<table>
<thead>
<tr>
<th>Unit</th>
<th>Module number and description</th>
<th>Time in hours</th>
<th>Electives</th>
<th>Core concepts and ideas</th>
<th>Assess no.</th>
<th>Assessment technique, description and conditions</th>
<th>Dimensions</th>
</tr>
</thead>
</table>
| 4 (B) | **Module 5: full stack webapps**  
Design and develop Web Apps  
**Hardware**  
- server requirements to host published website  
- porting to different devices, e.g. personal computers, gaming consoles, portable devices, robotic devices and components  
- Screen resolutions  
- Type of input and output devices need to be considered when designing software, e.g. touch or keyboard  
- use and production of user guides and manuals used to problem-solve hardware-related issues  
**Software**  
- relationship between front/back end development exploring concepts such as local, cloud or server-side processing  
- issues relating to developing software to suit a range of devices, e.g. iOS and Android devices, consoles, robots, desktops and laptops  
- develop user guides/technical documentation and 'how-to' documents to assist clients  
- debug code and programs  
- beta and final testing  
- produce concept drawing, e.g. storyboarding, animation planning, story trees, digital illustrations, freehand sketches  
- purposes of websites determine design  
- reference and adherence to web accessibility guidelines, World Wide Web Consortium (W3C) standards  
- navigation and content hierarchy to ensure easy access  
- production of a visual representation of a website showing navigation organisation of web pages and associated linked documents, images, audio and video files, e.g. file structures and naming conventions  
- identification of tools, methods and techniques for using different software  
- techniques for website development  
- techniques for web page design.  
**ICT in society**  
- health and safety procedures and techniques  
- ethical and legal issues — copyright, e.g. ethical access and use of video, audio and images, attributing sources, use of creative commons  
- emerging trends in website development | 28 | • Application development  
• Website production | 6 | Project  
Design and Develop Web App  
• Product component  
Web App  
Variable conditions  
• Written component  
Documentation of problem solving process  
Y11: 400–700 words  
Y12: 500–900 words  
**Extended response**  
Develop copyright policy document based on Australian Copyright Law  
• Written response  
Policy document Y11: 500–800 words  
Y12: 600–1000 words | • Knowing and understanding  
• Analysing and applying  
• Producing and evaluating |
| | | | | | 7 | | |
Knowing and understanding

<table>
<thead>
<tr>
<th>W.A.L.T</th>
<th>W.I.L.F</th>
</tr>
</thead>
<tbody>
<tr>
<td>identification and explanation of software and hardware requirements related to ICT problems</td>
<td></td>
</tr>
<tr>
<td>identification and explanation of the use of ICT in society.</td>
<td></td>
</tr>
</tbody>
</table>

Analysing and applying

<table>
<thead>
<tr>
<th>W.A.L.T</th>
<th>W.I.L.F</th>
</tr>
</thead>
<tbody>
<tr>
<td>analysis of ICT problems to identify solutions</td>
<td>answers to the challenges</td>
</tr>
<tr>
<td>communication of ICT information to an audience using visual representations and language conventions and features</td>
<td></td>
</tr>
<tr>
<td>application of software and hardware concepts, ideas and skills to complete tasks in ICT contexts.</td>
<td>answers to the challenges</td>
</tr>
</tbody>
</table>

Producing and evaluating

<table>
<thead>
<tr>
<th>W.A.L.T</th>
<th>W.I.L.F</th>
</tr>
</thead>
<tbody>
<tr>
<td>synthesis of ICT concepts and ideas to plan solutions to given ICT problems</td>
<td>answers to the challenges</td>
</tr>
<tr>
<td>production of solutions that address ICT problems</td>
<td></td>
</tr>
<tr>
<td>evaluation of problem-solving processes and solutions, and recommendations made.</td>
<td></td>
</tr>
</tbody>
</table>

References:

- [https://www.khanacademy.org/computing/computer-programming/html-css](https://www.khanacademy.org/computing/computer-programming/html-css)
- [https://www.codecademy.com/learn/build-interactive-websites](https://www.codecademy.com/learn/build-interactive-websites)
- [https://www.w3schools.com/js/js_htmldom.asp](https://www.w3schools.com/js/js_htmldom.asp)
- [https://www.w3resource.com/javascript-exercises/javascript-dom-exercises.php](https://www.w3resource.com/javascript-exercises/javascript-dom-exercises.php)

**Intro to HTML**

Work your way through the Khan Academy tutorial, [Intro to HTML](https://www.khanacademy.org/computing/computer-programming/html-css), recording your answers to the challenges in your onenote area.

