

**“iLaugh” Marketing Website**

**Student name :**

**Student number :**

**Module title : WEB TECHNOLOGIES (QHO431)**

**Module tutor : Akinola Siyanbola**

**Contents**

Contents

[1. Introduction 2](#_Toc115127088)

[2. Website Technologies 2](#_Toc115127089)

[3. Website Development 4](#_Toc115127090)

[4. Accessibility 20](#_Toc115127091)

[5. Testing 21](#_Toc115127092)

[6. References and bibliography 21](#_Toc115127093)

[7. Appendix 22](#_Toc115127094)

[A. Validating different pages conform to W3C standards 22](#_Toc115127095)

# Introduction

This is an online marketing website, promoting a night club with live entertainment.

**\*\* talk about scenario**

# Website Technologies

I've done my utmost not to use prohibited technology and have found workarounds to avoid using JQerry in particular.

Using only HTML, CSS, Javascript, and Node JS and adhering to all W3C standards, I feel I have satisfied the majority of the requirements.

Frontend

In the frontend I used the required technologies, HTML, CSS, Javascript and with a bit of practice I was able to produce a successful wireframe and apply a design to it. All the images and other elements have been imported from non-copyright websites like pexels, unsplash and pixabay while icons are from ionic.io.

Along with these elements, AJAX scripts are in place around the website, in the contact form and line-up information as well, to help easily retrieve the data.

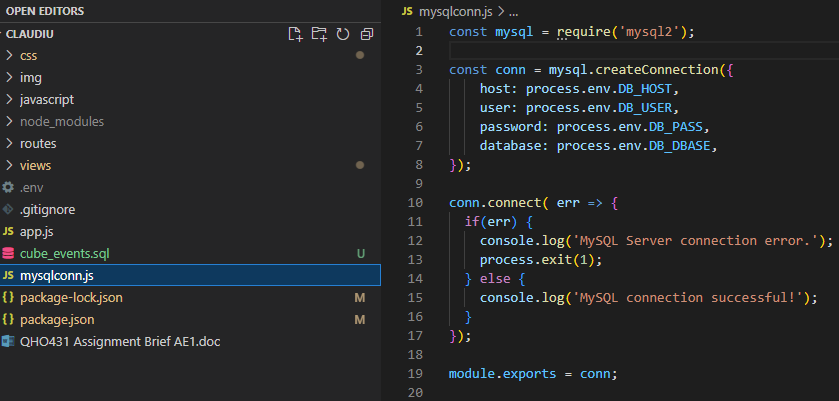
Backend

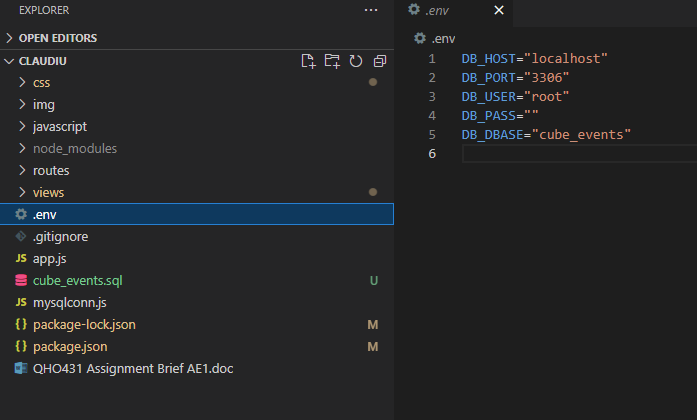
Backend technologies used as required, NodeJS with ExpressJS as framework. This together with EJS as view engine made it easy to transmit data dynamically form the server into beautiful HTML code without the need of hard coding. To help me out, in NodeJS, the following packages have been imported:

* Dotenv – to store all the possible secret variables
* EJS – a view engine template
* Express – as required by the project, one of the most popular framework
* Express-EJS-Layouts – helped me save a lot of code repetition, by declaring a layout and applying it to every page thus helping our future website users to easily remember, recognize and find whatever they are looking to achieve.
* MySQL2 - Node.js MySQL client with a performance-focused design
* Nodemon – node package to help with the process of development while loading in all the changes without the need of manually restarting the node server.

