Cipher Web Technologies Project

Using Caesar and Vigenere’s Ciphers

Author

Ronan Coyne

40417360@live.napier.ac.uk

Date

06 March 2019

Insert relevant location disclaimers

# Introduction:

The problem I was presented with was to design a web page using html, JavaScript and CSS to implement that enables a user to encode text messages using at least two different classical cyphers or encoding schemes.

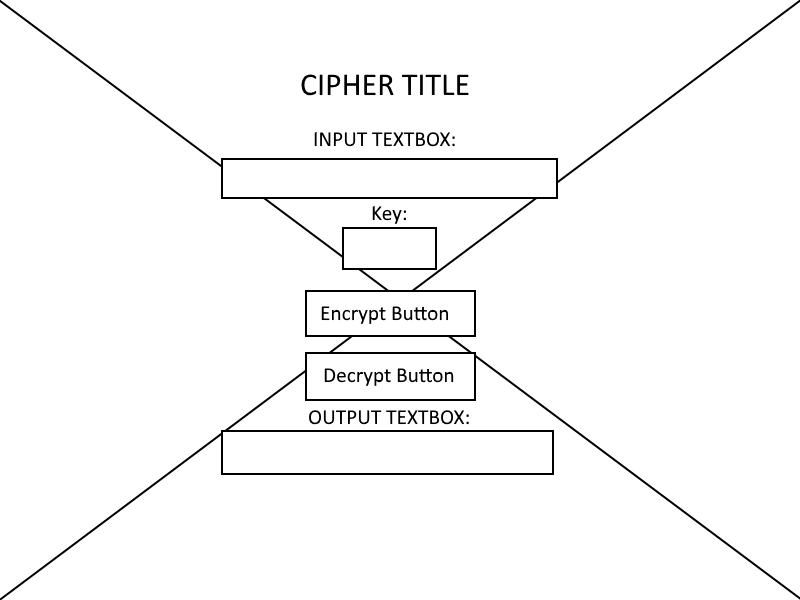
My solution was to implement the Caesar Cipher which is also known as Rot13 cipher, this is a simple encoding mechanism that simply shifts the letter by the chosen key in the alphabet/character set. I chose this as it is relatively simple and had experience implementing it in python. My second cipher was the Vigenere cipher, this was chose as I am familiar with how it works and thought it would be an interesting implementation in JavaScript to test my knowledge of loops and input validation while keeping the algorithm similarly easy to the Caesar Cipher.

In addition to this I had to provide a usable UI with a nice aesthetic using CSS and html.

# Software Design:

My approach was to begin by drawing out the base UI for the cipher in paint to give myself an idea of how the webpage would be structured.

Design of page:



Next I had a look at my python code for the Caesar cipher and thought about how I would implement this in JavaScript.

After this I read up on Vigenere’s cipher and found a very useful resource on implementing it in JavaScript.