**Intro to CSS**

Work your way through the Khan Academy tutorial, [Intro to CSS](https://www.codecademy.com/learn/build-interactive-websites), recording your answers to the challenges in your onenote area.
### Knowing and understanding

<table>
<thead>
<tr>
<th>W.A.L.T.</th>
<th>W.I.L.F.</th>
</tr>
</thead>
<tbody>
<tr>
<td>identification and explanation of software and hardware requirements related to ICT problems</td>
<td></td>
</tr>
<tr>
<td>identification and explanation of the use of ICT in society.</td>
<td></td>
</tr>
</tbody>
</table>

### Analysing and applying

<table>
<thead>
<tr>
<th>W.A.L.T.</th>
<th>W.I.L.F.</th>
</tr>
</thead>
<tbody>
<tr>
<td>analysis of ICT problems to identify solutions</td>
<td></td>
</tr>
<tr>
<td>communication of ICT information to an audience using visual representations and language conventions and features</td>
<td></td>
</tr>
<tr>
<td>application of software and hardware concepts, ideas and skills to complete tasks in ICT contexts.</td>
<td></td>
</tr>
</tbody>
</table>

### Producing and evaluating

<table>
<thead>
<tr>
<th>W.A.L.T.</th>
<th>W.I.L.F.</th>
</tr>
</thead>
<tbody>
<tr>
<td>synthesis of ICT concepts and ideas to plan solutions to given ICT problems</td>
<td></td>
</tr>
<tr>
<td>production of solutions that address ICT problems</td>
<td></td>
</tr>
<tr>
<td>evaluation of problem-solving processes and solutions, and recommendations made.</td>
<td></td>
</tr>
</tbody>
</table>

### References:

- [https://www.khanacademy.org/computing/computer-programming/html-css](https://www.khanacademy.org/computing/computer-programming/html-css)
- [https://www.codecademy.com/learn/build-interactive-websites](https://www.codecademy.com/learn/build-interactive-websites)
- [https://www.w3schools.com/js/js_htmldom.asp](https://www.w3schools.com/js/js_htmldom.asp)
- [https://www.w3resource.com/javascript-exercises/javascript-dom-exercises.php](https://www.w3resource.com/javascript-exercises/javascript-dom-exercises.php)

### Intro to HTML

Work your way through the Khan Academy tutorial, [More HTML tags](https://www.khanacademy.org/computing/computer-programming/html-css), recording your answers to the challenges in your onenote area.

### Intro to CSS

Work your way through the Khan Academy tutorial, [CSS text properties](https://www.khanacademy.org/computing/computer-programming/css), recording your answers to the challenges in your onenote area.
<table>
<thead>
<tr>
<th>Knowing and understanding</th>
<th>W.A.L.T</th>
<th>W.I.L.F</th>
</tr>
</thead>
<tbody>
<tr>
<td>identification and explanation of software and hardware requirements related to ICT problems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>identification and explanation of the use of ICT in society.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Analysing and applying</td>
<td>analysis of ICT problems to identify solutions</td>
<td></td>
</tr>
<tr>
<td>communication of ICT information to an audience using visual representations and language conventions and features</td>
<td>answers to the challenges</td>
<td></td>
</tr>
<tr>
<td>application of software and hardware concepts, ideas and skills to complete tasks in ICT contexts.</td>
<td>answers to the challenges</td>
<td></td>
</tr>
<tr>
<td>Producing and evaluating</td>
<td>synthesis of ICT concepts and ideas to plan solutions to given ICT problems</td>
<td></td>
</tr>
<tr>
<td>production of solutions that address ICT problems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>evaluation of problem-solving processes and solutions, and recommendations made.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

References:
https://www.khanacademy.org/computing/computer-programming/html-css
https://www.codecademy.com/learn/build-interactive-websites
https://www.w3schools.com/js/js_htmldom.asp
https://www.w3resource.com/javascript-exercises/javascript-dom-exercises.php
https://www.udemy.com/course/javascript-dom-dynamic-web-interactive-content-boot-camp/

Intro to HTML
Work your way through the Khan Academy tutorial, [JS and the DOM](https://www.khanacademy.org/computing/computer-programming/html-css), recording your answers to the challenges in your onenote area.