Database

MySQL another project requirement, is installed and used as in the following code:

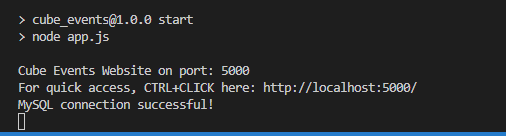




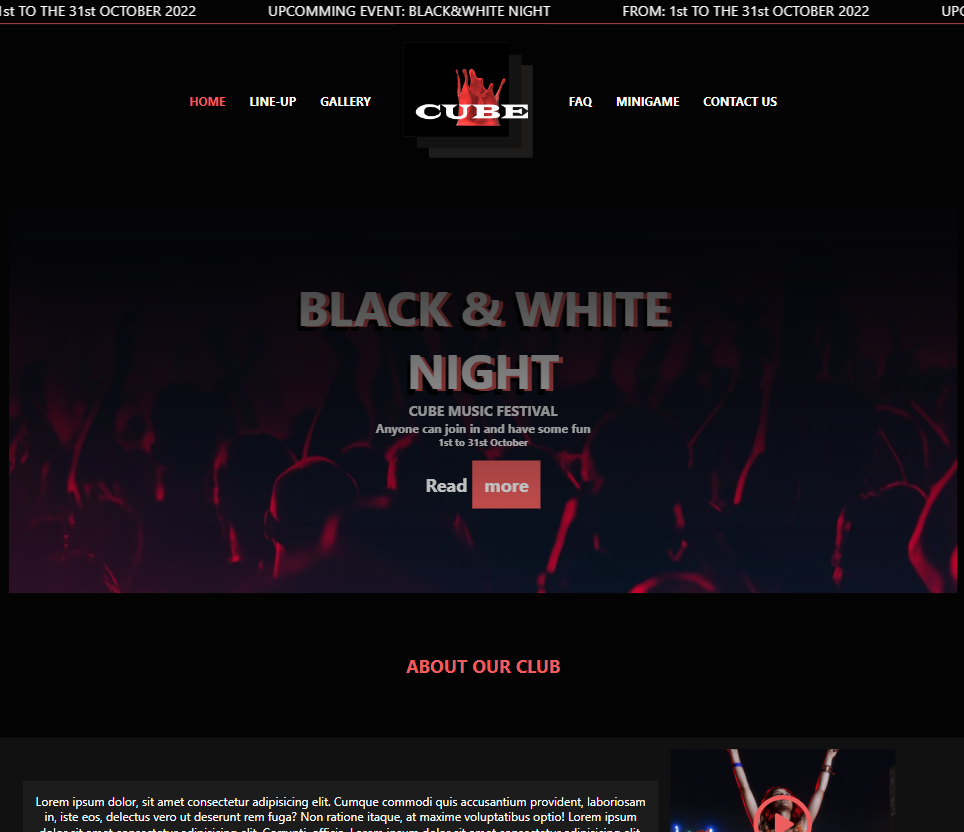
# Website Development

The structure of the project is as follows:

To start the server, the folder is opened in a terminal and the command “npm start” is introduced.



