Weekly Logs and Individual Report

Web Technology

Adam Logan

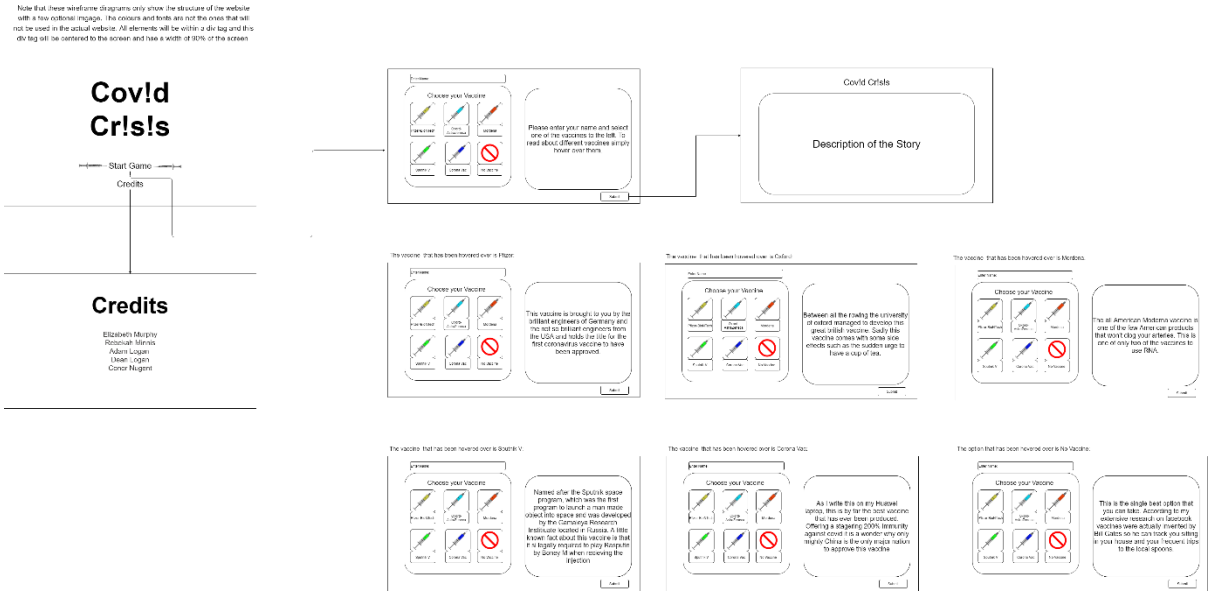
Week 1 (starting 22nd February) Log – Adam Logan

What have you done in the first week?

In the first week, I have meet with the team twice and we decided on a theme and the general structure for the project. We also created a rough timeline for the project and assigned jobs for each member of the group to complete for the next week.

My role this week was to create a few key wireframe diagrams to define the general structure of the key web pages and these can be seen below:

Start Screen and Vaccine pages:

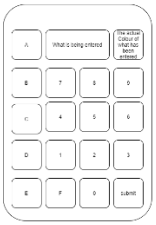


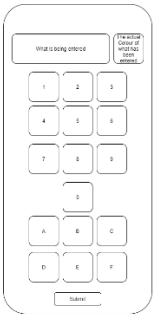
Leaderboard:

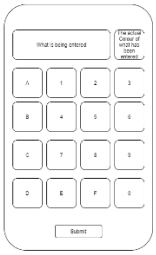
Final Screens:

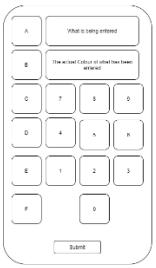
Note that these wireframe diragrams only show the structure of the website with a few optional imgage. The colours and fonts are not the ones that will not be used in the actual website. All elements will be within a div tag and this div tag will be centered to the screen and hae a width of 90% of the screen

Several designs for the colour pad (we decided to go with the design were the colours were used as the buttons):









I was playing an escape room game and they randomised the layout of the keypads they used to add a bit of difficulty. Do you think we should do that as well?

Below is when the user picks a colour corresponding to the hex code which they have discovered:

What will you undertake over the next week?

Over the next week I intend to create the HTML & CSS for the colour pad page and the fuse box web page. If there is time, I then intend to implement some JavaScript to these pages and to implement background music for the game.

Are there any issues currently impacting your progress on the project?