Intro to CSS
Work your way through the Khan Academy tutorial, [DOM access methods](https://www.codecademy.com/learn/build-interactive-websites), recording your answers to the challenges in your onenote area.

Notes
Knowing and understanding

<table>
<thead>
<tr>
<th>WALT (We Are Learning To...)</th>
<th>WILF (What I'm Looking For...)</th>
</tr>
</thead>
<tbody>
<tr>
<td>identification and explanation of software and hardware requirements related to ICT problems</td>
<td></td>
</tr>
<tr>
<td>identification and explanation of the use of ICT in society.</td>
<td></td>
</tr>
</tbody>
</table>

Analysing and applying

| Analysis of ICT problems to identify solutions | |
| communication of ICT information to an audience using visual representations and language conventions and features | answers to the challenges
| application of software and hardware concepts, ideas and skills to complete tasks in ICT contexts. | answers to the challenges

Producing and evaluating

| Synthesis of ICT concepts and ideas to plan solutions to given ICT problems | |
| production of solutions that address ICT problems | |
| evaluation of problem-solving processes and solutions, and recommendations made. | |

References:

https://www.khanacademy.org/computing/computer-programming/html-css
https://www.codecademy.com/learn/build-interactive-websites
https://www.w3schools.com/js/js_htmldom.asp
https://www.w3resource.com/javascript-exercises/javascript-dom-exercises.php
https://www.udemy.com/course/javascript-dom-dynamic-web-interactive-content-boot-camp/

Intro to HTML
Work your way through the Khan Academy tutorial, DOM modification, recording your answers to the challenges in your onenote area.

Intro to CSS
Work your way through the Khan Academy tutorial, DOM events, recording your answers to the challenges in your onenote area.
Knowing and understanding

<table>
<thead>
<tr>
<th>W.A.L.T</th>
<th>W.I.L.F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identification and explanation of software and hardware requirements related to ICT problems</td>
<td></td>
</tr>
<tr>
<td>Identification and explanation of the use of ICT in society.</td>
<td></td>
</tr>
</tbody>
</table>

Analysing and applying

| analysing ICT problems to identify solutions |
| analysis of ICT problems to identify solutions |
| Communication of ICT information to an audience using visual representations and language conventions and features |
| Application of software and hardware concepts, ideas and skills to complete tasks in ICT contexts. |
| Generate Game with html, CSS and JS |

Producing and evaluating

| synthesis of ICT concepts and ideas to plan solutions to given ICT problems |
| Production of solutions that address ICT problems |
| Evaluation of problem-solving processes and solutions, and recommendations made. |
| Generate Game with html, CSS and JS |

References:

- [Khan Academy](https://www.khanacademy.org/computing/computer-programming/html-css-js)
- [Codecademy](https://www.codecademy.com/learn/build-interactive-websites)
- [W3Schools](https://www.w3schools.com/js/js_htmldom.asp)
- [FreeCodeCamp](https://www.freecodecamp.org/news/an-introduction-to-the-javascript-dom-512463dd62ec/)
- [W3Resource](https://www.w3resource.com/javascript-exercises/javascript-dom-exercises.php)
- [Udemy](https://www.udemy.com/course/javascript-dynamic-web-interactive-content-boot-camp/)
- [YouTube](https://www.youtube.com/watch?v=Y6Kubg4f7E&list=PLpc_YwcwbaaQCISTJisUGJenFEvUBzBd9)

Two-Up

Part 1: Introduction

Ref: [Two-Up](https://en.wikipedia.org/wiki/Two-up)

Part 2: Assets

1. Save these into the following folder structure: TwoUp/images/

2. Start an index.html file with this starter html

```html
<!DOCTYPE html>
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
</head>
```
Part 3: Creating The Markup

