



COURSEWORK 1 - REPORT

Richard Cadelano

40315121

EDINBURGH NAPIER UNIVERSITY – MODULE TITLE
(SET08101)

Richard Cadelano
40315121@live.napier.ac.uk

Contents

Introduction	2
Software Design	3
Implementation	4
Critical Evaluation	8
Personal Evaluation.....	8
References	9

Introduction

The main task of this coursework will be to develop a website that will allow the user to choose between two different ciphers and encrypt a message. The website will be divided into three different pages:

- The Main Index.html page: This is the main website page, here the user will be able to: choose between the two different ciphers implemented, and read a brief explanation about Cryptography,
- The Caesar Cipher page: In this page the user will be able to: encrypt a message with the Caesar cipher implemented in the page, and read a brief explanation about the origins and how the cipher works,
- The Morse Code page: Here the user will be able to encrypt a message into Morse Code, and read a brief explanation about the creation of the Morse code and the telegraph.

Will be possible to navigate between pages thanks to the presence of animated buttons that will direct the user to the chosen cipher.

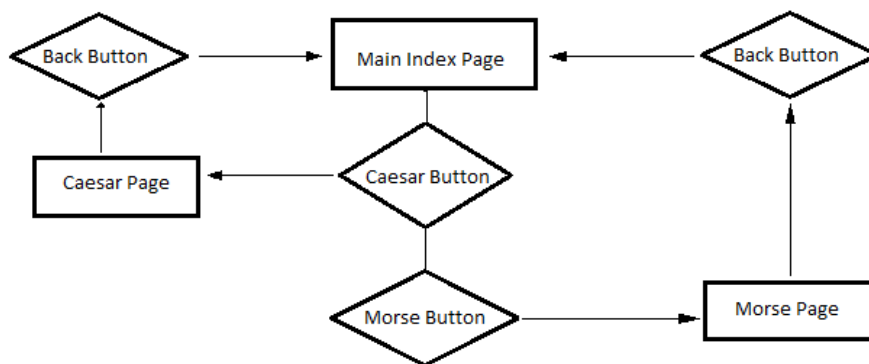
During this Report will be analysed the creation of the website, the HTML code of each page and the CSS code that determines the final appearance, the ciphers are implemented in the page thanks to a JavaScript function wrote apart and called in the code.

Software Design

Before the actual development of the webpages was necessary to draw a plan to clarify how actually realise it, which cipher develop and basically make a list of the requirements and code knowledge needed to commute an idea into a real design.

My personal approach started with an initial design on paper, where I drawn a draft with a basic page layout trying to describe how heading and paragraphs had to appear, and the space for any eventual image and page decoration. My original solution was originally implemented in one page where the user could choose between the ciphers through a dropdown menu, but at the end I have opted for the navigation between three pages.

The navigation the pages could be represented by the following scheme:



For each page were planned the following components:

- Heading: Using the font Impact with a size of 50px, capitalized and underlined,
- Paragraph: a short paragraph section where explains the history of the selected cipher and how it works, in the index page the paragraphs explain briefly the history of cryptography, font selected is Times New Roman with a size of 20px,
- Images: Some images with captions to enrich the page. For each image is provided a caption with font-style italic.
- A button: to navigate between pages.

Focusing on the ciphers side of the webpage were planned the following components:

- Input text-area: To insert the message to encode, for the Caesar cipher will be included also an input area to decide how many positions shift the letters,
- A button: to call the JavaScript function, named "Encodify Message",
- An output area: To visualise the encoded message.

The whole plan includes three pages plus connected three CSS files, one for each page and two JavaScript functions, one for each cipher.

Implementation

After the planning stage of the coursework it is time to focus on the code implementation, as mentioned in the Software Design paragraph each page is composed by: Headings, Paragraphs, and Images with captions.

In the following screenshots will be showed the appearance of each component and the coded used.

Headings:

Ciphers And Encryption

Heading visualised on browser.

```
h1{
  text-align: center;
  text-decoration: underline;
  text-transform: capitalize;
  font-family: Impact, Haettenschweiler, 'Arial Narrow Bold', sans-serif;
  font-size: 50px;
}
```

Here is possible to see the CSS code used to personalise the heading, it contains every attribute of the heading text.

