MULTIMEDIA WEB PAGE DEVELOPMENT

Candidate Number: 215717

Abraham Rey

Introduction

In this report I will be detailing the development of my web site for the introduction to multimedia module, created using HTML5 [2], CSS [10] and Java script. It will demonstrate how I have implemented the animation, how I created the website and any additional content that I have added. I will be using the guidelines in HCI to help make my website [1].

Website

Colours

I chose white background for the navigation bar and a slightly darker white for the body background, this keeps the website nice and simple and is easy to see the text without straining the eyes too much (e.g. using green for text and blue as the background wouldn't be particularly 'nice' to view). I tried to use a colour palette to assist me in choosing the colours [4], I ended up using a black colour for when the user hovers over one of the navigation buttons and a mustard yellow to indicate what page they are on; these colours have great contrast with white. I also made sure the text switched colours (to white) when the user hovered over the buttons so that the text can still be read since the background was black.

Navigation System

At first, I was going to use a sidebar for the navigation bar, but I realised it takes too much space, so it wasn't a good design choice and I also noticed that most websites had the bar at the top of the page such as Facebook, Google, reddit etc. Creating a horizontal navigation bar allowed me to create a drop-down menu for the coursework section as there were sub sections available for it (labs and the report). I made sure this was made consistent by making the same navbar on every page.

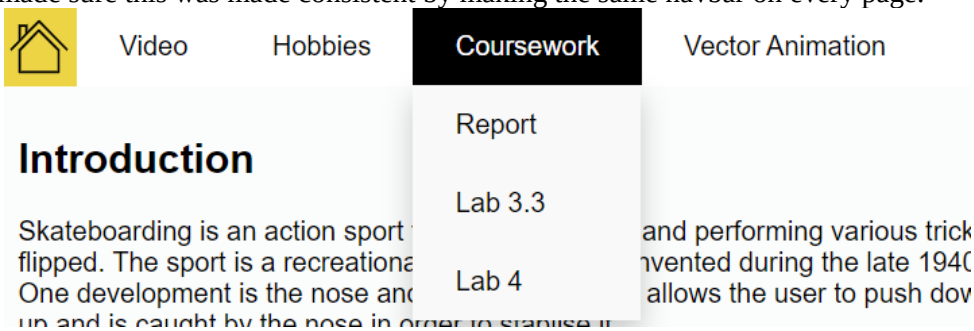

Figure 1: Home Page showcasing horizontal navigation bar and dropdown menu

I then implemented a sticky navbar [8] so that when the user scrolls down, the navbar is still available to use. This allows the user to easily navigate at all times, instead the user having to scroll all the way up to reach the navbar. I used a java script file that set the navbar to stick when the user scrolls down and offsets the page.

In figure 1, you can also see that I used a home icon [5] for the home page which was a svg element, which meant if you zoomed in, the resolution is kept at high quality. Also shown in the picture, I have used a colour to represent which page the user is on.

I also implemented a back button in java script which used the "history.back()" method [9] that was referenced to in the html file. The back button had an animation to it where, when hovering over it, it drops a shadow behind the button [7].

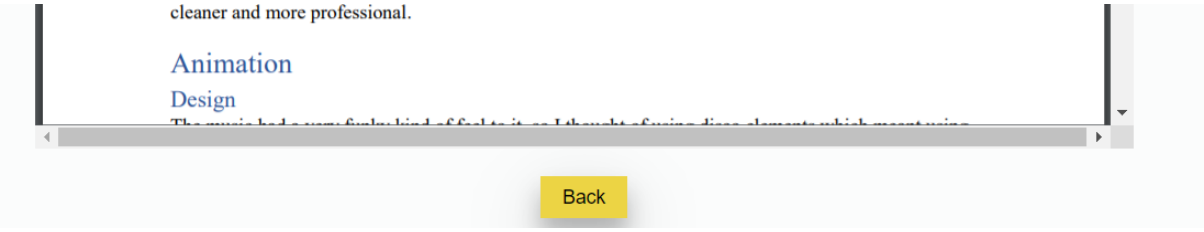

Figure 2: Showing the shadow behind the back button

Text

I decided to use a clean font than doesn't look old but looked more modern as most websites use more of rounded fonts such as Google; which is in the sans-serif family. Using this made the website cleaner and more professional.

I optimised the text by making the width smaller so that the user can easily read it.

Animation

Design

The music had a very funky kind of feel to it, so I thought of using disco elements which meant using lots colours and a disco floor. In the beginning of the music the voice over says '1234' and I thought I could sync an animation with that. As soon as the countdown completes, I thought of using a disco ball that would animate from the top and come into the frame.

Development

I used the raphael.js [3] library to create the animations and shapes in the animation and used some other functions.

Working out the rhythm

I used a stopwatch to work out how many bars there were per minute, which was about 33, then I found how many seconds there were per bars which ended up being 1.81 seconds per bar. This value allowed me to sync the animations to the rhythm.

