Creative Coding Creative Audio-Visual Application:

Cameron Tebbenham-Small, 18003324, Date: 16/03/2020

Department of Computer Science and Creative Technology
University of the West of England
Coldharbour Lane
Bristol, UK
cameron2.tebbenham-small@live.uwe.ac.uk

This report details the inspirations and processes involved in creating my Audio-Visual project for my Introduction to Creative Coding module, Year 1, Semester 2.

Table of Contents

URL link to piece.

Design:

During preliminary research, I had come up with three main ideas and jotted them down on paper:

1. I would create moving pipes as per the old windows screensaver.
    - An array of objects would be added to by clicking. These objects would move randomly across the screen, refreshing after a period of time.
    - 3D pipes screensaver (1080p) Youtube video, added by ronjamesscaresme. Available at https://www.youtube.com/watch?v=MKqrLGFoK9E
2. Expanding shapes that fill the screen.
    - This idea was taken from a couple p5 examples I found on button pressing Booleans.
    - On holding down a button, different shapes would expand, encompassing the screen, before another random shape taken from a list of 3-4 potential shapes would start expanding.
    - TimSherbert, Expanding Circle. Available from: https://editor.p5js.org/TimSherbert/sketches/ryC0T0caX
    - AndreasRef, button Pressed Boolean. Available from: https://editor.p5js.org/AndreasRef/sketches/oCb-MMETX

1. Wavelength that changes in shape and colours over time. Potentially link to sound.
    - The most ambitious idea as it is the one I have explored the least in class or outside.
    - McCarthy, L. Sine Wave. Available from: https://p5js.org/examples/math-sine-wave.html

However, this all changed once I began coding.

At first I took the code from the pipeline example on p5js.org (methio, Pipeline-Move) and tried to see how everything worked in the code. I took out elements I didn't like, changed others and ended up with a sketch that would simply place a few circles on the screen. I then turned to the Coding Train YouTube channel, something that became a mainstay of my research. I managed to create a sketch that would start with no objects, then add objects into an array and display them once the mouse had been pressed. I also used my knowledge and tutorials from the latest creative coding workshop – vectors, to replace the generic randomised movement and jittering with vector positions and velocity.

Most of my coding from this point onwards followed a repeating process:

- Inspiration occurs
- Research + find tutorials to learn how to implement idea
- Code idea and test until it successfully works.

For me, a lot of my inspiration came from my own desire to improve. The method or design of the idea would then be driven by whatever tutorials or examples I found. In some instances, this resulted in drastic changes. In others, I kept with my original idea. For instance, I decided to change my design from a simple mouse press, to a button press – taking inspiration from the sketches behind my second idea. Every time I press the button, a new object would appear. Eventually I reached a point where I could add and subtract ellipses with a click of a button. The ellipses would bounce off of the canvas edges and their colour could be changed through three RGB sliders.

Next, I decided to create a more visually different second stage. However, the second stage would still contain elements from the first stage. Using tutorials from Coding Train, I learnt how to remove an object from an array by clicking it. Furthering this, I built upon the session 10 worksheet (Meckin,D. Session 10) by adding a popping sound, an attractor and three new objects when I clicked on the stage 1 ellipses. I changed the colour of the attractors to black so that they wouldn't show up and added force to the ellipses. This allowed me to then tie the ellipses into the update functions for

the attractor so that adding attractors would alter the movement of the ellipses. Lastly, I added three smaller ellipses inside the ellipse constructor that would jitter inside the ellipse's diameter.

Evaluation:

Overall I felt that my project was very successful and I do feel proud of what I have created. During semester 1 I changed my ideas because I didn't understand the code or didn't know how to code certain things. During this semester 2 project however, perhaps because I had more time to work on the one p5 piece, I researched my way through each issue as they came up. First with adding randomly moving ellipses to a canvas, to creating RGB sliders and nearing the end of my project, implementing my canvases inside a website shell. Due to this, I have a much more thorough understanding of how p5.js works, and have frequently been helping other classmates with their own issues.

Throughout this piece of work, I was constantly thinking about ways in which to improve my project. At first, I was very worried about the coursework as I didn't know what I wanted to do or how to do the few things I could think of. However, over time, I came to enjoy the project and it became more of a passion project rather than a chore or something I had to do. Examples of this can be seen in my inclusion of RGB sliders – something I suddenly thought about creating and dedicated time learning more about how to link html variables with p5js.

What's more, I found myself imagining myself as a user of my interactive design, something I had been learning about in my Design Context and Web Design modules. I used version control through GitHub to manage my website and code and the server host to test the site on multiple devices. Due to this, I found myself creating a responsive canvas for phones and tablets, not just a computer.

One issue I had was using GitHub and local servers for version control as updating my code wasn't updating the code on the servers. I did manage to find a solution by going into dev tools on the website, right clicking refresh and emptying the cache to force a hard reset. Once I had found this out, using GitHub and local servers was a lot more enjoyable.

Making the website was very challenging as I was using a html5up template: (ajlkn. HTML5 UP, Dimension)

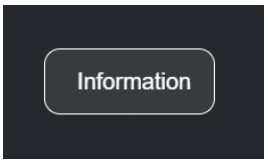
First, I had to merge my already made website with the new html5up template, as I had created a basic shell for my canvas. Due to the way the website is laid out, the template doesn't use separate pages. Thus, when I clicked off the currently open article, the image would close but the interactivity

project would still be running in the background. Through some comprehensive code reading, I managed to implement my reset function within the template's own JavaScript code. One of the biggest issues I encountered was a personal stretch challenge; adding my separate semester 1 sketches onto the same page. It took me all of a Sunday to get everything to work properly. In the end, I had to combine all my sketches into one code and convert the present code into three separate instances, then call each instance in a separate piece of html code. Despite how long it took, I am happy that I managed to struggle through this issue amongst all the others I encountered while learning new things outside of class.