Paragraphs:

Cryptography is a method to secure communications by translating a "plain text" message into an undecryptable message. Is a method used since the ancient times to protect secret conversations and provide security.

The first known example of Cryptography is represented by the use of hieroglyphics in the Ancient Egypt, where the "original message" was translated into that kind of text by scribes.

```
p{text-align: center;
  font-family: 'Times New Roman', Times, serif;
  font-size: 20px;
}
```

Here is possible to see how paragraphs appears in the browser and the CSS code used.

The Information contained in the paragraphs are referenced at the points: 4-5-6-7-8 of the References paragraphs.

Images and Captions:



Image visualised in the browser

Morse Telegraph

```
figure{
  float: center;
  width: 95%;
  text-align: center;
  font-style: italic;
  font-size: smaller;
}
```

```
<figure>
<p>
| | <figcaption>Morse Telegraph</figcaption>
</figure>
```

Here is possible to observe the CSS code (on the left), where is built a new class (figure) with all the attributes for the captions; on the right is possible to observe the code used to call the class and set the image and the caption text.

The code is referenced at the point 9 of the References paragraph.

Button:



Button visualised on the Browser

```
<style>
.button {
  display:inline-block;
  border-radius: 4px;
  background-color: darkturquoise;
  border: none;
  color: #FFFFFF;
  text-align: center;
  font-size: 28px;
  padding: 20px;
  width: 200px;
  transition: all 0.5s;
  cursor: pointer;
  margin: 5px;
}

.button span {
  cursor: pointer;
  display: inline-block;
  position: relative;
  transition: 0.5s;
}

.button span:after {
  content: '\00bb';
  position: absolute;
  opacity: 0;
  top: 0;
  right: -20px;
  transition: 0.5s;
}

.button:hover span {
  padding-right: 25px;
}

.button:hover span:after {
  opacity: 1;
  right: 0;
}
</style>
```

```
<div style="display:flex; justify-content:center;">
  <a class="button" style="vertical-align: middle" href="..\index.html"><span>Back</span></a></button><br>
</div>
```

Here is possible to see: the code implemented into the HTML file to create the button class (on the left), because it is an animated button, beside all the normal attributes, contained in the first group (background-color, text-align, etc), was necessary to implement also the animation attributes (span/after, hover span/after); on the right is possible to see how the button class is called inside a <div> to organise its position in the page.

The Button code is referenced in the point 3 of the References paragraph.

JavaScript functions:

```

function encode_cypher(){
    var plain_text = document.getElementById("message").value;
    var alphabet = "abcdefghijklmnopqrstuvwxyz";
    var fullAlphabet = alphabet + alphabet + alphabet;
    var cipherOffset = document.getElementById("offset").value;
    cipherOffset = (cipherOffset % alphabet.length);
    var cipherFinish = [];

    for(var idx=0; idx<plain_text.length; idx++){
        var letter = plain_text[idx];
        var upper = (letter == letter.toUpperCase());
        letter = letter.toLowerCase();

        input = alphabet.indexOf(letter);
        if(input == -1)
        {
            cipherFinish.push(plain_text[idx]);
        }
        else
        {
            var coded = ((input + cipherOffset) + alphabet.length);
            var nextLetter = fullAlphabet[coded];
            if(upper) nextLetter = nextLetter.toUpperCase();
            cipherFinish.push(nextLetter);
        }
    }
    document.getElementById("finish").innerHTML = cipherFinish.join("");
    return cipherFinish.join
}

```

Here is possible to observe the JavaScript function for the Caesar Cipher, as we can see it takes the plain text from the input-area (id= message in the html file), and the position to shift from the number input area (id= offset in the html file). The variable cipherFinish is an empty list where will be appended the result of the function (Encrypted Message).

Is present a for statement that will check if a Letter is on Uppercase or lowercase and will allow the cipher to recognise that and shift the letter without losing the case.

After we can observe the if statement that will check if the text area is empty or, if there is something, will shift the letters adding the result to the cipherFinish variable, that will be displayed in the browser.