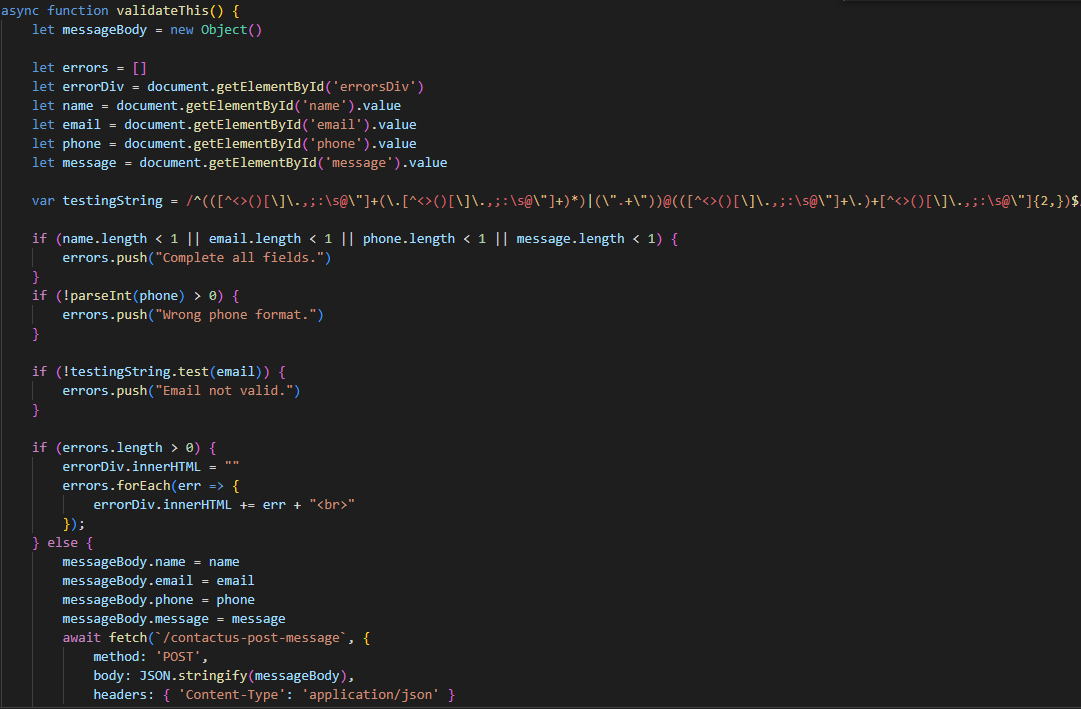
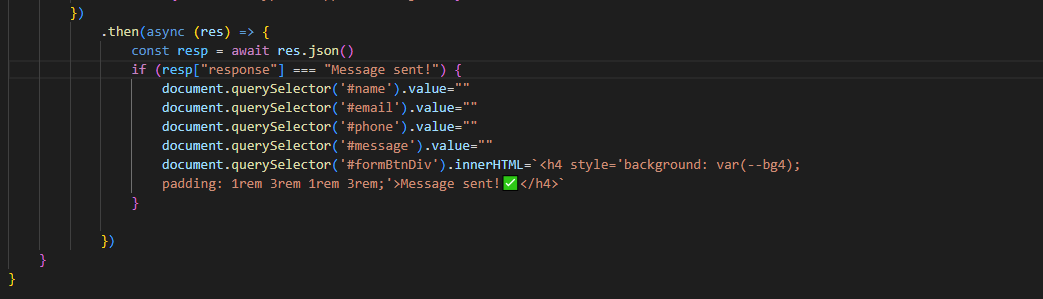
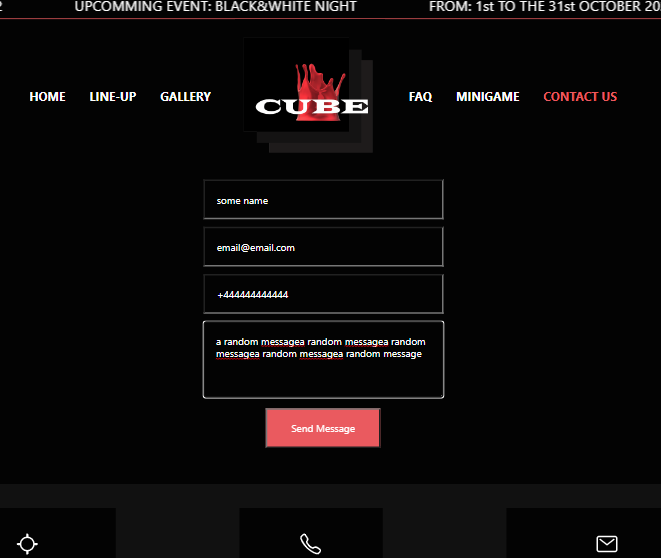
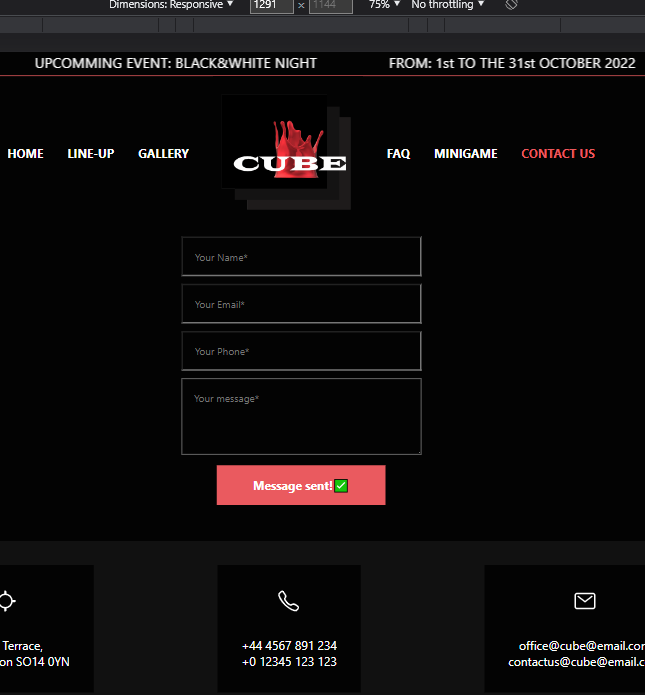
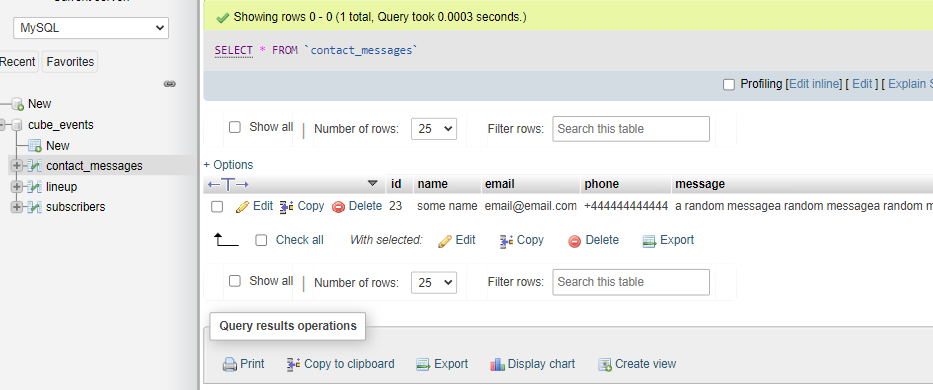
Website layout is quite standard with all the good practices out there, consisting with a navigation bar, main content and footer with the very top a small div with some news.



