Introduction to Multimedia:

Multimedia Web Page Development
Assignment 1

Candidate number: 215813

Table of Contents

Introduction

The main reason behind writing this report is to discuss my first web page development in HTML. This web site was developed on Google Chrome and coded using sublime text editor. In order to help with the design and overall appearance of the page, JavaScript and CSS were also used. Throughout this report, I will discuss the development of my website and give an overview of what I have implemented and where it has been implemented. My struggles and difficulties on this journey to create my first web page will also be mentioned. The report will be divided up into section. These sections represent the overall design of the web site and each page that has been created in order to keep the report organised and easy to read.

Overall design

The web site is composed of 4 tabs which will be described and explained further down. When the web site is first open, the user is presented with the home page. From here we can see the colourful design of the web site. The navigation bar is the same throughout the whole web site. It was made using an unordered list within a coloured box. In order to make the navigation bar more intuitive, the *:hover* selector was used so that when the mouse hovers over one of the options, it will change the box colour in order to make it visible that the user's mouse is over a certain option. To enable the user to know which page they are viewing, **.nav .current** status was added so that when the option that was clicked on, would stay the same colour as the background. The <nav> tag was used to implement the navigation bar onto the web site.

```
/*Nav bar design*/
nav ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
    overflow: hidden;
    background-color: #00FFCC;
}
nav li {
    display: inline;
    float: left;
}
nav li a {
    display: block;
    color: blue;
    text-align: center;
    padding: 14px 16px;
    text-decoration: none;
}
/**background nav bar colour when mouse hovers over*
nav li a:hover {
    background-color: #808080;
}
/*Highlights current page*/
.nav .current{
    background-color: #FFCE55;
}
```

Figure 1 Navigation Bar Design CSS

On all the pages (except hobbies), I implemented a "Dark Mode" button which changes the background colour to black as well as changes black fonts into white fonts in order to be able to see the text. This switchable CSS style was design in CSS, and I used a JavaScript function named "toggledarkmode" to create it. The "Dark Mode" button was then set to specific location in all pages by setting its position property to absolute. The <button> tag with the class onclick was used to create the button itself.

```
/*Changes colour of letters*/
.dark-mode{
    background: black;
    color: white;
}
/*Dark mode button position*/
button{
    position: absolute;
    padding-top: 0.5em;
}
```

Figure 2 Dark Mode Design CSS

```
/*Toggle dark mode*/
function toggleDarkMode() {
    var element = document.body;
    element.classList.toggle("dark-mode");
    console.log("Dark mode enables/disabled")
}
```

Figure 3 Dark Mode Design JavaScript

Home Page

The home page acts as a main menu as it's the first web page to appear. On this page, other than the navigation bar and the dark mode button mentioned above, there is an image imbedded using the using "*<img src="images/multimedia.jpg" alt="Multimedia" width="600" height="402">.*"

Above the image is a welcome message. To make it bigger I used <h1> tag to set it to heading 1. Bellow, I added a footer with my candidate number in case it was necessary. When the Dark mode(on/off) button is clicked, is changes the background colour to black, and the black text under the image changes to white for it to be seen in dark mode.

Figure 4 Home Page

Video Greeting Page

This page, like the previous home page, follows the same simplistic and easy to use design. On this page the navigation bar remains but the "Video greeting" option has changed to the same colour as the background to let the user know what page they are on. Underneath we have the dark mode button again. In the middle of the page I have embedded a short 10second video made using the Microsoft Video Editor. When editing this video, the slides are in sync with the implemented background music so that when the beat changes in the music, the content of video changes.

To set the size of the video on the screen, this is used: "<video width="900" height="550" controls>". This also displays controls for the video such as the play/pause button, a bar to select where to play the video and volume controls to adjust the music volume. To imbed the video itself, the <source src="images/videogreeting.mp4" type="video/mp4"> tag is used which finds the location of the video in mp4 format.

To adjust where the video is placed on the web page, CSS is used to position the video a certain amount away from the top and left margins using the unit em so that always remains in the same position in all browsers.
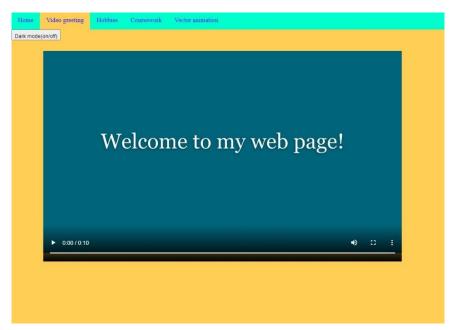