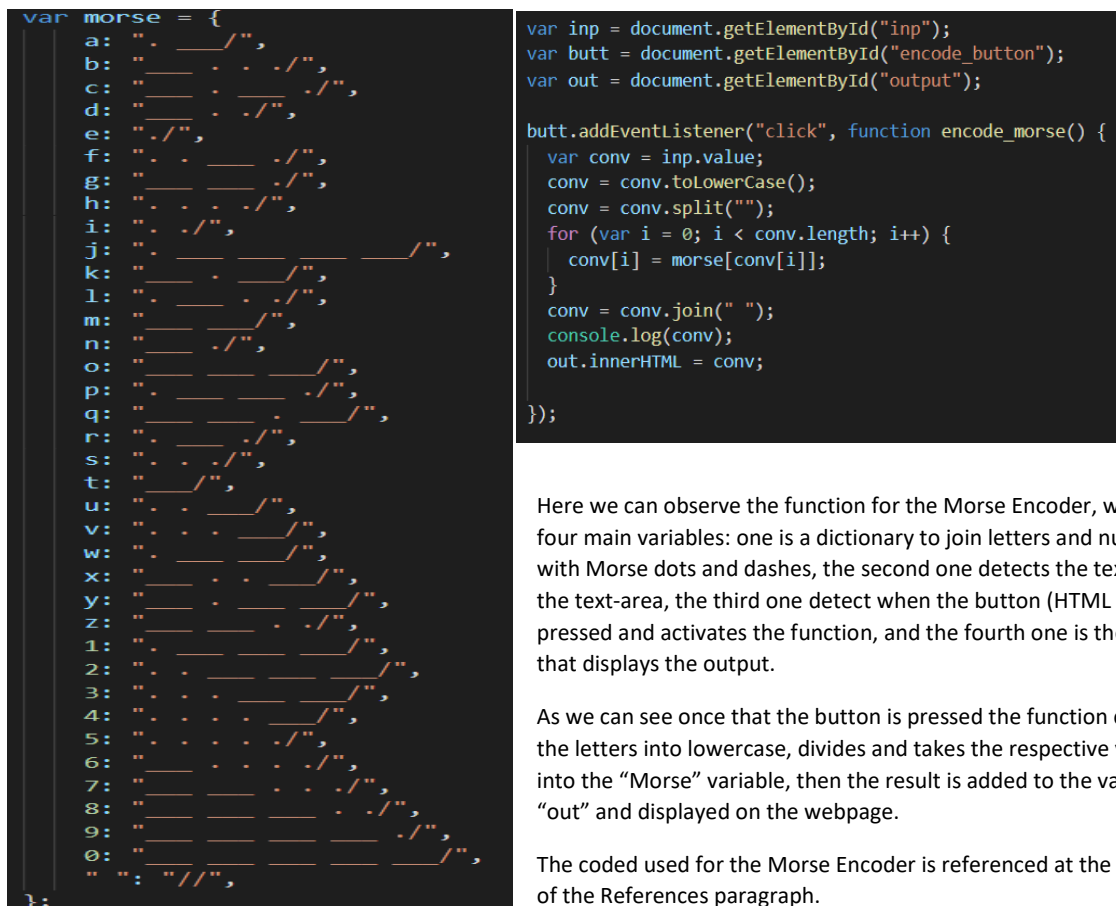
The code used for the Caesar Cipher is referenced at the point 2 of the References paragraph.

Example of Caesar cipher:

Insert Message	7	▼	Encodify Message
----------------	---	---	------------------

Message:

Puzlya Tlzzhnl



Here we can observe the function for the Morse Encoder, we have four main variables: one is a dictionary to join letters and numbers with Morse dots and dashes, the second one detects the text into the text-area, the third one detect when the button (HTML page) is pressed and activates the function, and the fourth one is the one that displays the output.

As we can see once that the button is pressed the function converts the letters into lowercase, divides and takes the respective value into the "Morse" variable, then the result is added to the variable "out" and displayed on the webpage.

The coded used for the Morse Encoder is referenced at the point 1 of the References paragraph.

Here an Example of Morse Encrypter:

Encode Message

Message:

. / _ _ . / . . . / . / . _ _ . / _ _ / // _ _ _ / . / . . . / . . . / . _ _ / _ _ _ . / . /

Text Areas:

```

<div id = "input_area"; style="display:flex; justify-content:center;">
  <textarea rows="1" cols="50" autofocus id="inp">Insert Message</textarea><br>
  <button type="button" onclick="encode_morse()" id="encode_button">Encode Message</button>
</div>
<div id="output_area">
  <p>Message:<br />
  <div id="output"; style="display:flex; justify-content:center;"></div>
</p>
</div>

```

Here is possible to see the code that defines the input area and the output area for the ciphers, as we can see the text-area, the output area, and the button are identified with the same ids that we saw in the JavaScript functions, to make them able to interact.