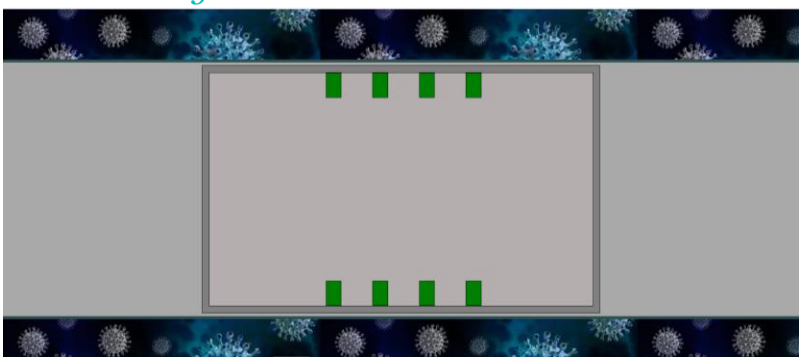
There are no current issues with the project.

Week 2 (starting 1st March) Log – Adam Logan

What have you done since your last post?

Since my last post I have meet with my team several times discussing the progress of the project. I have created the html and css for the fuse box, I have also created the html, css and some JavaScript for the colour pad and finally I have created the html and css for the basement page with a working torch effect.

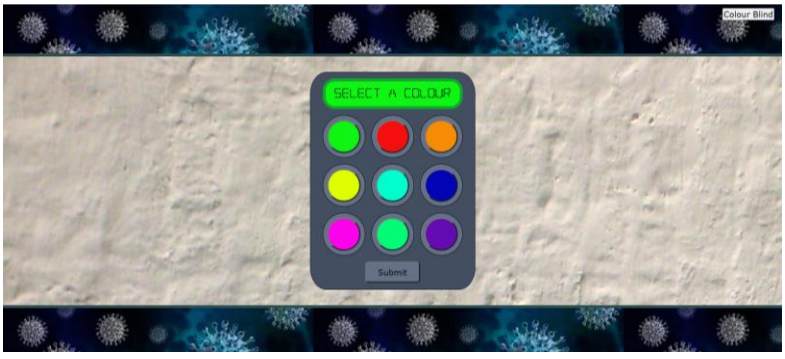
Fuse Box Page



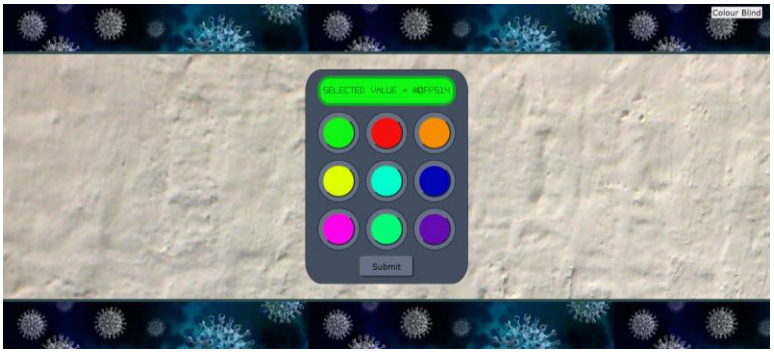
Colour Pad

When page is first loaded:

When the first (green) button is selected:

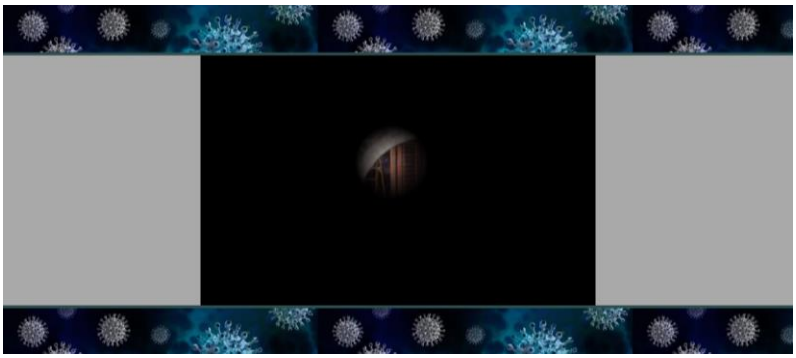


When the ‘submit’ button is pressed after this:



When the ‘Colour Blind’ button is pressed, and the second (red) button is selected:

Basement



What will you undertake over the next week?

Over the next week I will continue tweaking the pages shown above to improve the look of the pages and attempt to change all properties measured in pxs to %s to be more accommodating for different screen sizes. I will also attempt to implement a rough version of the wire game and to find images for the wire game.

Are there any issues currently impacting your progress on the project?