Figure 5 Video Greeting Page

Hobbies Page

Next is the hobbies page, where a few of my hobbies and corresponding photos belong. Originally, this page was composed of three small texts about the hobbies and pictures of each bellow. In order to make it more unique and stand out, an automatic photo gallery with corresponding captions was added. In order to do this, the dark mode button had to be eliminated as the photo gallery occupies most of the page. To be able to do this, a <div> tag was used with the id attribute which species what the division is for, in this case it was named "album-with-captions". inside that is a <figure> tag with the corresponding class: <figure class="slider">. This is so it can be styled with CSS. Inside that are 3 more <figure> tags with one of the images in each one with the corresponding caption. To make the sliding gallery work, the @keyframes rule in CSS is used. This allows the images to "slide" little by little as the change into the next photo. JavaScript is also applied to make the automatic gallery possible. To do this these functions are used: function plusDivs(n){} and function showDivs(n){}.

```html
<div id="album-with-captions">
    <figure class="slider">
        <figure>
            <img src="images/paddle.jpg" alt>
            <figcaption>Paddle tennis is enjoyable with friends</
            figcaption>
        </figure>
        <figure>
            <img src="images/football.jpg" alt>
            <figcaption>I enjoy playing football with my friends in the
            park</figcaption>
        </figure>
        <figure>
            <img src="images/golf.jpg" alt>
            <figcaption>I play golf with my father and in competitions</
            figcaption>
        </figure>
    </figure>
</div>
</body>

</html>
```

Figure 6 Slideshow HTML Code

```css
/*photo gallery slideshow*/
@keyframes slide {
0% { left: 0%; }
20% { left: 0%; }
25% { left: -100%; }
45% { left: -100%; }
50% { left: -200%; }
70% { left: -200%; }
75% { left: -300%; }
95% { left: -300%; }
100% { left: -400%; }
}
* {
  box-sizing: border-box;
}
body, figure {
  margin: 0; background: #FFCE55;
  font-weight: 100;
}
div#album-with-captions {
  width: 100%; overflow: hidden;
}
figure.slider {
  position: relative; width: 500%;
  font-size: 0; animation: 18s slide infinite;
}
figure.slider figure {
  width: 20%; height: auto;
  display: inline-block;  position: inherit;
}
figure.slider img { width: 100%; height: auto; }
figure.slider figure figcaption {
  position: absolute; top: 0;
  background: #0F1D18;
  color: white; width: 100%;
  font-size: 2rem;
}
```
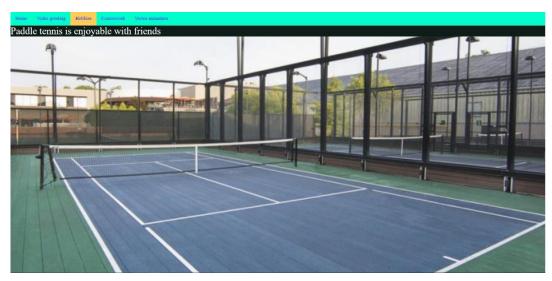
Figure 7 Slideshow CSS Code

```javascript
/*slideshow*/
var slideIndex = 1;
showDivs(slideIndex);

function plusDivs(n) {
  showDivs(slideIndex += n);
}

function showDivs(n) {
  var i;
  var x = document.getElementsByClassName("slideshow");
  if (n > x.length) {slideIndex = 1}
  if (n < 1) {slideIndex = x.length};
  for (i = 0; i < x.length; i++) {
    x[i].style.display = "none";
}
x[slideIndex-1].style.display = "block";
}
```

Figure 8 Slideshow JavaScript Code

Figure 9 Hobbies Page

Coursework Page

This is the page to access the coursework I have done so far this semester. On this page, we have the same navigation bar as before and the dark mode button bellow. Bellow all of this is a text in heading 1 format. So far, both text on the home page and this page are the same font, size and colour. To help reduce the amount of repeated code, these texts were styles using CSS. In the main.CSS file, all <h1> tags are set to the colour red and a certain distance from the left margin(2em) so that they are both in the same position.

Bellow the header, there are separate texts with links underneath each one. The texts explained what each link is. The links are for the coursework from lab 3(part 3), lab 4 and this very same report.

Lab 3(part3) coursework was an exercise on CSS where we had to modify the design of a webpage cover. Lab 4 coursework was a JavaScript animation of a blue background and a green ball that bounces from one side to the other while descending to the middle of the bottom. The third link will show a pdf document of this report. The coursework will open in a new tab as a result of using "target="_blank"" in the <a href => tag.

Figure 10 Coursework Page

SVG Animation

The final page is an animation. When the user clicks on "SVG animation", the animation starts in a new tab the same way the coursework does. The animation is a blue ball that bounces off the sides of the coloured area. The instructions for this project mentioned a specific audio file (brighton.mp3) that must be implemented in the animation. Instead, I added a "play audio" button so that if/when the user would like to listen to the music, all they must do is click on the button at the top left of the page. This was done by creating a function in the JavaScript file called playAudio().

