

ICP-3

Keenan Flynn – kpfxn8@umsystem.edu – <https://github.com/kfly2fly/Web-Mobile-Spring-2022>

Jasmine Naraine- jnytc@umstyem.edu -<https://github.com/JNaraine/Web>

1. Rock-Paper-Scissors

The task was to create a Rock, Paper, Scissors game using JavaScript. Within the game, images have to be used and be able to be clicked on when playing the game.

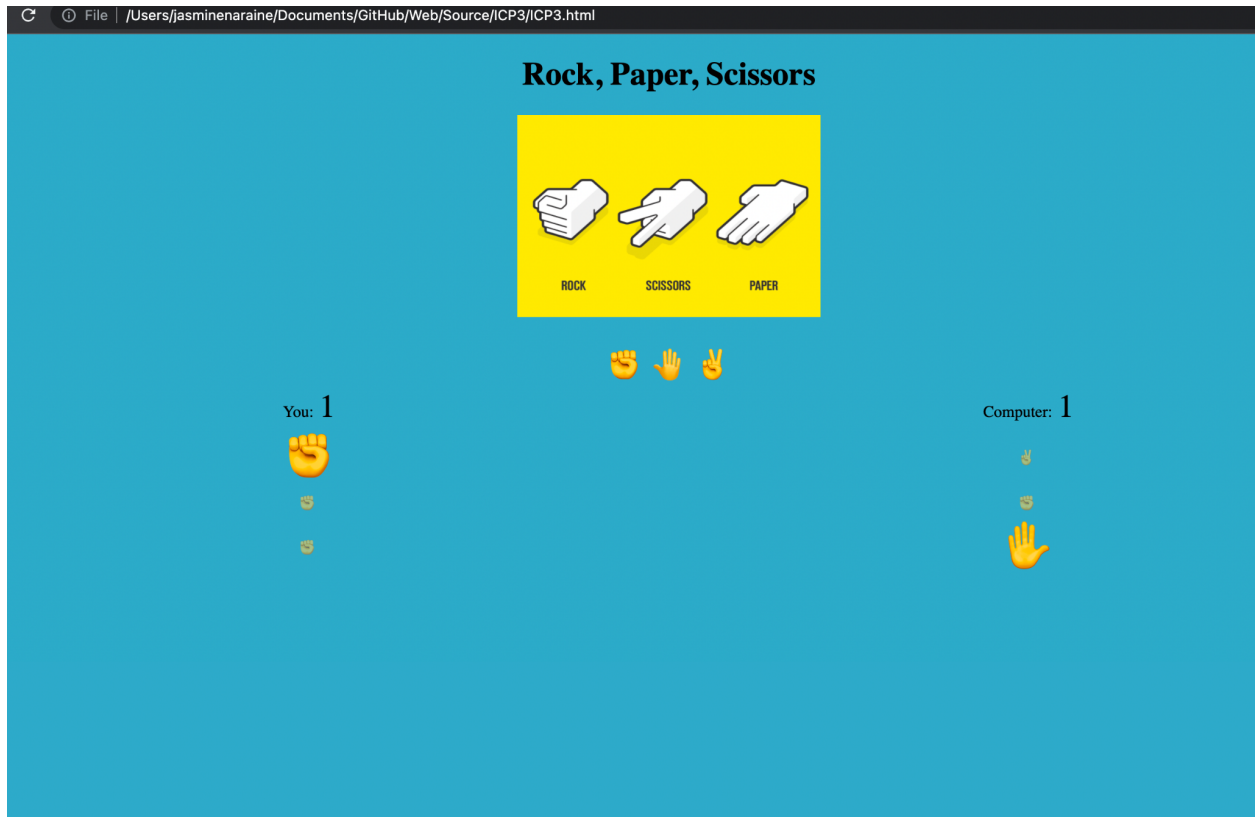
```
1  const selectionButtons = document.querySelectorAll('[data-selection]') //Be able to click on each emoji
2  const finalColumn = document.querySelector('[data-final-column]') //images will show at the top when playing
3  const computerScoreSpan = document.querySelector('[data-computer-score]') //Computer score
4  const yourScoreSpan = document.querySelector('[data-your-score]') //Update the score of the game
5  const SELECTIONS = [
6    {
7      //rock beats scissors
8      name: 'rock',
9      emoji: '🪨',
10     beats: 'scissors'
11   },
12   {
13     //paper beats rock
14     name: 'paper',
15     emoji: '📄',
16     beats: 'rock'
17   },
18   {
19     //scissors beat paper
20     name: 'scissors',
21     emoji: '✂️',
22     beats: 'paper'
23   }
24 ]
25 // To be able to click on each emoji
26 selectionButtons.forEach(selectionButton => {
27   selectionButton.addEventListener('click', e => {
28     const selectionName = selectionButton.dataset.selection
29     const selection = SELECTIONS.find(selection => selection.name === selectionName)
30     makeSelection(selection)
31   })
32 })
33 //Winner is depending on which ever selections is being clicked on
34 function makeSelection(selection) {
35   const computerSelection = randomSelection()
36   const yourWinner = isWinner(selection, computerSelection)
37   const computerWinner = isWinner(computerSelection, selection)
38
39   addSelectionResult(computerSelection, computerWinner)
40   addSelectionResult(selection, yourWinner)
41
42   if (yourWinner) incrementScore(yourScoreSpan)
43   if (computerWinner) incrementScore(computerScoreSpan)
44
45 }
```

```

45 |
46 | }
47 | //Increase score of which ever winner
48 | function incrementScore(scoreSpan) {
49 |     scoreSpan.innerText = parseInt(scoreSpan.innerText) + 1
50 | }
51 | //Showing the result using the emoji
52 | function addSelectionResult(selection, winner) {
53 |     const div = document.createElement('div')
54 |     div.innerText = selection.emoji
55 |     div.classList.add('result-selection')
56 |     if (winner) div.classList.add('winner')
57 |     finalColumn.after(div)
58 |
59 |
60 | }
61 | // Winner is the best selection
62 | function isWinner(selection, opponentSelection) {
63 |     return selection.beats === opponentSelection.name
64 |
65 |
66 | }
67 | // Random selection for the computer
68 | function randomSelection() {
69 |     const randomIndex = Math.floor(Math.random() * SELECTIONS.length)
70 |     return SELECTIONS[randomIndex]
71 |
72 |
73 | }
74 |
75 |
76 |
77 |