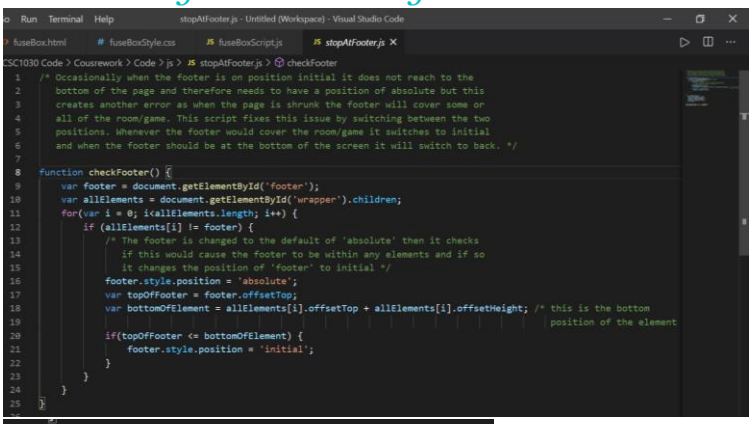
The current issue that is impacting my progress is being able to find appropriate images for the project and there has been several bugs in relation to the torch css effect which has been resolved but took a considerable amount of time to fix.

Week 3 (starting 8th March) Log – Adam Logan

What have you done since your last post?

Since my last post I have meet with my team several times discussing the progress of the project. I have changed the width and height of all elements to percentages as this allows a consistent layout across all screens. I have also added some code to prevent the colour pad to look squished when the screen size is below a certain point. I have also fixed a small bug within the layout code through JavaScript. I have also started working on connecting the wires for the fuse box game and I have experimented with background music for the game.

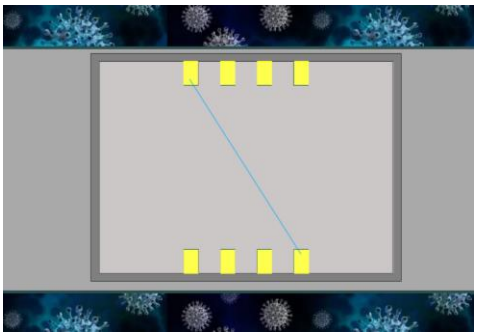
General Layout Footer Bug



```
/* Occasionally when the footer is on position initial it does not reach to the
   bottom of the page and therefore needs to have a position of absolute but this
   creates another error as when the page is shrunk the footer will cover some or
   all of the room/game. This script fixes this issue by switching between the two
   positions. Whenever the footer would cover the room/game it switches to initial
   and when the footer should be at the bottom of the screen it will switch to back. */

function checkFooter() {
    var footer = document.getElementById('footer');
    var allElements = document.getElementById('wrapper').children;
    for(var i = 0; i<allElements.length; i++) {
        if (allElements[i] != footer) {
            /* The footer is changed to the default of 'absolute' then it checks
               if this would cause the footer to be within any elements and if so
               it changes the position of 'footer' to initial */
            footer.style.position = 'absolute';
            var topOfFooter = footer.offsetTop;
            var bottomOfElement = allElements[i].offsetTop + allElements[i].offsetHeight; /* this is the bottom
                                                                                              position of the element
            if(topOfFooter <= bottomOfElement) {
                footer.style.position = 'initial';
            }
        }
    }
}
```

```
function getHeight(element) {
    element.style.visibility = "hidden";
    document.body.appendChild(element);
    var height = element.offsetHeight + 0;
    document.body.removeChild(element);
    element.style.visibility = "visible";
    return height;
}

window.addEventListener('resize', checkFooter);
window.addEventListener('load', checkFooter);
```

Rough Wire Game



What will you undertake over the next week?

Over the next week I will continue working on the fuse box game and hopefully have this completed. I will also attempt to record the time taken on each of my pages to be stored for the leaderboard. I will also attempt to implement the torch effect onto the fuse box and maybe implemented some sound effects on these pages.

Are there any issues currently impacting your progress on the project?

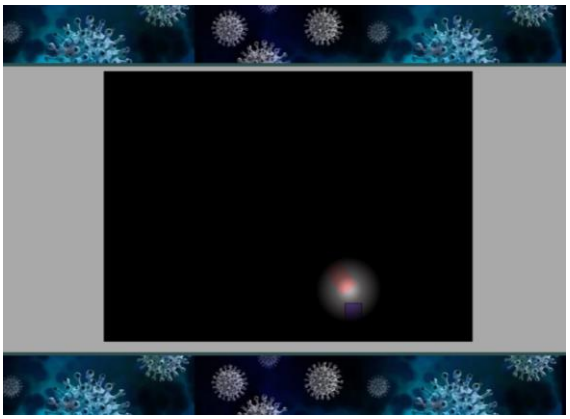
The main issue impeding the progress of the project is that there are a few bugs within relation to the fuse box game and it is difficult to find the correct royalty free images for the wire game.