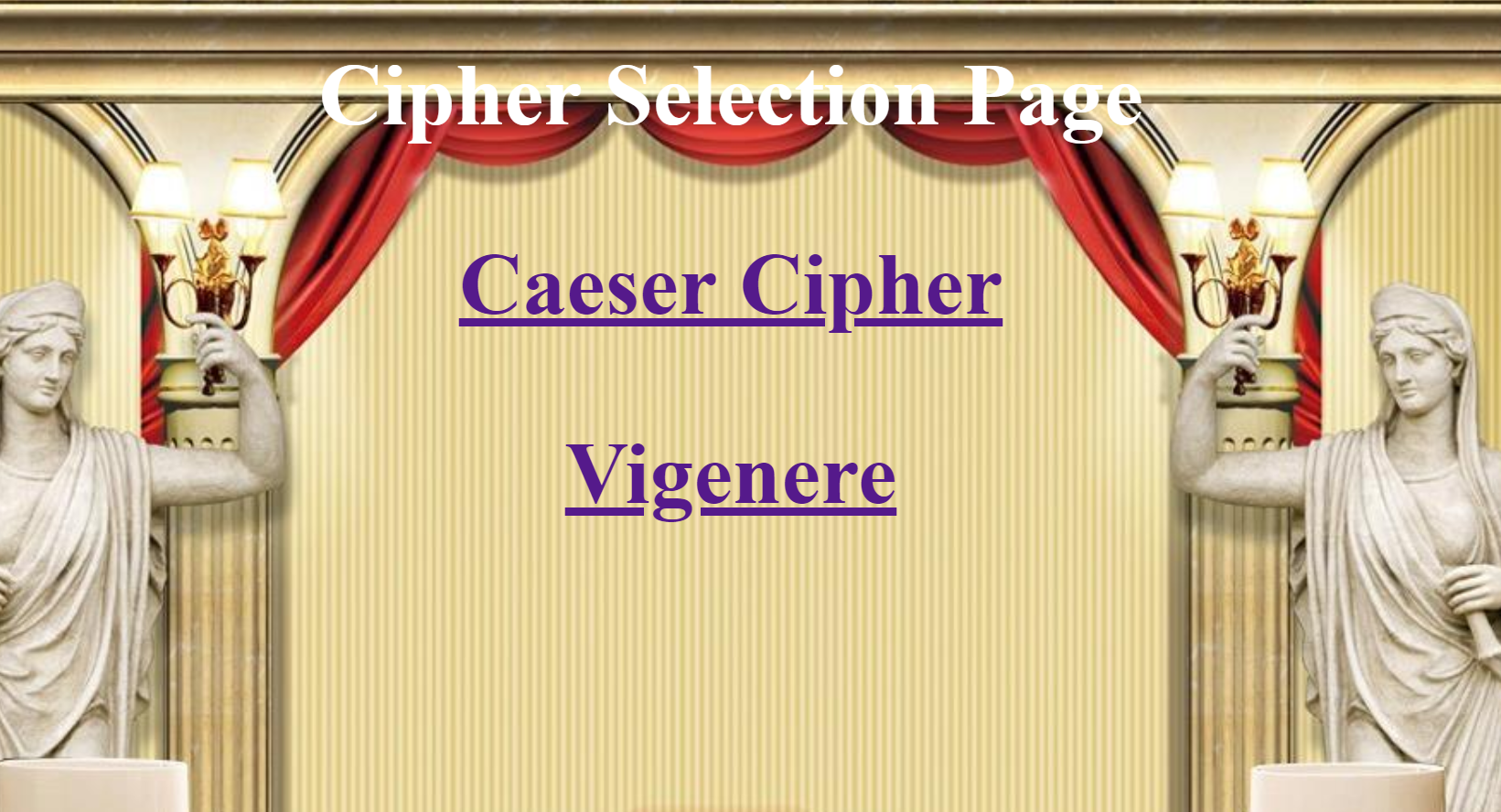
<https://www.nayuki.io/page/vigenere-cipher-javascript>

After doing this preparation I felt ready to build my webpage.