Another challenge arose when I attempted to create a timed countdown to display information about the project whenever a button was clicked. Despite spending a long time researching and testing different methods to try and get it to work, I ultimately had to scrap it and replace it with a much more simple piece of code where if the SHIFT keyIsDown, text is displayed.

During this project I feel that I improved my troubleshooting and testing methods. An example of this is when I was testing the above feature, I discovered issues with the text sizes on different screen sizes and what's more, the SHIFT key can't be pressed on a phone – so phone or SmartScreen users would not be able to see the information following the first 10 seconds. Due to this, I decided to fix both issues at once by instead utilising the inbuilt alert feature.

```
function information() {
  alert('Press the add button to add a ball. \r
}
```




```
textAlign(CENTER, CENTER);
fill(255);
textSize(12);
if (millis() <= 1000){
  text('Press the add button to add a ball.
  text(timer, height/18, width/20);
  infoCircle();
}
if (frameCount % 60 == 0 && timer > 0) {
  timer --;
}
if (millis() >= 10000){
textAlign(CENTER, BOTTOM);
text(waveArray, 0, height/1.05, width);

}

if (keyIsDown(SHIFT)){
  alert('Press the add button to add a ball
}
```

Something I could maybe improve on is sticking to a single idea, with a pre-planned goal in mind. On the other hand, I much prefer stumbling my way through artistic creations as it leaves me open to new ideas along the way – such as adding the attractors and particles on button click. I didn't make as much use of my tutors as I could have, but I compensated by spending more than the recommended amount of time each week research.

References, inspiration and tutorials used:

ajlkn. HTML5 UP, Dimension. Available from: https://html5up.net/dimension

Creative Applications. Available from: http://www.creativeapplications.net/

HTML5 Canvas. Available from: https://www.w3schools.com/html/html5_canvas.asp

Input Range value Property. Available from:

https://www.w3schools.com/jsref/prop_range_value.asp

joemckaystudio. Multiple p5.js sketches in separate divs on one HTML page. Available form:

http://www.joemckaystudio.com/multisketches/

Meckin, D. Session 10. Available from:
https://github.com/davemeckin/Intro_to_Creative_Programming/blob/master/session_10/session_10.md

wjsandbe. mousePressed create objects from an array. Available from:
https://forum.processing.org/two/discussion/18231/mousepressed-create-objects-from-an-array

P5js sources:

McCarthy,L. P5js. Available from: https://p5js.org/

- https://p5js.org/reference/#/p5/createSlider
- https://p5js.org/reference/#/p5/mouseClicked
- https://p5js.org/reference/#/p5/mousePressed
- https://p5js.org/examples/objects-array-of-objects.html
- https://p5js.org/examples/math-sine-wave.html
- https://p5js.org/examples/instance-mode-instantiation.html
- https://p5js.org/examples/math-sine-wave.html

TimSherbert, Expanding Circle. Available from:

https://editor.p5js.org/TimSherbert/sketches/ryC0T0caX

AndreasRef, button Pressed Boolean. Available from:

https://editor.p5js.org/AndreasRef/sketches/oCb-MMETX

Methio. Pipeline-Move. Available from: https://editor.p5js.org/methio/sketches/MAo4isGMN

2sman. Basic timer. Available from: https://editor.p5js.org/2sman/sketches/ry3eS_cgG

Denaplesk2. Timer. Available from: https://editor.p5js.org/denaplesk2/sketches/BkNW2NCkM

Denaplesk2. Start Button Timer. Available from: https://editor.p5js.org/denaplesk2/sketches/B1taLQvxG

3D pipes screensaver (1080p) Youtube video, added by ronjamesscaresme. Available at https://www.youtube.com/watch?v=MKqrLGFoK9E

Stack Overflow forums. Available from:

- https://stackoverflow.com/questions/39288562/how-to-get-value-of-html-element-in-javascript/39288612
- https://stackoverflow.com/questions/29103818/how-can-i-retrieve-and-display-slider-range-value
- https://stackoverflow.com/questions/42001276/css-bootstrap-container-background
- https://stackoverflow.com/questions/53980755/how-to-auto-position-my-buttons-on-my-website-with-different-screen-resolution
- https://stackoverflow.com/questions/40948611/how-to-link-p5-js-setup-and-draw-with-html-canvas

WW3schools. Available from:

- https://www.w3schools.com/tags/att_div_align.asp
- https://www.w3schools.com/howto/howto_js_rangeslider.asp
- https://www.w3schools.com/tags/att_div_align.asp

YouTube sources:

3D pipes screensaver (1080p) Youtube video, added by ronjamesscaresme. Available at https://www.youtube.com/watch?v=MKqrLGFoK9E

The Coding Train Youtube channel. Available from: https://www.youtube.com/channel/UCvjgXvBlbQiydffZU7m1_aw

Coding Challenge #66: JavaScript Countdown Timer. The Coding Train. Available from: https://www.youtube.com/watch?v=MLtAMg9_Svw

Cameron Tebbenham-Small, 18003324

html5up website template:

Dimension by HTML5 UP

html5up.net | @ajlkn (twitter)

Free for personal and commercial use under the CCA 3.0 license (html5up.net/license)

- https://html5up.net/dimension

Credits:

Demo Images:

Unsplash (unsplash.com)

Icons:

Font Awesome (fontawesome.io)

Other:

jQuery (jquery.com)

Responsive Tools (github.com/ajlkn/responsive-tools)