1. Under the <body> tag, create a new container, to contain everything else on the page:

   `<div class = "container">`

2. Now a heading:

   `<h1> Two Up</h1>`

3. Then containers for the images of the pennys.

   `<div class = "penny">
   <div class = "coin1">
   <p>Coin 1</p>
   </div>
   </div>`

   `<div class = "penny">
   <div class = "coin2">
   <p>Coin 2</p>
   </div>
   </div>`

4. Eventually, we will display the result, so add a container for that:

   `<div class = "result"></div>`

5. Lastly, we need to add a button to start the game:

   `<button class = "startBtn">Play</button>`

When you save and run, you should get this:
Remember, we haven't styled anything yet.

**Part 4: Adding Basic Styling**

1. We will forgo using an external style sheet and just add styles to our html script. Do this by putting the `<style></style>` tags under the `<head>` tag at the top of the page.

2. Add the following styles to the container class:
   - width: 650px;
   - margin 0 auto;
   - text-align: center;
   - padding: 25px;

3. Add the following styles to the penny class:
   - width: 350px;
   - height: 350px;
   - margin 100px 50px;
   - display: inline-block;

4. Style the `<p>` tag in .penny container with font weight 700, so that it stands out.
5. The coin1 and coin2 containers need to be styled to height and width 100%. This can be done by just styling the <div> tag within the .penny container.

```
.penny div {
  width: 100%;
  height: 100%;
}
```

It should look like this:

```
Two Up

Coin 1      Coin 2
```

6. Now let’s style the penny images, heads and tails.

```
.Heads{
  background-image: url('./images/heads.png');
}
.Tails{
  background-image: url('./images/tails.png');
}
```

7. Then we need to add them to the page by adding these classes to coin1 and coin2 containers.

```
<div class = "penny">
  <div class = "coin1 Heads">
    <p>Coin 1</p>
  </div>
</div>
```

```
<div class = "penny">
  <div class = "coin2 Tails">
    <p>Coin 2</p>
  </div>
</div>
```

It should look like this
Two Up

Lets now style each of the .Heads and .Tails classes so that the background image fits nicely in the container and doesn’t repeat.

Hint

```css
.Heads, .Tails {
  background-size: contain;
  background-repeat: no-repeat;
}
```

8. Now lest style the button by removing the border and adding some padding

Hint

```css
.startBtn {
  border: none;
  padding: 10px 100px;
}
```

Part 5: Creating the Animation

1. We want the coin images to move up and then down slightly to simulate a coin toss. To do this, we will create a css @keyframes that can then be applied to other classes. We will call it count-in and set the following keyframes:

```
0% don't move along y-axis
60% move up y-axis by -100px (0,0 is top left of screen)
100% move back to original position
```

Hint
@keyframes count-in {
  0% {
    transform: translateY(0);
  }
  60% {
    transform: translateY(-100px);
  }
  100% {
    transform: translateY(0);
  }
}

2. Now we use this @keyframes animation in a class we will call .count-in. We will run this for .7s, using the count-in keyframes, 3 times. This should raise and lower the coin image 3 times.

Hint

.ctin{
  animation: .7s .count-in 3;
}

Then we need to apply this .count-in class to coin1 and coin2 containers.

Hint

<div class = "penny">
  <div class = "coin1 Heads count-in">
    <p>Coin 1</p>
  </div>
</div>

<div class = "penny">
  <div class = "coin2 Tails count-in">
    <p>Coin 2</p>
  </div>
</div>

Test it out!

Part 6: Getting The Elements with Javascript

1. Add a <script></script> tag to the bottom, just above </body>

2. We will create some variables ( coin1Result, coin2Result) to hold the document elements for coin1 and coin2 classes (eg <div class = "coin1 Heads count-in">)

Hint

var coin1Result = document.querySelector('.coin1');
var coin2Result = document.querySelector('.coin2');

3. We also want to manipulate the start button (<button class = "startBtn"> Play</button>):

var startBtn = document.querySelector('.startBtn');

And get a result (<div class = "result"></div>):
Part 7: Adding Event Listeners

1. We can make events for when the start button is pressed. This will essentially toss the coins. Start by adding an `addEventListener` function to the `startBtn` variable that we created.

```javascript
startBtn.addEventListener('click', function(){
});
```