# Implementation:

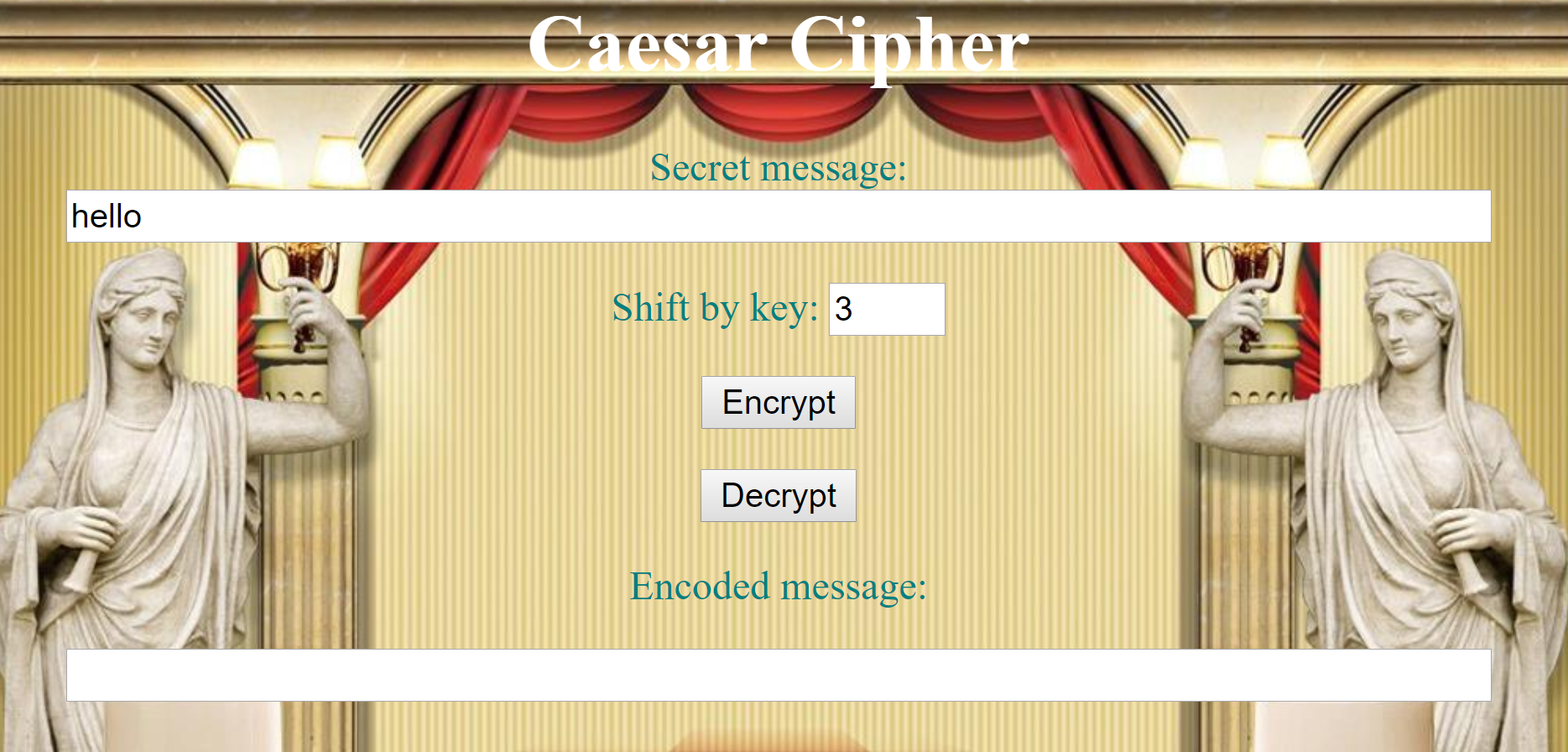
My implementation uses a homepage (index.html) where you select the cipher you wish to use (either Caesar or Vigenere) by clicking on its link which brings you to the webpage related to that cipher.