```

Within the game, whenever the player chooses one of their selections. The computer will respond with their own selection. If a player wins, the emoji will look bigger and the loser the opacity will look dimmer. It is the same way for the computer. If there is a tie game both the player and computer emoji opacity will look dim. Within each game, there is a score keeping up with the player and computer.



2. RWD Task

The task was to imitate a responsive webpage that looked like the following. We are able to do this by implementing several responsive techniques. To make sure the webpage scales to our screen, it is important that we have the following tag.

```
<meta name="viewport" content="width=device-width">
```



STEVE SMITH

FRONT-END DEVELOPER

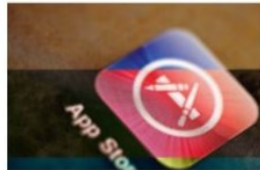
FOR
REFERENCE
ONLY



BCBBBB
747704



Featured Work



APIFY

<https://github.com/sxxc5/Appify/>



SUNFLOWER

<https://github.com/sxxc5/Sunflower/>



BOKEH

<https://github.com/sxxc5/Bokeh/>

The webpage was created using classes from Bootstrap. These classes allow us to create a grid layout which is important for design. The grid was created by using `<div>` tags with either row or column classes. Inside of these grid cells is where the content will go. These grid cells allow us to customize spacing by using classes such as the container class or mb/mt classes.

Bootstrap also comes with preinstalled classes that will make our page responsive such as img-fluid and the display headings. When making personalized classes for things such as image size, we were careful to not specify large pixel numbers, rather we used percentages. Finally, we also used @media tags to specify what our webpage will look like on screens smaller than 600 pixels. This is helpful for mobile devices.