2. After the `startBtn` is clicked, we will just disable it, using `this.disabled = true;` We will also just blank out the result for now: `result.innerHTML = '';` For testing purposes, we will animate `coin1Result` and `coin2Result` with:

```javascript
coin1Result.className = "coin1 Heads count-in";
coin2Result.className = "coin2 Heads count-in";
```

```javascript
startBtn.addEventListener('click', function(){
  this.disabled = true;
  result.innerHTML = '';
  coin1Result.className = "coin1 Heads count-in";
  coin2Result.className = "coin2 Heads count-in";
});
```

Also, remove the count-in class animation from `coin1` and `coin2` containers.

```javascript
<div class = "coin1 Heads">
  <p>Coin 1</p>
</div>
```

```javascript
<div class = "coin2 Tails">
  <p>Coin 2</p>
</div>
```
**Test it out!**

Leg-up

```html
</div>

2. We now want to get a random coin toss. We do this by adding an 'animationend' addEventListener to document type.

```javascript
document.addEventListener('animationend', function(){
});
```

Re-enable the start button

```javascript
startBtn.disabled = false;
```

Make a new variable `coin1Land` to hold the result of a random selection between Heads or Tails. Remember, we created a `winStates` array.

```javascript
var coin1 = winStates[Math.floor(Math.random() * 2)];
```

Do the same for the other coin with `coin2Land`

```javascript
var coin2 = winStates[Math.floor(Math.random() * 2)];
```

We can now manipulate the value of the .coin1 and .coin2 classes with:

```javascript
coin1.className = "coin1 " + coin1;
coin2.className = "coin2 " + coin2;
```

**Hint**

```javascript
document.addEventListener('animationend', function(){
    startBtn.disabled = false;
    var coin1 = winStates[Math.floor(Math.random() * 2)];
    var coin2 = winStates[Math.floor(Math.random() * 2)];
    coin1Result.className = "coin1 " + coin1;
    coin2Result.className = "coin2 " + coin2;
});
```

---

**Part 8: Checking if Spinner Won**

The basic format of the game:

- Two heads means the spinner wins.
- Two tails means the spinner loses both their bet, and the right to spin.
- Odds ("one them") means the spinner throws again.


1. Lets make a new function called `getWinner` and pass in the parameters or values of `coin1` and `coin2`. 
```javascript
function getWinner(coin1, coin2){
    if (coin1 === 'Heads' && coin2 === 'Heads'){
        return "Spinner wins!";
    } else if (coin1 === 'Heads' && coin2 === 'Tails'){
        return "Spin again!";
    } else if (coin1 === 'Tails' && coin2 === 'Heads'){
        return "Spin again!";
    } else if (coin1 === 'Tails' && coin2 === 'Tails'){
        return "Spinner loses!";
    }
}

2. Remember we have `<div class = "result"></div>`. So we can now pass the `return` value to the .result class using `innerHTML`. Add this to the 'animationend' event listerer.

```javascript
result.innerHTML = getWinner(coin1, coin2);
```
2. Let’s also add the css rule for font so we can apply it to the <body>

```
Hint

body{
    font-family: 'Open Sans', sans-serif;
}
```

3. Now lets add some more style to the <body>:

```
body{
    font-family: 'Open Sans', sans-serif;
    background-color: brown;
    text-shadow: 0 0 8px rgba(0,0,0,0.6);
    color: seashell;
}
```

4. We could also update the container so that the gamespace stands out.

```
.container{
    width: 650px;
    margin 0 auto;
    text-align: center;
    padding: 25px;
    background-color: tan;
    box-shadow: 0 0 34px rgba(0,0,0,0.4);
}
```
5. Then we can style the start button:

```css
.startBtn{
  background-color: indianred;
  color: lemonchiffon;
  border: none;
  padding: 10px 100px;
}
```

6. Let's style the results:

```css
.result{
  margin-bottom: 25px;
  font-weight: 700;
  height: 50px;
  font-size: 2em;
}
```
### Phase 1: Problem identification

