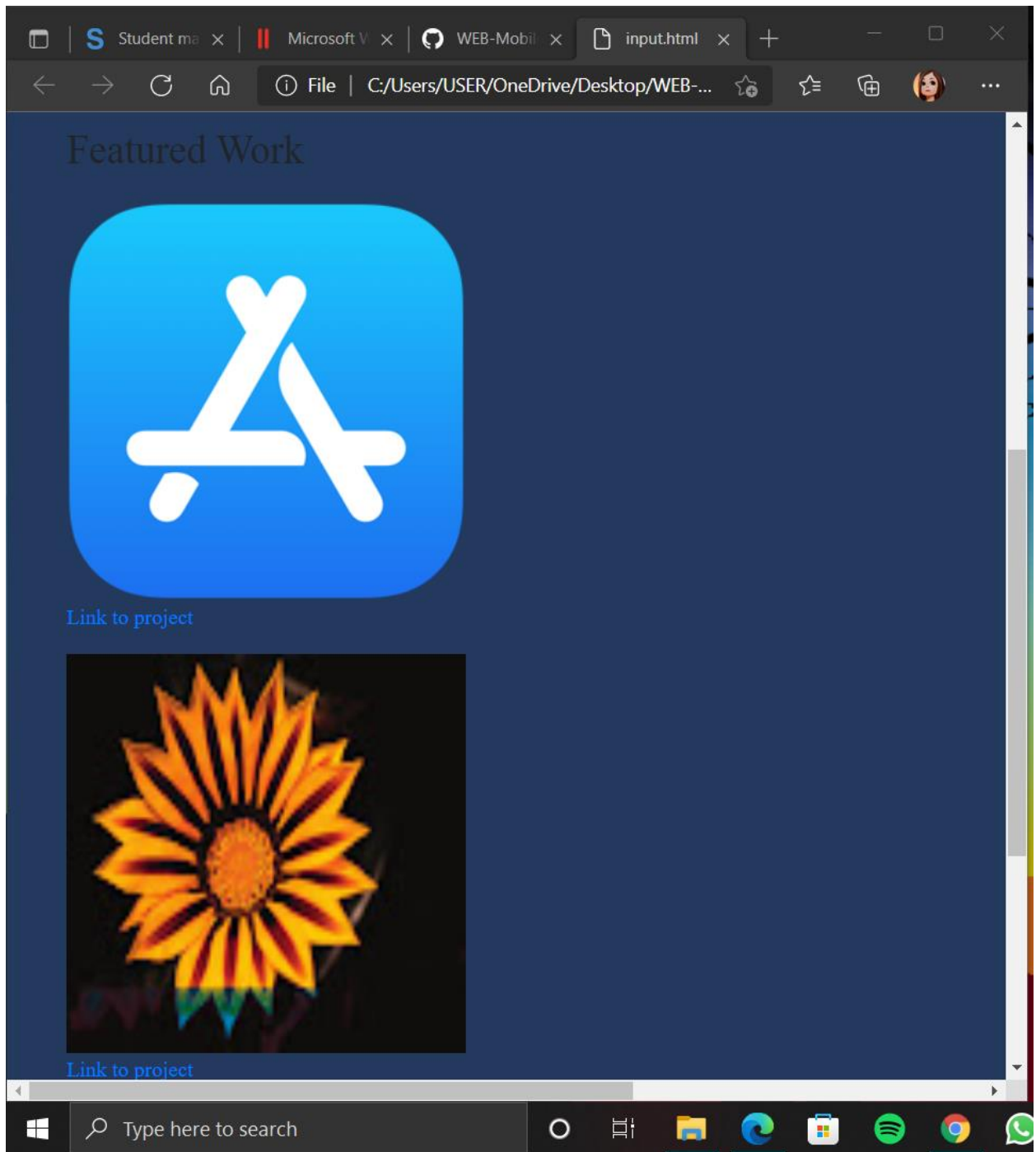


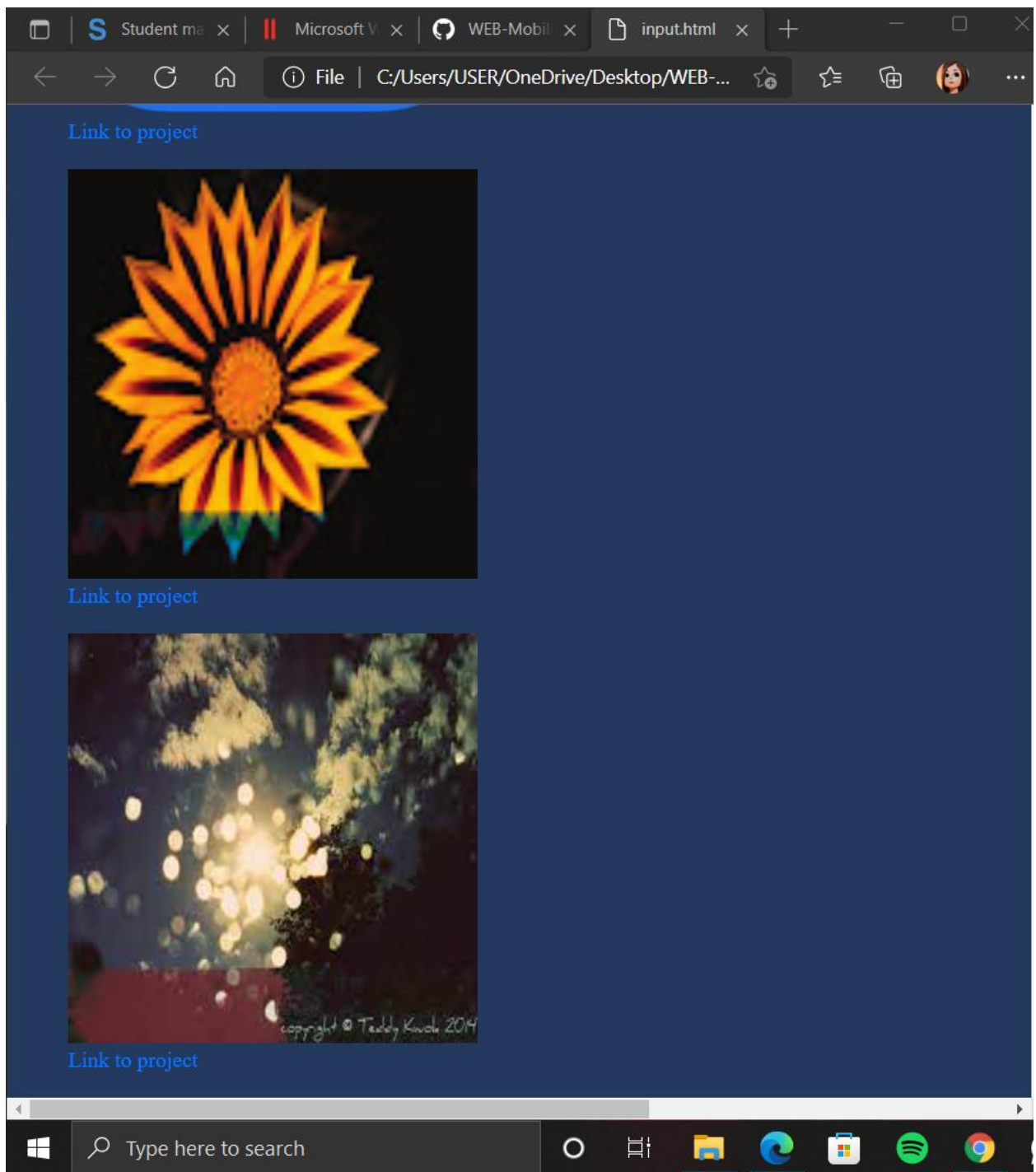
Below are the html code snaps we worked on.

OUTPUT










Student management system

Microsoft Word - Document1


WEB-MobileProgramming/AWE


input.html


C:/Users/USER/OneDrive/Desktop/WEB-MobileProgramming/Web-and-Mobile-programming-spring-2022/Webpart/ICP3/source/RWD%20task/input/input.html




Featured Work


[Link to project](#)


[Link to project](#)


[Link to project](#)

Type here to search



2247
01-02-2022