Critical Evaluation

Examining the Coursework requirements and my personal webpage, the main discrepancy can be found in the JavaScript code: main aim of the coursework was to develop at least two ciphers able to encrypt and recover the message, analysing my code, and obviously seeing the webpage, can be noted that in the Morse webpage is not present a button, and a function, to decode the message, so the first thing to implement would be for sure a Morse Code Decrypt function. Checking on the Caesar cipher page, is also absent the button to decrypt the message but if we check on the number offset we can see that is possible to set the position shifter on negative numbers so will be possible to retrieve our original message by going back with the same, but negative, value of positions.

The condition that makes the user able to navigate between pages has been fulfilled, thanks to the implementation of: two buttons in the index page to transport him to the selected cipher and a "Back" button in the cipher pages to bring the user back to the main page, thinking about a possible improvement, could be a nice change to substitute the buttons in the index page with a Hover Dropdown menu. The general appearance of the website looks nice and organised but maybe still too basic, it could be improved by adding a background image or some nice decorations to fill the margins.

Personal Evaluation

Beside the Slides and the Labs done during the university days, this coursework gave me the occasion to improve my skills of html coding, the main challenge that I found in front of me was actually: "How could I transform a simple idea into something real?" so from that point started the "research part" of the job. After the initial draft of the webpage aspect and layout, my challenge was to chose at least two ciphers to include in my webpage, of course we examined in the lab the ROT13 cipher so my first choice went to the Caesar Cipher, similar in the work but a little bit different to implement, my secondo choice went to the Morse code, fascinating but difficult to implement, and after lots of attempts, left incomplete.

Beside the ciphers the new challenge was: "How to allow the user to navigate between the pages?", was natural to think about buttons, so mixing the code present in the W3 tutorials, and some resources found on the web, I have implemented three animated buttons.

The Heading and the paragraph parts have been examined during the Labs so most of the job was related on which font use and on the style to apply, after various attempts focused on find the most pleasant appearance I have opted for a central text-alignment with a Times New Roman font, easy to read.

Different speech for the Images and the captions, of course during the Labs we had practiced on how insert an image into an HTML file, but I wanted to add a proper caption to make them, and the webpage, looking more "academicals". Once again mixing the knowledge from the labs, and the resources from the web, I achieved my goal.

References

- ¹ ForScale(2016)Untitled[Online] Available at: <https://codepen.io/ForScale/pen/KVabZq>. Accessed on 19/02/2019
- ² MattC.K(2016)Caesar Cipher[Online] Available at: <https://codepen.io/MattCK/pen/OyQyxE>. Accessed on 19/02/2019
- ³ Thomas Vaeth(2016)Link to Button Animation on Hover[Online] Available at: <https://codemyui.com/link-button-animation-hover/>. Accessed on 01/03/2019
- ⁴ Huzaifa Sidhpurwala(2019)A Brief History of Cryptography[Online] Available at: <https://access.redhat.com/blogs/766093/posts/1976023>. Accessed on 04/03/2019
- ⁵ TutorialsPoint(2019)Origin of Cryptography[Online] Available at: https://www.tutorialspoint.com/cryptography/origin_of_cryptography.htm. Accessed on 014/03/2019
- ⁶ James Lyons (2012)Caesar Cipher[Online] Available at: <http://practicalcryptography.com/ciphers/caesar-cipher/>. Accessed on 04/03/2019
- ⁷ NRICH team(2004)History of Morse[Online] Available at: <https://nrich.maths.org/2198>. Accessed on 04/03/2019
- ⁸ History.com Editors(2018)Morse Code & the Telegraph[Online] Available at: <https://www.history.com/topics/inventions/telegraph>. Accessed on 04/03/2019
- ⁹ Bert Bos(2018)Web Style Sheets CSS tips & tricks[Online] Available at: <https://www.w3.org/Style/Examples/007/figures.en.html>. Accessed on 05/03/2019