**Knowing and understanding**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Identification and</td>
<td>identification and explanation of software and hardware requirements related to ICT problems</td>
</tr>
<tr>
<td>Explanation of the</td>
<td></td>
</tr>
<tr>
<td>Problem</td>
<td>identification and clarification of the problem</td>
</tr>
<tr>
<td>Identification of the</td>
<td>identification of the hardware and software requirements related to the problem</td>
</tr>
<tr>
<td>Hardware Requirements</td>
<td></td>
</tr>
<tr>
<td>Identification of the use</td>
<td></td>
</tr>
<tr>
<td>of ICT in society</td>
<td>identification of the purpose and audience</td>
</tr>
<tr>
<td>• Identification of the</td>
<td>identification of WHS procedures</td>
</tr>
<tr>
<td>Purpose and Audience</td>
<td></td>
</tr>
<tr>
<td>Analysing and applying</td>
<td>analysis of ICT problems to identify solutions</td>
</tr>
<tr>
<td>Communication of ICT</td>
<td></td>
</tr>
<tr>
<td>Information to an audience</td>
<td>using visual representations and language conventions and features</td>
</tr>
<tr>
<td>Application of software and</td>
<td></td>
</tr>
<tr>
<td>Hardware concepts, ideas</td>
<td></td>
</tr>
<tr>
<td>and skills to complete</td>
<td></td>
</tr>
<tr>
<td>tasks in ICT contexts.</td>
<td></td>
</tr>
<tr>
<td>Producing and evaluating</td>
<td>synthesis of ICT concepts and ideas to plan solutions to given ICT problems</td>
</tr>
<tr>
<td>Production of solutions</td>
<td></td>
</tr>
<tr>
<td>that address ICT problems</td>
<td></td>
</tr>
<tr>
<td>Evaluation of problem</td>
<td></td>
</tr>
<tr>
<td>solving processes and</td>
<td></td>
</tr>
<tr>
<td>solutions, and recommendations made.</td>
<td></td>
</tr>
</tbody>
</table>

**Clarification of the Problem**

Restate the problem in your own words.

**Identification of Hardware Requirements**

a. Where will you host your finished webapp?

b. Will you be able to port your solution to different devices? e.g. personal computers, gaming consoles, portable devices, robotic devices and components

c. What screen resolutions will your solution be viewed on?

d. What type of input and output devices need to be considered when you are designing your software. e.g. touch or keyboard

**Identification of Software Requirements**

a. Will your solution be front or back end? Give details.

b. Will your solution be local, in the cloud or use server-side processing?

c. What issues will you have relating to developing software to suit a range of devices, e.g. iOS and Android devices, consoles, robots, desktops and laptops

d. Outline, in detail, the DOM and how to manipulate it.

**Identification of ICT in society**

a. Identify the purpose and audience for your solution.

b. Identify the health and safety procedures and techniques you will need to employ when developing your solution.

**Resources**

- Getting to know the ...
- JS the browser a...
- Create Interactiv...
### Knowing and understanding
- Identification and explanation of software and hardware requirements related to ICT problems.
- Identification and explanation of the use of ICT in society.

### Analysing and applying
- Analysis of ICT problems to identify solutions.
  - Analysis and examination of the problem requirements.
  - Identification of possible solutions.
- Communication of ICT information to an audience using visual representations and language conventions and features.
- Organisation of information.
- Application of software and hardware concepts, ideas and skills to complete tasks in ICT contexts.
- Application of concepts, ideas, knowledge, understanding and skills and procedures, related to developing and producing web apps with DOM, to complete tasks.

### Producing and evaluating
- Synthesis of ICT concepts and ideas to plan solutions to given ICT problems.
- Production of solutions that address ICT problems.
- Evaluation of problem-solving processes and solutions, and recommendations made.

### Problem Requirements
a. What will your solution need to be able to do?
b. What DOM techniques will you need to use?

d. Define Data and Functional Requirements, in terms of Input, Processing and Output.