In order to try and make the animation more complex, a button was added bellow the animation to enable the user to change the shade/colour of the background. This was done using a function called colorChange(). To add to the complexity, a slider bar was added to modify the size of the ball. This is done by swapping the size of the ellipse in the "function draw ()" for" slider.value()" which sets the size of the ball to whatever is set on the slider bar.

```
var bgcolor;
var button;
var audio;
var slider;
let x = 300;
let y = 300;
let xspd = 10;
let yspd = 7;
let r = 25;
function setup(){
  canvas = createCanvas(600, 600);
  bgcolor = color(200);
  button = createButton("Change background colour");
  button.mouseClicked(colorChange);
  slider = createSlider(5, 200, 20);
}
function playAudio(){
  new Audio("brighton.mp3").play();
}

function colorChange(){
  bgcolor = color(random(100));
}


function draw() {
  background(bgcolor);
  fill("blue");
  ellipse(x, y, slider.value(), slider.value());
  x += xspd;
  y += yspd;
  if (x > width - r || x < r) {
    xspd = -xspd;
  }
  if (y > height - r || y < r) {
    yspd = -yspd;
  }
}
```
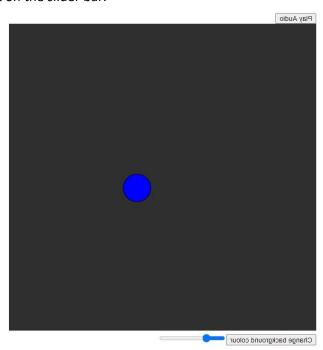
Figure 11 Animation JavaScript Code

Figure 12 Animation

```
<!DOCTYPE html>
<html>
<head>
    <title>Animation</title>
    <script src="p5/p5.js" type="text/javascript"></script>
    <script src="raphael.min.js" type="text/javascript"></script>
    <script src="svganimation.js" type="text/javascript"></script>

</head>
<body>
    <button onclick="playAudio()">Play Audio</button>
</body>
</html>
```

Figure 13 Animation HTML Code

Conclusion

   To conclude, I believe this project has been a success and I have learnt a lot about HTML and its relationship with CSS and JavaScript. I find it enjoyable to do HTML and it makes me realise how much time must go into making normal website such as the Sussex Direct. Improvements I would do if I had more time would the overall design of the website. I would change the colours, fonts and positions texts and images better. Another improvement would be to the SVG animation, I would make more complex by adding a way to change the colour of the ball and add more balls and make them bounce into each other and change colour at the same time. For me personally, the animation was the toughest part of this project, both creatively and knowledge wise as my knowledge on animation is extremely limited.

References

Alan, M., 2018. *What is Padel / Paddle tennis / Platform tennis? And what are the differences?.* [Online]
Available at: https://www.epiruslondon.com/blogs/tennis-hacks-blog/what-is-padel-paddle-tennis-platform-tennis-and-what-are-the-differences
[Accessed 14 November 2020].

BGIA.org.uk authors, 2020. *Who are the BGIA?.* [Online]
Available at: https://bgia.org.uk/
[Accessed 14 November 2020].

iconmultimedia.in authors, 2017. *iconmultimedia about us.* [Online]
Available at: https://iconmultimedia.in/about.html
[Accessed 14 November 2020].

Soueidan, S., 2018. *A Guide to SVG Animations (SMIL).* [Online]
Available at: https://css-tricks.com/guide-svg-animations-smil/
[Accessed 9 November 2020].

w3schools.com authors, 2020. *CSS Tutorial.* [Online]
Available at: https://www.w3schools.com/css/default.asp
[Accessed 10 November 2020].

w3schools.com authors, 2020. *CSS Units.* [Online]
Available at: https://www.w3schools.com/cssref/css_units.asp
[Accessed 15 November 2020].

w3schools.com authors, 2020. *How TO- Slideshow.* [Online]
Available at: https://www.w3schools.com/howto/howto_js_slideshow.asp
[Accessed 15 November 2020].

w3schools.com authors, 2020. *JavaScript Tutorial.* [Online]
Available at: https://www.w3schools.com/js/default.asp
[Accessed 10 November 2020].

w3schools.com authors, 2020. *SVG Tutorial.* [Online]
Available at: https://www.w3schools.com/graphics/svg_intro.asp
[Accessed 11 November 2020].

w3schools.com authors, 2020. *W3.CSS Slideshow.* [Online]
Available at: https://www.w3schools.com/w3css/w3css_slideshow.asp
[Accessed 15 November 2020].

worcester city council, 2019. *In-principle agreement announced to bring Worcester City Football Club home.* [Online]
Available at: https://www.worcester.gov.uk/news/in-principle-agreement-announced-to-bring-worcester-city-football-club-home
[Accessed 14 November 2020].

Table of Figures