Week 4 (starting 15th March) Log – Adam Logan

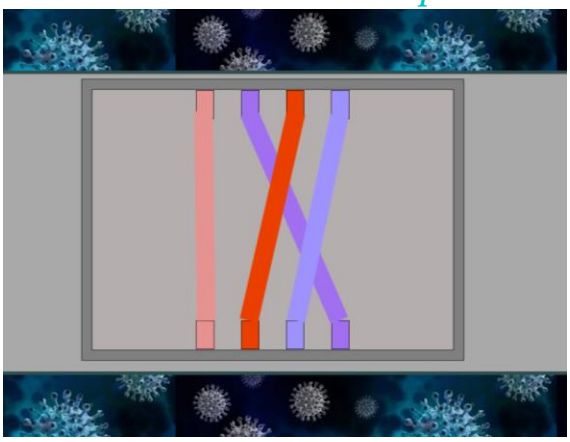
What have you done since your last post?

Since my last post I have meet with my team several times discussing the progress of the project. I have also fully implemented the fuse box game in which the colours of the wires are randomly generated. The position of which endpoint connects to another endpoint is also randomly generated on each node so each playthrough will be different. I have the torch effect implemented within the fuse box game and when all the wires are linked to the correct colour the 'lights' in the fuse box are turned on. The wires which the user drags across the screen will only stay on the screen once the user has connected it to the correct endpoint.

Fuse Box Game When First Loaded



Fuse Box Game When Complete



What will you undertake over the next week?

Over the next week I will attempt to fix the bugs discussed in the section below and as I did not have time this week I will once again attempt to record the time taken on each of my pages to be stored for the leader board. Another feature which I did not have time to implement is the sound effects for the project which I will attempt to implement this week. I will also add a coat stand to the basement for the user to find the second half of the hex code and go back to the colour pad and for the colour-blind option and display the names of the colours instead of the hex codes.

Are there any issues currently impacting your progress on the project?

The main issue impeding the progress of the project is that when the fuse box game is completed, I am not sure how to transition to the basement again and when I go back to the basement there is another bug that does not allow the torch effect to be turned off.

Week 5 (starting 22nd March) Log – Adam Logan

What have you done since your last post?

Since my last post I have meet with my team several times discussing the progress of the project. I have also implemented audio into the fuse box game. The sound occurs when the user has connected a wire. I have also implemented Elizabeth's text box code within my web pages to allow the story to progress through the game.

Another feature which I have implemented is that when the user completes the fuse box game it will display the fuse box without the torch effect, then at the side it will display a message through Elizabeth's text box and after two seconds the page will move back to the basement. When it has moved back to the basement the torch effect will be turned off and the user will be able to click on the coat stand and this currently displays an alert stating that the coat has been clicked.

I have also made the basement page keep its aspect ratio when the page is resized so that the buttons for the fuse box and the coat stand stay over the images.

Basement After the completion of the fuse box game

Code to Check if the lights are on

```
function checkLights() {
    if(sessionStorage.getItem('isLightsOn') == 'true') {
        document.getElementById('coatButton').addEventListener('click', coatClick);
        var basement = document.getElementById('basement');
        basement.className = 'lightsOn';

        var buttons = document.getElementsByClassName('buttonInBasement');
        for(var i=0; i<buttons.length; i++) {
            buttons[i].style.cursor = 'initial';
        }

        var message = document.getElementById('textBox');
        message.innerHTML = "NAME: Now the lights are back on I can find that note!";
    }
}
```

Code to keep aspect ratio

```
function keepAspectRatio() {
    var basement = document.getElementById('basement');
    var width = window.innerWidth;
    var height = window.innerHeight;
    aspectRatio = width/height;
    if(aspectRatio < 2 || aspectRatio > 2.3) {
        basement.style.width = basement.offsetWidth + 'px';
        basement.style.height = basement.offsetHeight + 'px';
    } else {
        basement.style.height = '67%';
        basement.style.width = '40%';
    }
}
```

What will you undertake over the next week?

Over the next week I will attempt to fix the bug discussed in the section below. My team also attends merge all our web pages together to complete the escape room game.

Are there any issues currently impacting your progress on the project?

The current issue that is impacting the progress of the project is that occasionally when the user is connecting their second wire the cursor changes to a no entry symbol, and it does not show the user dragging the wire. It is only when the user lifts the mouse up the wire is shown and can be connected.