Keenan & Jasmine

Students



Featured Work



APPIFY

[Link to Appfiy](#)



SUNFLOWER

[Link to Sunflower](#)



BOKEH

[Link to Bokeh](#)

```

input.html  index.css  x
1  <!DOCTYPE html>
2  <html lang="en">
3  <!-- Create responsive webpage -->
4  <meta name="viewport" content="width=device-width">
5  <!-- Import Bootstrap and index.css as stylesheet -->
6  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-18mE4kWA"
7  <link rel="stylesheet" type="text/css" href="index.css">
8  <head>
9    <title>RWD Task</title>
10   <meta content="text/html; charset=windows-1252" http-equiv="Content-Type">
11 </head>
12 <body class="container mt-4 mb-4"> <!-- Create margins around the webpage -->
13 <div class="row"> <!-- Create a grid for styling -->
14   <div class="col">
15     <!-- The uppermost grid row includes the name and headshot -->
16     <div class="row">
17       <div class="col"></div>
18       <div class="col right"> <!-- .right makes the text align right -->
19         <h1 class="display-1 green">Keenan & Jasmine</h1> <!-- Overwrite the display classes to get correct color -->
20         <h3 class="display-5 green">Students</h3> <!-- The bootstrap display classes are responsive -->
21       </div>
22     </div>
23     <!-- The horizontal rule divides the uppermost row from the rest of the page -->
24     <hr class="hr_class">
25     <br>
26     <!-- Second grid row includes main picture -->
27     <div class="row">
28       <!-- img-fluid makes the image responsive and fit_main specifies max dimensions and fit -->
29       
30     </div>
31     <br>
32     <!-- Third grid row provides links to featured works -->
33     <div class="row">
34       <h2 class="display-3 grey">Featured Work </h2>
35       <!-- Divide grid row into 3 columns, one for each app -->
36       <!-- Center the text and make it green -->
37       <div class="col-4 text-center green">
38         <!-- Create responsive text and bottom margin -->
39         
40         <h3 class="display-5">APPIFY</h3>
41         <!-- Remove underline from link -->
42         <p class="small"><a href="#" class="app_link">Link to Appfiy</a></p>
43       </div>
44       <!-- Repeat above steps for other app -->
45       <div class="col-4 text-center green">
46         
47         <h3 class="display-5">SUNFLOWER</h3>
48         <p class="small"><a href="#" class="app_link">Link to Sunflower</a></p>
49       </div>
50       <!-- Repeat above steps for other app -->
51       <div class="col-4 text-center green">
52         
53         <h3 class="display-5">BOKEH</h3>
54         <p class="small"><a href="#" class="app_link">Link to Bokeh</a></p>
55       </div>
56     </div>
57   </div>
58 </div>
59 </body>
60 </html>

```

input.html

index.css

✕

```
7  .right {
8      text-align: right;
9      align-content: right;
10 }
11
12 /*Add color to text*/
13 .green{
14     color: #747704;
15 }
16 .grey{
17     color: #8CB88B;
18 }
19
20 /*Customize horizontal bar*/
21 .hr_class {
22     min-height: 10px;
23     color: #747704;
24 }
25
26 /*Customize image fit*/
27 .fit_main{
28     width: 100%;
29     max-width: 100%;
30     max-height: 500px;
31     overflow: hidden;
32     object-fit: cover;
33 }
34
35 .fit{
36     width: 100%; /* You can set the dimensions to whatever you want */
37     height: 80%;
38     overflow: hidden;
39     object-fit: cover;
40 }
41
42 /*Remove underline from links and add color*/
43 .app_link{
44     color: #747704;
45     text-decoration: none;
46 }
47
48
49 /* If the screen size is 600px wide or less, set the display-5 font-size will be 150% and small will be 15px*/
50 @media screen and (max-width: 600px) {
51     .display-5 {
52         font-size: 150%;
53     }
54     .small {
55         font-size: 15px;
56         background-color: skyblue;
57     }
58 }
59
60 /*If screen size is bigger, font-size will be 120%*/
61 @media screen and (min-width: 601px) {
62     .small {
63         font-size: 120%;
64     }
65 }
```

Contribution

Jasmine Naraine - Rock, Paper, Scissors

Keenan Flynn - RWD Task