Anusha Nelluri – anhx2@umsystem.edu
GitHub Link -https://github.com/NelluriAnusha/Demo_Remote/tree/main/Webpart/ICP3

Achyuth Kumar Valeti – avgh3@umsystem.edu
GitHub Link - https://github.com/AchyuthValeti/Demo_Remote/tree/main/Webpart/ICP3

ICP3 (Bootstrap, JavaScript)

Introduction:

Bootstrap:
Bootstrap is a free and open-source CSS framework for front-end web development which is responsive and mobile-first. It incorporates templates for typography, forms, buttons, navigation, and other UI components which are based on CSS and (optionally) JavaScript.

JavaScript:
JavaScript is a scripting language that is lightweight, cross-platform, and interpreted. It is well-known for web page creation, but it is widely used in numerous non-browser applications. JavaScript can be used to create both client-side and server-side applications.
The most well-known application of JavaScript is web page development.

Tasks:

Task-1: Build Rock-Paper-Scissors Game using JavaScript

Rock paper scissors is a classic two player game. Each player chooses either rock, paper, or scissors. The items are compared, and whichever player chooses the more powerful item wins.

The possible outcomes are:

- Rock destroys scissors.
- Scissors cut paper.
- Paper covers rock.
- If there's a tie, then the game ends in a draw.
-

Coming to the coding part, we used HTML, CSS & JavaScript to develop the webpage and build the game.

HTML: It is used to define the structure of a webpage. CSS defines its style. To create any webpage, we need to know about HTML and CSS.

- Firstly, we have created a HTML and CSS file and placed it in the source folder of html file.
- The following webpage is created using HTML and CSS and we have explained the each tag below.

- **Heading:** To display our titles or subheadings in a webpage we use Headings in HTML. Headings can be displayed by using h tags (<h1 >to <h6>) based on the size you want to present. We have shown "Play the game of Rock, Paper and Scissors" as main heading using h1 tag and also used h2 tag to print subdivision content.
- We used <p> tags for including both computer and player choices.
- We have also used and tags for representing the group of elements and its styling purposes.
- We also used container class for responsive, fixed-width container, which means its maximum width varies at each breakpoint. The following code is used in my html page.

Code:

```
<div class="container">
  <!-- Using h1 tag for printing header -->
  <h1 id="info"> Play the game of Rock, Paper and Scissors</h1>
```

CSS:

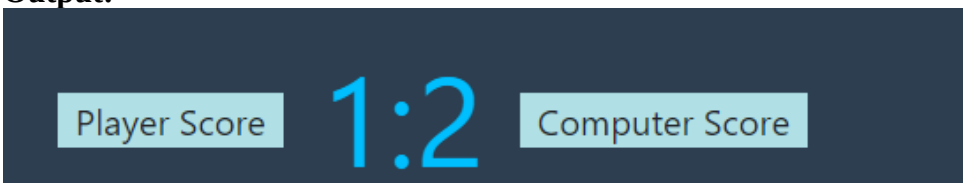
CSS is mainly used for styling purposes, and I have used different classes and their properties for my web page designing.
Here is the code which I have used for printing with different colors and properties.

Code:

```css
.score-container{
    position: relative;
    margin: 0 auto;
    width: max-content;
}

.user{
    position: relative;
    left: 0;
    top: 50%;
    transform: translate(-50%, -50%);
    background-color: powderblue;
    width: 40px;
    height: 15px;
    font-size: 20px;
    padding: 2px 10px;
    box-sizing: border-box;
}
.computer{
    position: relative;
    right: 0;
    top: 50%;
    transform: translate(50%, -50%);
    background-color: powderblue;
```

Output:



Javascript:

- Initially, we have created a class “playGame” using the class keyword in main.js file.
- Then Created an object using “new” keyword in html file and linked the javascript file using <script> tag with HTML.

```
let result = new playGame('score', 'player', 'computer', 'rock', 'paper', 'scissors', 'info');
```

- We used the Constructor method to initialize all the objects values like rockElement, paperElement etc. This method is called automatically when we create an object. So, I have created an object in html page and this method will be called automatically.
- Initial values are initialized in the constructor method.

```
constructor(scoreId, playerId, computerId, rockId, paperId, scissorsId, infoId) {
```

- We used “this” keyword and it referes to an object & which object is used depending on how this is done (used or called). When used in an object method, `this` refers to the **object**.

```
// assigning values for score, computer and player
this.scoreElement = document.getElementById(scoreId);
this.playerElement = document.getElementById(playerId);
this.computerElement = document.getElementById(computerId);

// assigning values for player options(rock, paper & scissors)
this.rockElement = document.getElementById(rockId);
this.paperElement = document.getElementById(paperId);
this.scissorsElement = document.getElementById(scissorsId);

// info element is used to display the result info
this.infoElement = document.getElementById(infoId);
```

- I have written one function I.e., “play” to get the scores for PLayer and Computer.