Easter and Week 6 (starting 29th March) Log – Adam Logan

What have you done since your last post?

Since my last post, I have fixed the bug mentioned within the previous log by calling the function 'e.preventDefault()'. I have also implemented individual timers on all my web pages to be displayed within the statistics page at the end of the game. I have also created two success screens for the game, one which is displayed if the user has selected no vaccine and another if the user has selected any other vaccine. Another feature which I have implemented is when the user resizes the screen the aspect ratio of the basement image will stay the same to prevent the coat button and the fuse box button from moving from their desired spot on the image.

I have inserted the group timer and the text boxes, which were developed by other team members, into my web pages. I also created a script to prevent the text box from displaying over the game area when the user has resized the web page. The text box is set to disappear when it collides with the fuse box game area and is set to just maintain its positioning on the screen when it collides with the basement game area or the colour pad game area.

My team also worked together to join all our separate web pages together and we all fixed any bugs that occurred during this process.

No Vaccine End Screen

Vaccine End Screen

What will you undertake over the next week?

As the project has been completed there is nothing to complete over the next week.

Are there any issues currently impacting your progress on the project?

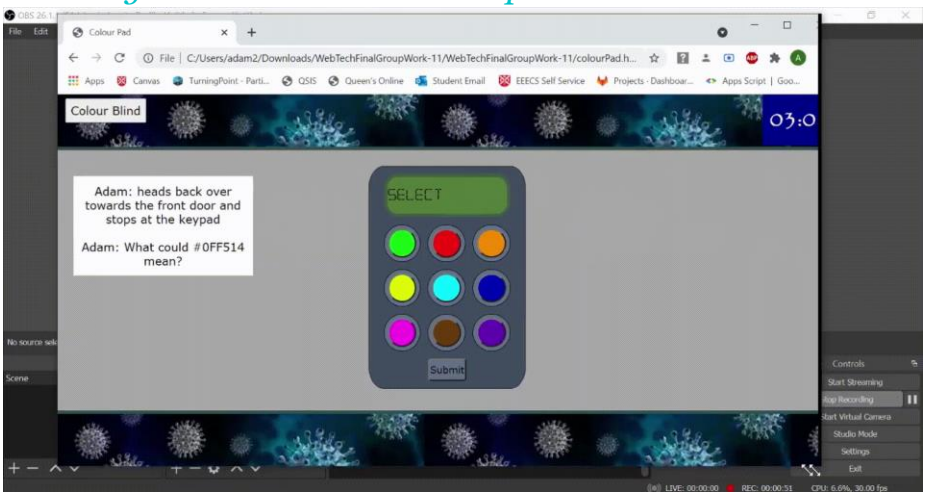
As the project has been completed there is nothing impacting the progress of the project.

Individual Report – Adam Logan

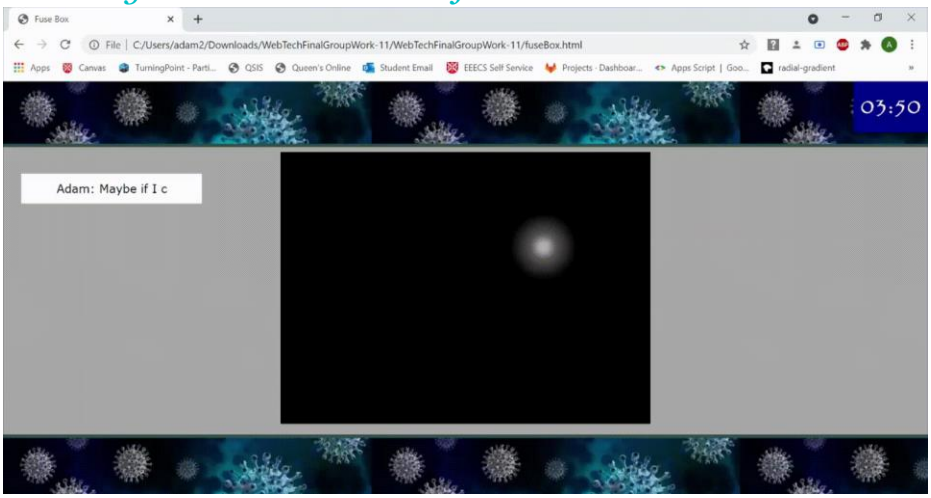
Resizing Screen Elements

I have created two scripts which are called ‘stopAtFooter.js’ and ‘textBoxSizing.js’ and these scripts are used on all my web pages. These scripts allow the user to resize the web page with the user interface still being friendly. The ‘stopAtFooter.js’ script changes the ‘position’ property of the footer to prevent a bug in which the footer would not always be completely at the bottom of the web page at all times. This bug is described in more detail in a comment within the script and in my week 9 log. The second script ‘textBoxSizing.js’ just prevents the text box to appear over the main game. On both the colour pad and the basement this just stops the game from moving any further but on the fuse box it makes the text box disappear.