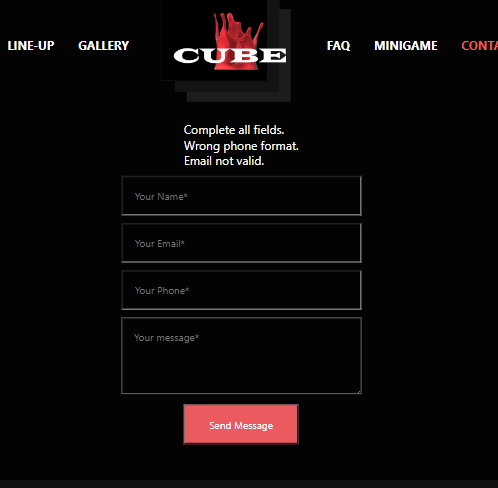
Some Javascript functionality is added in the :

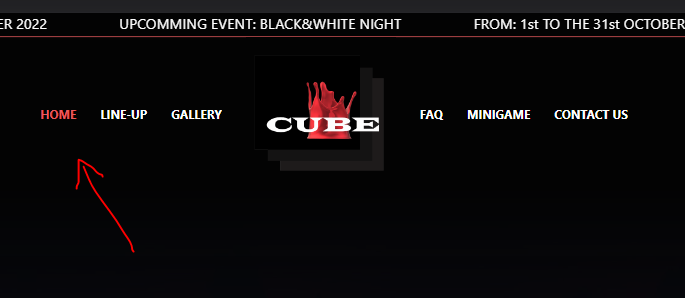
* Contact form

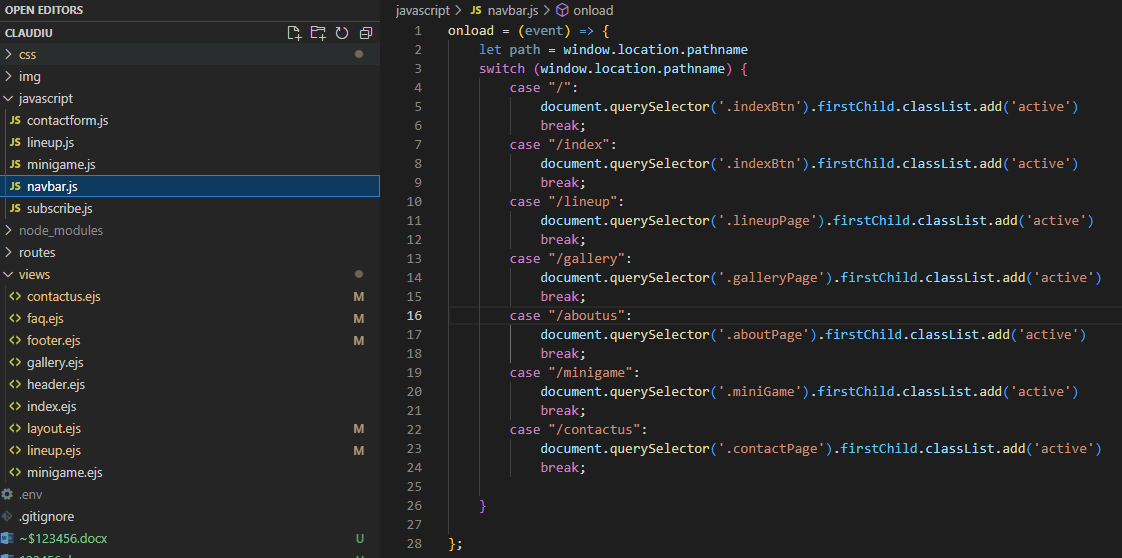
Build a simple form validation and error reporting script where is checking for the values and making sure they are as defined and within parameters. Accumulating every error and reporting them at the end of the checkup. If all the values are correct, an AJAX script is executed, inserting the contact form into the MySQL database and processing results.