When the payer click on User choice(like rock, paper, scissors) then it will go to the “play” method.

```
this.rockElement.onclick = () => {
    //calling play method when we click on "rock" option
    this.play( playerChoice: 1);
}
this.paperElement.onclick = () => {
    //calling play method when we click on "paper" option
    this.play( playerChoice: 2);
}
this.scissorsElement.onclick = () => {
    //calling play method when we click on "scissors" option
    this.play( playerChoice: 0);
}
```

- Using the definition of “play” method, scores will be displayed on web page as below.

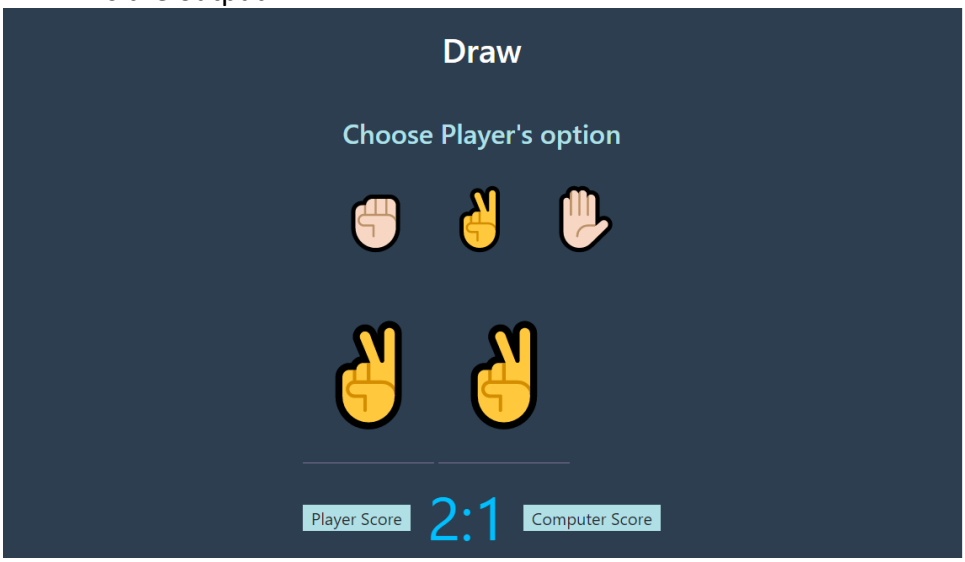
- This “play” method again calls for “evaluate” & “updateUI” methods to get the results of Player choices. Using these methods, images on webpages will be displayed correctly.
- In “evaluate” method, we have used conditional statements to check the different combinations in between player & computer and returned the output like “Win”, “Draw” and “You Lost”.

```
// definition for evaluate method and it will return the result of player & computer choices
evaluate(playerChoice, computerChoice) {
    if (playerChoice == computerChoice) {
        return 'Draw';
    }else if (playerChoice == 'Rock' && computerChoice == 'Scissors') {
        return 'Win';
    }else if (playerChoice == 'Paper' && computerChoice == 'Rock') {
        return 'Win';
    }else if (playerChoice == 'Scissors' && computerChoice == 'Paper') {
        return 'Win';
    }else {
        return 'You lost';
    }
}
```

- UpdateUI method is used to display the output like score on HTML page directly.

Output:

- When Players choice and computer's choice is same, then the result would be **Draw**. Here is the output.



- Based on the selection of choice of player, it will check with computer choice. Thes the result will be displayed as either **Win** or **You Lost**. Here is the webpage look likes.