### Cipher Selection Page:

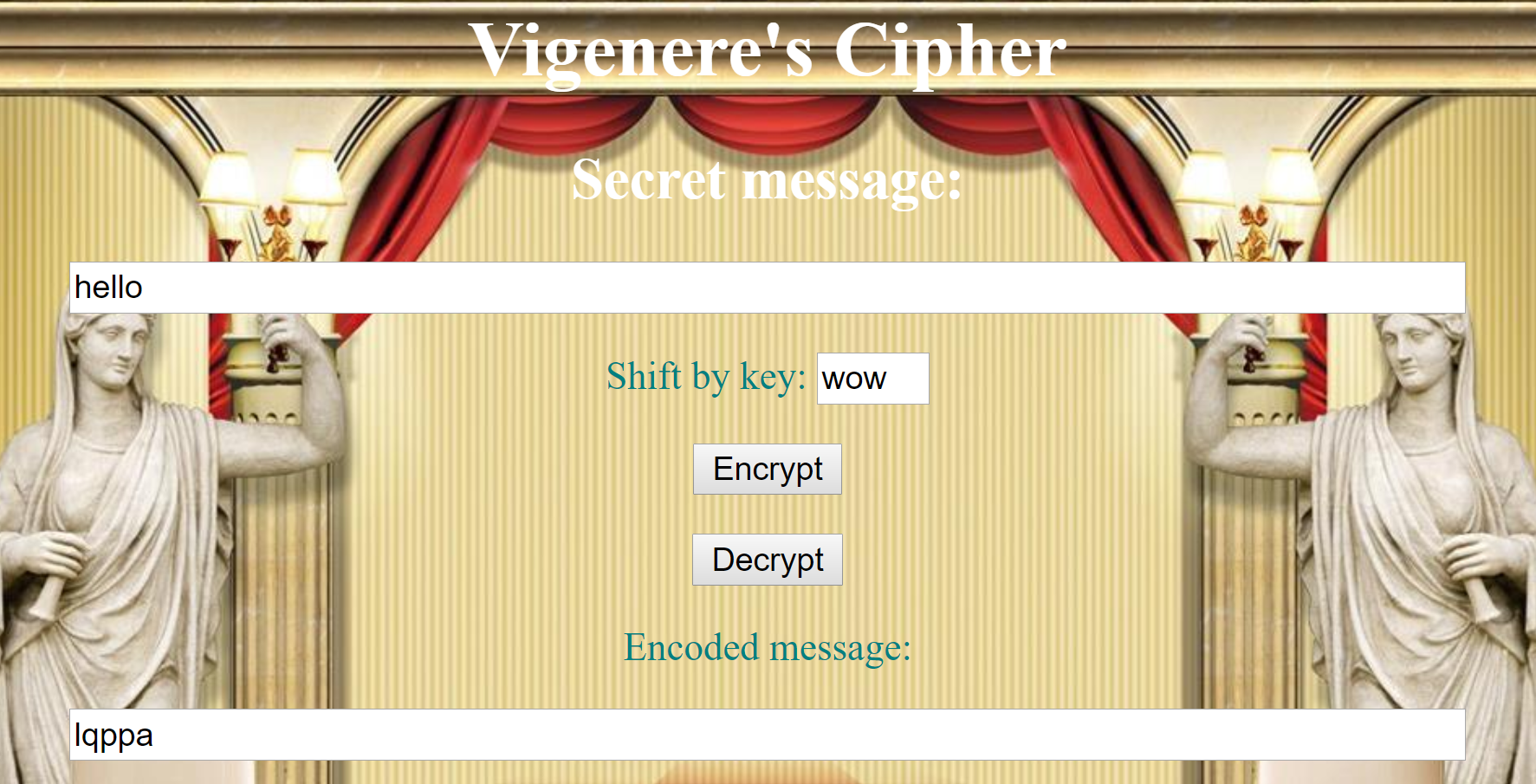


Upon clicking on one of the cipher links it brings you to either of these pages:

### Caesar Cipher Page:



### Vigenere Cipher Page:



I used these webpages in conjunction with external stylesheets and external JavaScript files to create a modular solution and reduce re-use of code. The css was kept simple as this is a simple application.

# Evaluation:

I believe my implementation fulfils all the requirements of the coursework.

1. *an area in which your user can type in a message,*

The input textbox covers this.

1. *an area where the encoded message can be displayed,*

*The output text box covers this.*

1. *a way to select between different cyphers (for example a dropdown),*

*The index cipher selection page covers this.*

1. *some method to cause the cypher to be computed (such as a button),*

*The encrypt button covers this.*

1. *some mechanism for deciphering your messages to recover the plaintext.*

*The decrypt button covers this.*

### Possible improvements:

I believe I meet the requirements quite well however would like to improve it by adding more cipher pages and maybe making an event listener that real-time encodes it to the output on key press instead of the user having to press the button after typing the message.

# Personal Evaluation:

I believe this was a great project for my development as it involved use of external style sheets and UI design which I do not do a lot of in my work-place as my development involvement in the work place is mostly back-end development and remediation of vulnerable web apps.

I think I dealt with the challenge well and am very happy with my implementation.

# References:

<https://www.dhgate.com/product/european-style-3d-roman-column-background/396427732.html>

<https://www.nayuki.io/page/vigenere-cipher-javascript>