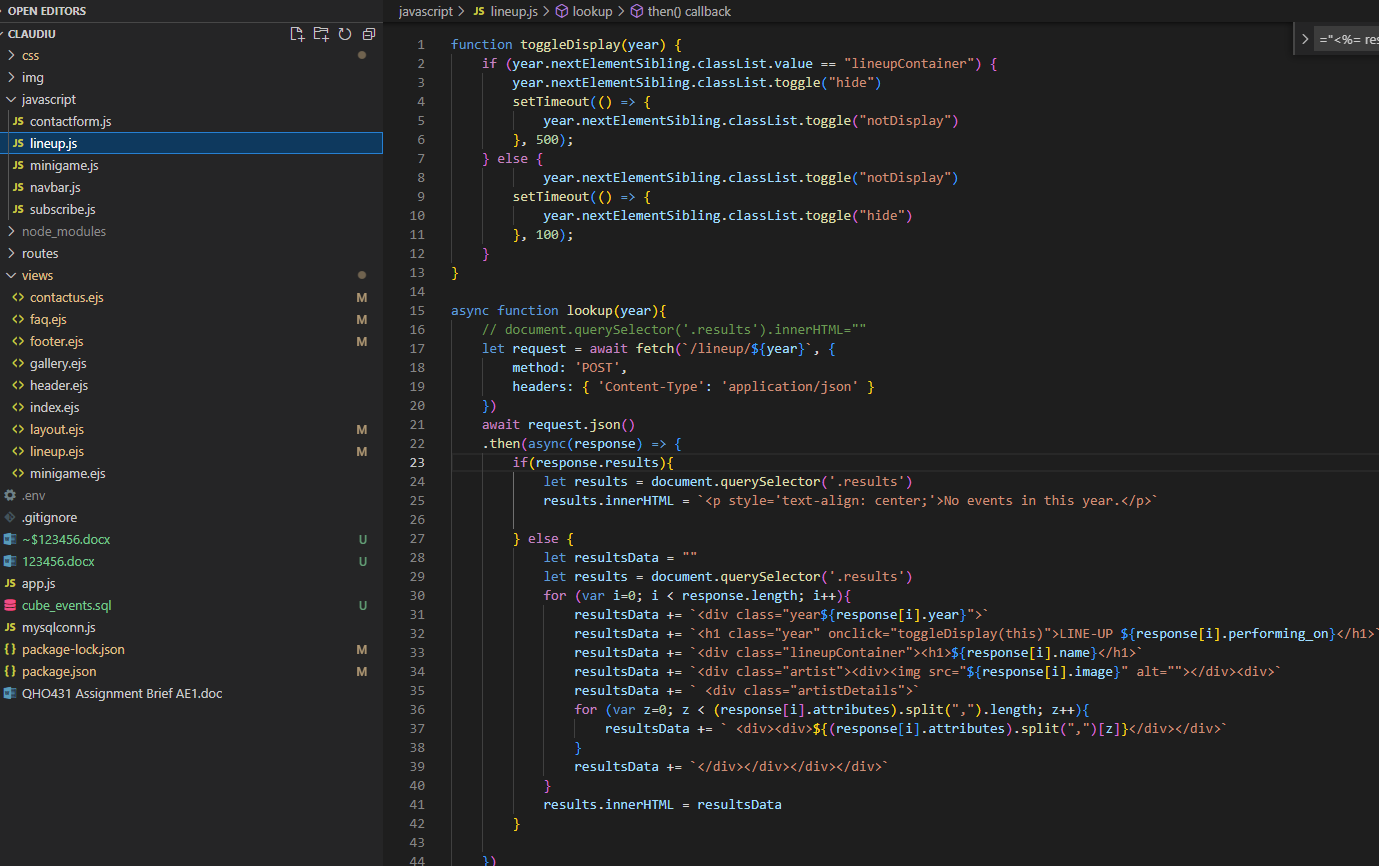
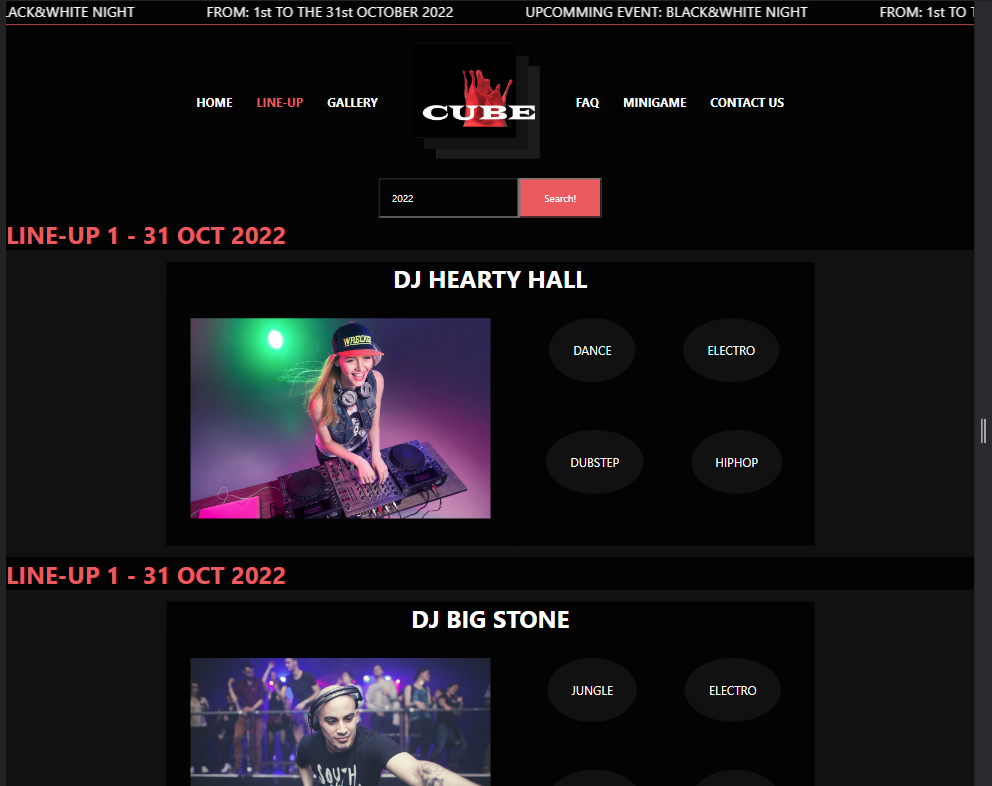
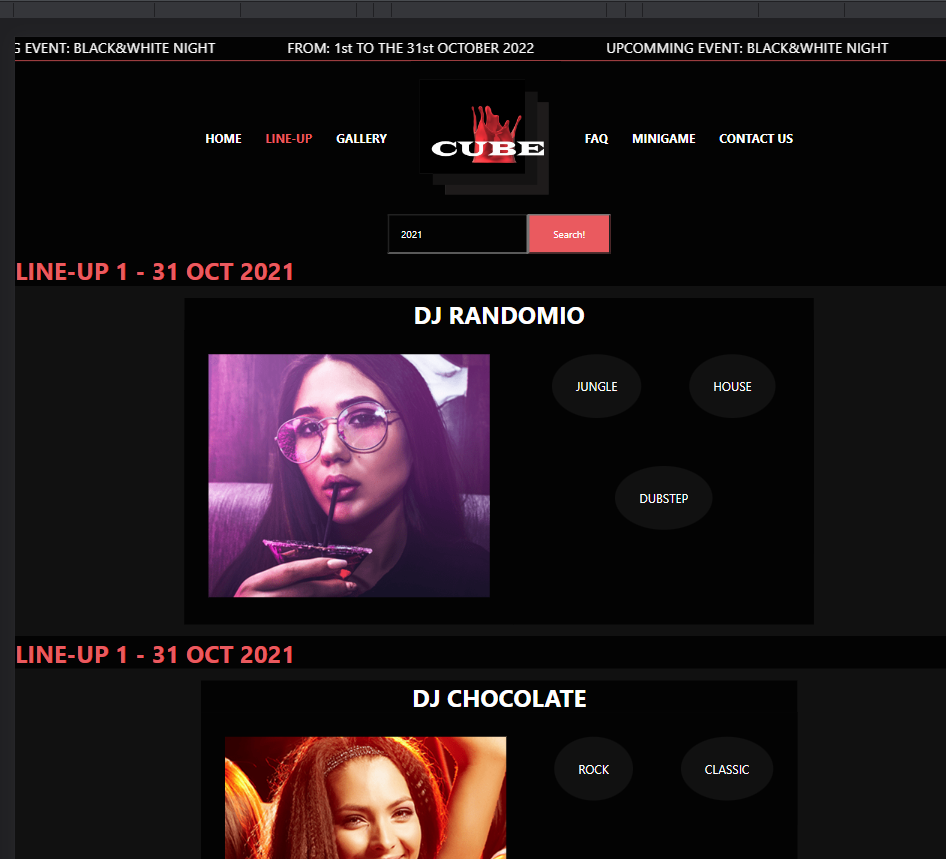