You lost

Choose Player's option

Player Score 2:2 Computer Score

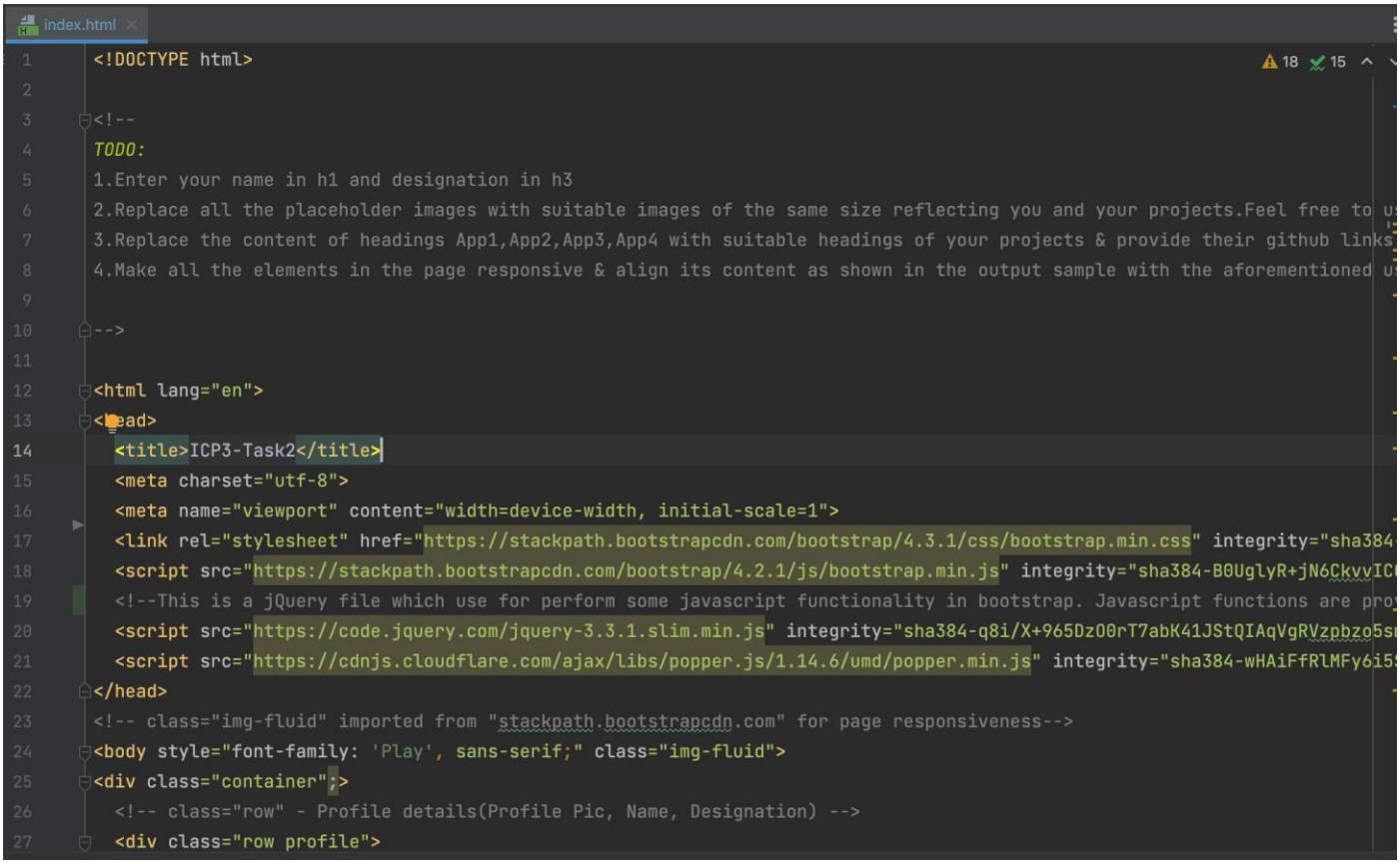
Win

Choose Player's option

Player Score 1:0 Computer Score

Task-2: Responsive Web Design (RWD)

Responsive web design (RWD) or responsive design is a web design approach that aims to ensure usability and satisfaction for showing web pages well on a variety of devices and window or screen sizes from small to big.



<meta> tags:

1. <meta charset=”UTF-8”>: The charset element specifies the HTML document’s character encoding. The UTF-8 character set, which covers practically all the characters and symbols in the world, is recommended by the HTML5 specification.
2. <meta name=”viewport” content=”width=device-width, initial-scale=1.0”>: This tells the browser how to change the dimensions and scale of the page. The “width=device-width” section adjusts the page’s width to match the device’s screen size (which will vary depending on the device). When the website is first loaded by the browser, the “initial scale=1.0” section specifies the initial zoom level.

<Link> Tag: It is used to link bootstrap CSS file into the code.

<class=" Containers">: Containers allow you to center and pad your site's content horizontally.
<class=" row">: Columns are wrapped in rows.
<class=" col-sm">: responsive class for small devices with a screen width of 576 pixels or more.
<hr>: Thematic break used to separate two paragraphs or content in a webpage.
<img>: This tag is used to embed an image in our webpage.

```
index.html
55            Git Introduction
56          </a>
57        </p>
58      </div>
59      <div class="col-sm-4">
60        <img alt="PlaceHolderImage3" src="../images/ICP2.jpeg" class="img-fluid" style="height:300px; width:555px">
61        <h3 style="font-size:3vw; color: #747704;">Web ICP2</h3>
62        <p style="font-size:1.5vw;">
63          <a href="https://github.com/AchyuthValeti/Demo_Remote/tree/main/Webpart/ICP2" target="_blank">
64            HTML, CSS
65          </a>
66        </p>
67      </div>
68      <div class="col-sm-4">
69        <img alt="PlaceHolderImage3" src="../images/app3.jpeg" class="img-fluid" style="height:300px; width:555px">
70        <h3 style="...">Web ICP3</h3>
71        <p style="font-size:1.5vw;">
72          <a href="https://github.com/AchyuthValeti/Demo_Remote/tree/main/Webpart/ICP3" target="_blank">
73            Bootstrap
74          </a>
75        </p>
76      </div>
77    </div>
78  </div>
79
80  </body>
81  </html>
```

Final Output:

Achyuth Kumar Valeti

UMKC Graduate Student

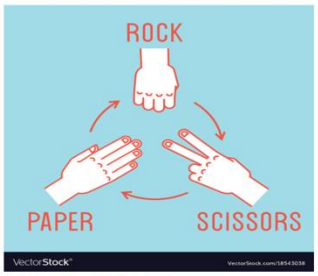
Featured Work

Web ICP1

Git Introduction

Web ICP2

HTML, CSS



Web ICP3

Bootstrap

Contribution

We have contributed equally.

Conclusion

In this ICP3, we have learned bootstrap, javascript and developed a webpage using the same.

Challenges

And, we have not faced any major challenges while doing the assignment.