Implementations

I used the setTimeout() and setInterval() functions to create the timings for each part of the animation.

```
function runAnimation()
{
    audio.play();
    oneBeat();
    setTimeout(generateFloorColours, 1818);
    setTimeout(generateFloorColours, 2272);
    setTimeout(generateFloorColours, 2726);
    setTimeout(generateFloorColours, 3180);
    setTimeout(intervalColours, 3634);
    setTimeout(twoBeat, 454);
    setTimeout(threeBeat, 909);
    setTimeout(fourOneHalfBeat, 1136);
    setTimeout(fourTwoHalfBeat, 1363);
    setTimeout(reduce, 1818);
    setTimeout(enlarge,2272);
    setTimeout(reduce, 2727);
    setTimeout(drop, 3181);
    setTimeout(moveDiscoBall, 3181);
    setTimeout(hideCircle, 3636);
    setTimeout(wavePairOne, 3636);
    setTimeout(wavePairTwo, 3693);
    setTimeout(wavePairThree, 3750);
    setTimeout(wavePairFour, 3807);
    setTimeout(intervalBars, 3807);
}
```

Figure 3: Timings for the animation

Shape changes were made at the start when the circle enlarges and reduces in size and the bars on the bottom resize randomly. This was done by using the “element.animate()” function in the raphael.js library where one the parameters were the final attributes for that element, I simply transformed the element by using the scaling string (e.g. s2 is increase by double the size).

At first, I implemented a colour generator which got a random colour from an array created at the beginning of the file, this allowed me to change colours of the bars and background. Colour changes were made while keeping in mind of the rhythm, so I created an interval for every four beats such that the colours on the disco floor and background changed every four beats. The background also only started changing after the voice says ‘1234’. I made the background animate to the colours (gradually changing) instead of going straight to the next colour, this gave a nice ambient look to the animation.

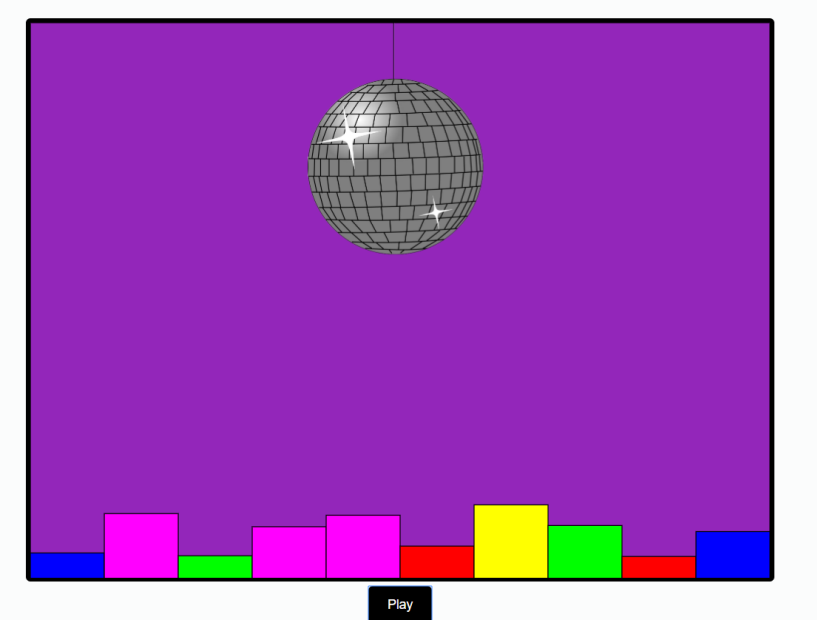

Figure 4: Showing the colours on the objects

I made the object move downwards after the '1234', towards the floor; this was done using animate function again. I used the 'ease-in' instead as it simulated gravity the best as I wanted to look like it fell. While this was happening, a disco ball png moves into the scene.

Upon making this animation I discovered a problem where the audio wouldn't play automatically in chrome, I found that there is a policy that doesn't allow certain media elements to auto-play when the page is loaded [6]. The animation will only play if the user has interacted with it, so I embedded a button and attached it to the java script file by using 'getElementByClassName' to retrieve it and execute a function when the button is clicked i.e. starting the animation.

Conclusion

Website conclusion

My website achieved the basic requirements with some extra functionality such as: the back button, sticky navbar, rollover buttons (the css changes when a cursor hover over it), and a shadow animation for the back button. I think I was successful in making a simple looking website that was easy for the user to use and navigate through, following most things in the HCI guidelines.

To improve my website, I would add some more functionality such as using a slideshow for displaying my lab work. I could have also allowed the user to stop the vector animation or mute it instead of just a play button with one function.

Vector animation conclusion

I have met the basic specifications: the audio and animation play when the button is clicked. A bitmap image (PNG) is animated into it, I have made rectangles and a circle animate (the circle enlarges and decreases in size and moves down, the rectangles change height randomly), I made sure that each animation was in the duration of each beat. I think it achieved my aim of making a relatively simple animation that sync to the soundtrack.

Enhancements that I would add to it would be to use my own drawn objects (via paths), that would give the animation more personality. I would have also animated the disco ball image more so that it appears to rotate by using multiple images that have different appearance (e.g. making the reflections

change when spinning). I would have also added more objects in the background, to show more animations such as rotations etc.

Bibliography

1. Guidelines in HCI - https://www.tutorialspoint.com/human_computer_interface/guidelines_in_hci.htm
2. HTML Tutorial - https://www.w3schools.com/html/default.asp
3. Raphael.js documentation - https://dmitrybaranovskiy.github.io/raphael/reference.html#Element.animate
4. Colour scheme - https://coolors.co/000000-ffffff-d5dfe5-c9b1bd-b49594
5. Home-symbol - https://www.flaticon.com/free-icon/home-symbol_16027
6. Auto-play policy - https://developers.google.com/web/updates/2017/09/autoplay-policy-changes
7. Button animation - https://www.w3schools.com/css/tryit.asp?filename=trycss_buttons_shadow
8. Sticky navbar - https://www.w3schools.com/howto/howto_js_navbar_sticky.asp
9. History method - https://www.w3schools.com/jsref/met_his_back.asp
10. CSS Tutorial - https://www.w3schools.com/css/