Colliding with text box in colour pad



Colliding with text box in the fuse box



Basement Web Page

Within the basement web page I have fully developed the torch effect, the positioning of both the coat button and the fuse box button, the individual basement timer (not to be confused with the group timer which was handled by different team members) and the switching on/off the lights within the basement using session storage. The basement timer counts the time in which the user spends on the basement web page and the fuse box web page and is displayed to the user at the end of the game within the statistics page.

In relation to the resizing of this web page I have created it where the image of the basement will stay a consistent aspect ratio and therefore the two buttons will always be over the same spot within the image. I also created the basement image by merging three images (basement image, coat stand image and the fuse box image) together.

The scripts that I developed for this web page are generalLayout.css, basement.css, torchEffect.js, basementScript.js, textBoxSizing.js and stopAtFooter.js.

Torch Effect CSS

```
.torchEffect {
  cursor: none;

  background-image: radial-gradient(
    circle at var(--cursorX) var(--cursorY), /* this specifies that the
                                                shape of the gradient is
                                                a circle and the center of
                                                the circle should be placed
                                                at the position of circle*/
    /* The brightness of the torch can be changed by changing the opacity of the following values.
       more 'light' layers can be added to provide a better 'blur' effect around the cursor */
    rgba(0,0,0,0) 1vh,
    rgba(0,0,0,.5) 3vh,
    rgba(0,0,0,1) 8vh /* This is the outer layer of the torch and is the dimmest layer */
  ), url('../images/basement3.png');
}
```

This is code is only specific for the basement and is simply taken out for the fuse box.

Torch Effect JavaScript

This is called every time the user has their mouse over the torch area:

```
function updateTorch(e){
    var y = e.pageY - this.offsetTop;
    var x = e.pageX - this.offsetLeft;

    document.documentElement.style.setProperty('--cursorX', x + 'px');
    document.documentElement.style.setProperty('--cursorY', y + 'px');
}
```

These are the CSS custom properties that store the position of the

Keeping Aspect Ratio of Basement Image

```
function keepAspectRatio() {
    var basement = document.getElementById('basement');
    var width = window.innerWidth;
    var height = window.innerHeight;
    aspectRatio = width/height;
    if(aspectRatio < 2 || aspectRatio > 2.3) {
        basement.style.width = basement.offsetWidth + 'px';
        basement.style.height = basement.offsetHeight + 'px';
    } else {
        basement.style.height = '67%';
        basement.style.width = '40%';
    }
}
```

This is the range of the aspect ratio that is acceptable and that keeps the coat and fuse box buttons in the same place. The reason for the range is the image does flicker when the user has the aspect ratio just on the edge.

Total Basement Time Code

```
function coatClick() {
    var lightsOffTime = sessionStorage.getItem('bTime');
    var fuseBoxTime = sessionStorage.getItem('fuseBoxTime');
    time = parseInt(time) + parseInt(lightsOffTime) + parseInt(fuseBoxTime);
    sessionStorage.setItem('bTime', time);
    clearInterval(timer);
    sessionStorage.setItem('gameCount',gameSecondCount);
    window.open('coatNote.html','_self');
}
```

The code which adds the time spent in the basement (including the fuse box)

Fuse Box Web Page

I also fully developed the wire game including the audio that is played when the wires are connected. The wire game is played by dragging from the connectors at the top to the

corresponding coloured connector at the bottom. When the user lets go of the wire the wire will disappear and the user will need to start dragging the wire from the beginning. I also implemented the torch effect within this web page. I also fully developed the random colour generator for this web page and developed the code to make sure that the colours of the wires are not similar.

The scripts that I developed for this web page are generalLayout.css, fuseBoxStyle.css, torchEffect.js, fuseBoxScript.js, rgbAndHexFunc.js, textBoxSizing.js and stopAtFooter.js.