### Solution Specification
a. Specify what you need to produce?
b. Identify the content and assets you have? What is their source?
c. Outline what the user interface and user experience will be like?
d. Define Data and Functional Requirements, in terms of Input, Processing and Output

e. What is the main algorithm that you will need?
Knowing and understanding

| Identification and explanation of software and hardware requirements related to ICT problems |
| Identification and explanation of the use of ICT in society. |

Analysing and applying

| Analysis of ICT problems to identify solutions |
| Communication of ICT information to an audience using visual representations and language conventions and features |
| Application of software and hardware concepts, ideas and skills to complete tasks in ICT contexts. |

Producing and evaluating

| Synthesis of ICT concepts and ideas to plan solutions to given ICT problems |
| Sketch solution plan an approach that produces the best solution to the problem |
| Production of solutions that address ICT problems |
| Produce solution and test the viability of the solution for the intended purpose |
| Evaluation of problem-solving processes and solutions, and recommendations made. |
| Use criteria to evaluate the effectiveness of the processes and solution in order to make recommendations |

**Sketch out what your webapp solution will look like.**
**Annotate to communicate how it will work**

**Plan the Solution**

- List the tasks that you need to complete, time needed and due date.

**Eg**

<table>
<thead>
<tr>
<th>Task</th>
<th>Time needed</th>
<th>Due date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compile the list of questions and answers</td>
<td>20 mins</td>
<td>29-5</td>
</tr>
<tr>
<td>Test Plan</td>
<td>20 mins</td>
<td>29-5</td>
</tr>
<tr>
<td>Make new metaverse experience</td>
<td>2 hrs</td>
<td>7-6</td>
</tr>
<tr>
<td>Test</td>
<td>20 mins</td>
<td>7-6</td>
</tr>
</tbody>
</table>

**Evaluation**

**Sketch Solution**

- Sketch out what your webapp solution will look like.
- Annotate to communicate how it will work

**Generate Solution**

**Test Solution**

- Create a test plan to probe the viability of your solution
- Run your test plan and revise any issues.

<table>
<thead>
<tr>
<th>Test Number</th>
<th>Test</th>
<th>Expected Results (What you think will happen?)</th>
<th>Actual Result (What actually happened?)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>start app</td>
<td>first screen will appear</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>press answer 1 button</td>
<td>Taken to correct response screen</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>press answer 2 button</td>
<td>Taken to correct response screen</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>press answer 3 button</td>
<td>Taken to correct response screen</td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>selfie</td>
<td>selfie uploaded</td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Evaluate Solution
Evaluate the problem solving process and the product by writing a report on the process of making the product, including:

- **process**
  - evaluate the application of your knowledge of software and hardware concepts, ideas and skills to complete the client’s request
    - What was the problem you were trying to solve? Did your product solve the problem?

  The problem was thoroughly/ effectively/ appropriately/ variably described from the user’s perspective because….
  The problem was thoroughly/ effectively/ appropriately/ variably/ solved within these recognized constraints: ….
  Existing solutions to similar problems were thoroughly/ discerningly/ effectively/ appropriately/ variably analyzed to identify possible solutions. The …. was the basis for the solution because….
  The relationships between users, solutions and the components of solutions in similar problems was thoroughly/ discerningly/ effectively/ appropriately/ variably understood. This was because of …

- evaluate the process of making the WEBAPP, including the difficulties you experienced and how they were overcome
  The solution to the problem was thoroughly/ discerningly/ effectively/ appropriately/ variably visualized with creative skill to represent and communicate ideas, by….
  Components of the solution were thoroughly/ discerningly/ effectively/ appropriately/ variably tested with conceptual models, using…

- **product**
  - evaluate the suitability of the product for the desired audience
    The solution was thoroughly/ discerningly/ effectively/ appropriately/ variably constructed, with the components of the preferred solution, using …

  - analyse the technical details of the product, including the style and format, image resolution and operation on the network
    The constructed solution was thoroughly/ discerningly/ effectively/ appropriately/ variably refined in response to alpha testing. The main refinement was…because…
    In comparison, the final solution was better/ more effective/ closer to the intention than the first attempt because …
    - Was using the problem solving process useful? Why/Why not?
    - Was your solution or product useful and valuable; did it solves the defined problem or meets the identified need?
    - Is your solution or product well crafted, striking, designed with a distinct style but still appropriate for the purpose?

- **make recommendations**
  - list and explain recommendations or advice you would provide for others as they prepare ICT solutions for clients.
    In future, it is recommended that the solution be refined further by:…
    This will ensure that….
    - If you had to solve this problem again, what would you do differently?