The code to check if a similar colour has been used

```
for(var i = 0; i<connecters.length; i++) {
  var similar = true;
  /* the following do while loop will check if if the random colour
   chosen is similar (I have decided to say it is similar if the r,g and b
   values are within a 100 values of each other) to a colour already chosen */
  do {
    var hex = randHexCode();
    var rgb = hexToRgb(hex);
    // loops through each of the already chosen colours
    for(var i = 0; i<colours.length; i++) {
      var rgbOfExisting = hexToRgb(colours[i]);
      // loops through the rgb values to check if at least 1 value is not within the 100 range
      for(var j = 0; j<rgb.length; j++) {
        var difference = Math.abs(rgb[j] - rgbOfExisting[j]);
        if(difference < 100) {
          similar = true;
        } else {
          similar = false;
          break;
        }
      }
      /* if the colour chosen is found to be similar to one
      there is no need to check the rest as it cannot be used */
      if(similar) {
        break;
      }
    }
  } while(similar && colours.length != 0);
  connecters[i].style.backgroundColor = hex; // as the invisible
                                             // connectors will be used to trigger the event
  // the colour needs to be stored to make the drawn line the same colour
  colours.push(hex);
  connecters[i].style.opacity = 0; // needs to be made invisible again
  visibleConnectors[i].style.backgroundColor = hex;
  connecters[i].addEventListener('mousedown', startDrawing);
}
```

This loop checks the red, green, and blue values of the chosen colour to see if it has a difference of a 100 with the colours that have already been selected.

This continuously generates a new random colour until it generates a colour which is not similar to any previous colour.

Code to Check for Connection

```
function connectionMade() {
  if(ctx != null) {
    var rgbOfBackground = getValuesFromRGB(this.style.backgroundColor);
    var hexOfBackground = rgbToHex(rgbOfBackground);
    if(hexOfBackground != ctx.strokeStyle) {
      endDrawing();
    } else {
      connection = true;
      ctx = null;
      var audio = new Audio('audio/Buzzer1.mp3');
      audio.play();
      var currentID = canvas.id;
      if ((parseInt(currentID[1]) + 1) < 5) {
        var nextID = 'c' + (parseInt(currentID[1]) + 1);
        canvas = document.getElementById(nextID);
      } else {
        var torchBox = document.getElementById('torchBox');
        torchBox.style.opacity = 0;
        sessionStorage.setItem('isLightsOn', 'true');
        var text = sessionStorage.getItem('name')+': Now the lights are back on I ' +
        'can go look for my coat!';
        consoleText(text,document.getElementById('textBox'));
        setTimeout(function() {
          sessionStorage.setItem('fuseBoxTime', time);
          clearInterval(timer);
          sessionStorage.setItem('gameCount',gameSecondCount);
          window.open('basement.html', '_self');
        }, (text.length+1)*60);
      }
    }
  }
}
```

This is the code to play the play the sound effect.

To check if all the wires have connected I checked the ID of the previous canvas and if it ends in a value less than 5 then connections still need to be made. This happens as the ID of the canvas is names as ‘c[number of connections made]’.

The Code to Draw the Wires

```
function startDrawing(e) {
    ctx = canvas.getContext('2d');

    colour = this.style.backgroundColor;

    canvas.style.display = 'initial';
    canvas.style.zIndex = 1;

    var x = e.pageX - canvas.offsetLeft;
    var y = e.pageY - canvas.offsetTop;

    startingPoint = [x, y]; // stores the point at which the mouse was clicked

    e.preventDefault();
}
```

```
function continueToDraw(e) {
  // if the context (ctx) is null that means the user has not clicked on the page
  // and therefore should not be able to draw.
  if(ctx != null) {
    // the 2 lines below get the current position of the mouse and
    // tracks this as the mouse moves
    var x = e.pageX - canvas.offsetLeft;
    var y = e.pageY - canvas.offsetTop;

    ctx.clearRect(0, 0, canvas.offsetWidth, canvas.offsetHeight); /* clears the canvas every time the mouse
                                                                     moves and therefore there is only ever
                                                                     1 line of the canvas at one time */
    ctx.strokeStyle = colour;
    ctx.lineWidth = document.getElementsByClassName('torchEndPoint')[0].offsetWidth;

    ctx.beginPath(); // starts the drawing
    ctx.moveTo(startingPoint[0], startingPoint[1]); // starts the line of the position where the mouse is clicked
    ctx.lineTo(x, y); // draws the line to the current position of the mouse
    ctx.stroke(); // ends the drawing
  }
}
```

Colour Pad

For the colour pad I fully developed the accessibility option within this web page which is the colour-blind option. I also developed the colour pad itself, the individual timer for this page and the function to randomly pick one of these colours which is passed through the web pages that require the hex code.

The scripts that I developed for this web page are generalLayout.css, colourPadStyle.css, colourPadScript.js, rgbAndHexFunc.js, textBoxSizing.js and stopAtFooter.js.

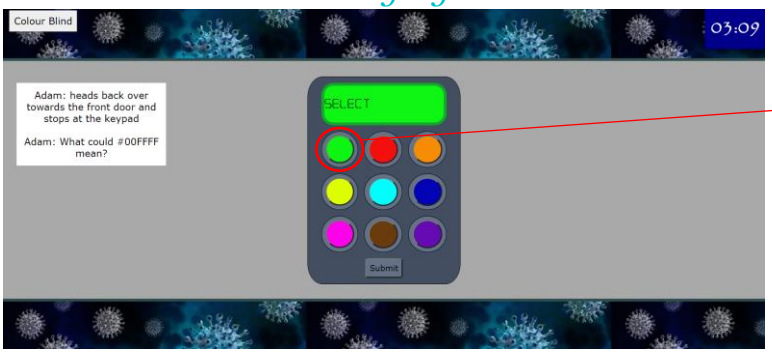
Generating Final Answer Code

```
function finalHexCode() {
    var colours = ['#0FF514', '#F50F0F', '#F78C05',
    '#DDFF00', '#00FFFF', '#0204B5',
    '#FC03EC', '#693B0D', '#640BB3'];

    var opt = Math.random() * 9;
    opt = Math.floor(opt);

    return colours[opt];
}
```

Colour Blind Code

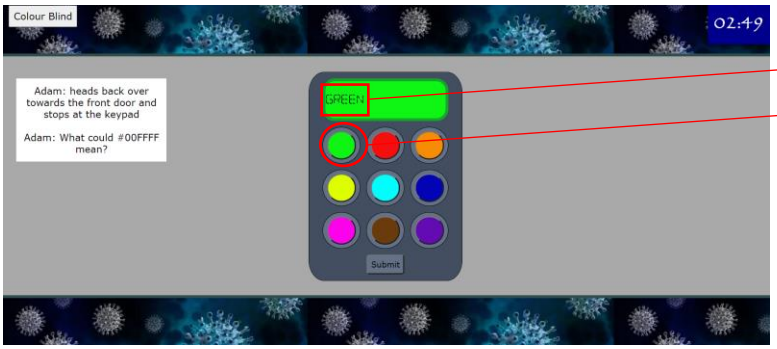
```
function colourButtonClicked() {
    var textScreen = document.getElementById('padScreen');
    var rgbOfButton = this.style.backgroundColor;

    textScreen.style.backgroundColor = rgbOfButton;

    var colourButtons = document.getElementsByClassName('colourButton');
    for(var i = 0; colourButtons.length > i; i++) {
        if(this == colourButtons[i]) {
            if (isColourBlind) {
                switch(colours[i]) {
                    case '#0FF514':
                        textScreen.innerHTML = 'Green';
                        break;
                    case '#F50F0F':
                        textScreen.innerHTML = 'Red';
                        break;
                    case '#F78C05':
                        textScreen.innerHTML = 'Orange';
                        break;
                    case '#DDFF00':
                        textScreen.innerHTML = 'Yellow';
                        break;
                    case '#00FFFF':
                        textScreen.innerHTML = 'Cyan';
                        break;
                    case '#0204B5':
                        textScreen.innerHTML = 'Blue';
                        break;
                    case '#FC03EC':
                        textScreen.innerHTML = 'Pink';
                        break;
                    case '#693B0D':
                        textScreen.innerHTML = 'Brown';
                        break;
                    case '#640BB3':
                        textScreen.innerHTML = 'Purple';
                        break;
                    default:
                        textScreen.innerHTML = 'Unknown Colour';
                        break;
                }
            } else {
                textScreen.innerHTML = 'Select';
            }
        }
        colourButtons[i].clicked = false;
    }
    this.clicked = true;
}
```

Colour Pad Screen Changing



This green button has been pressed and the colour pad screen has changed to the same colour as the button.

Colour Blind Button



The colour blind button has been pressed and then the green button has been pressed. As you can see the colour pad screen has turned green and the message on the colour pad screen has changed to 'GREEN' to tell the user the colour that they have selected.

Success End Screens

I fully developed both the success end screens (not including the escape video) and I implemented the code to decide which success screen to display using session storage which can be seen below.

The code which decides the end screen

```
function checkVaccine(){
  if(sessionStorage.getItem('vaccineSelected') == 'No Vaccine') {
    window.open('noVaxScreen.html','_self');
  } else {
    window.open('successScreen.html','_self');
  }
}
```