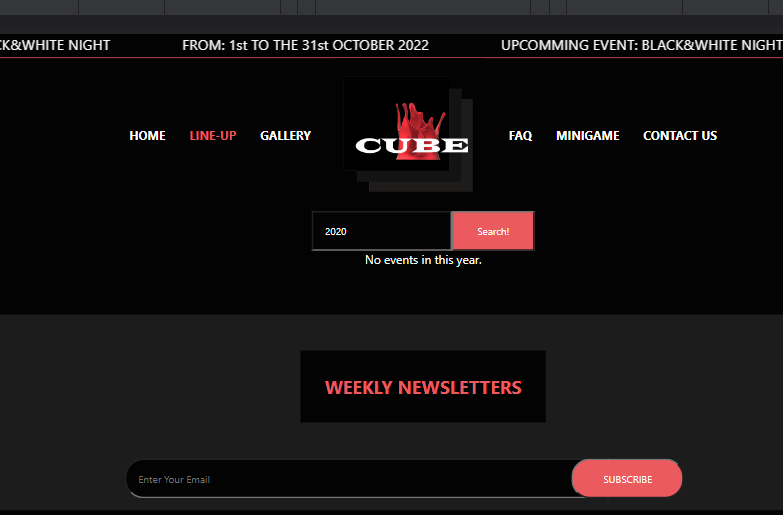
If the contact form has errors or the pattern is not satisfied the following errors will appear above the form:



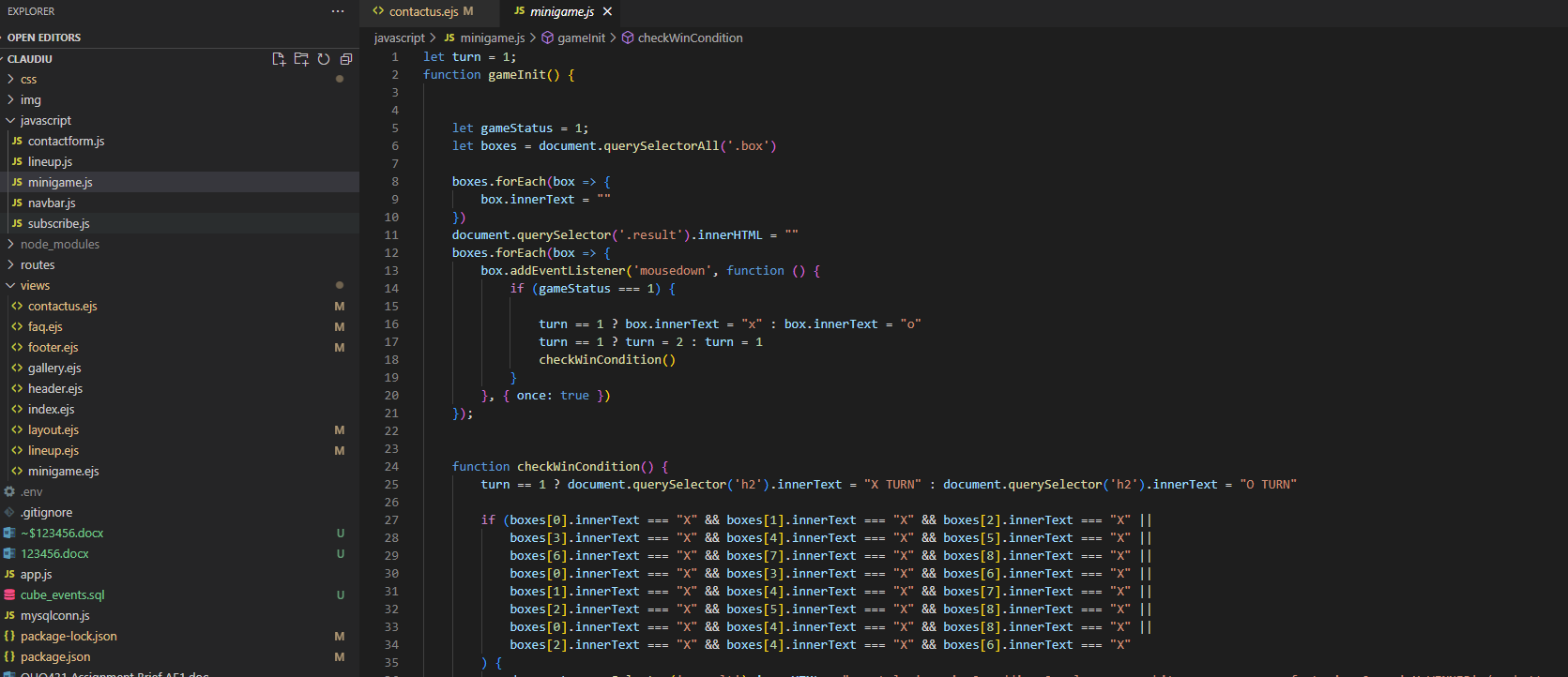
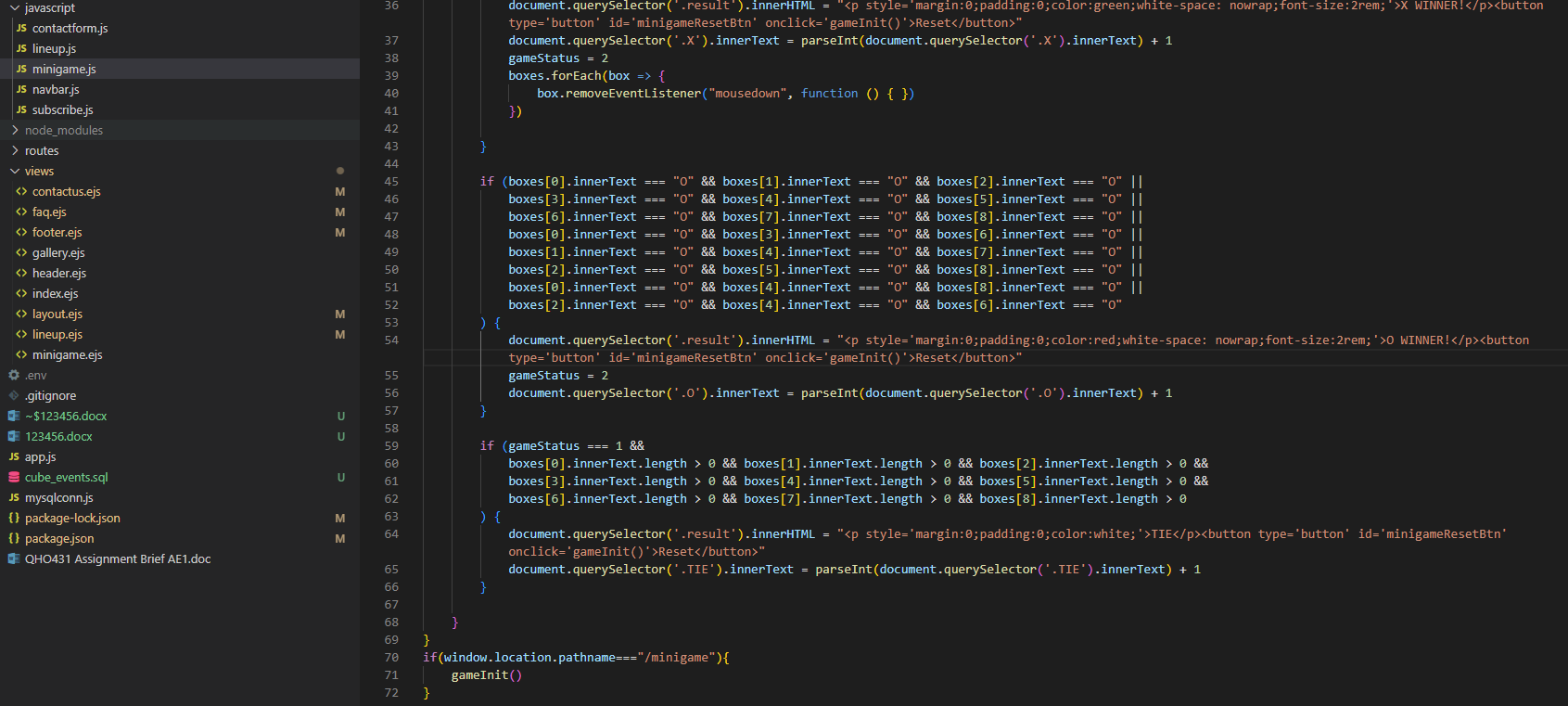
* Navigation Bar – adding the active class to let the user know where he is currently situating on the website through a red color stroke 

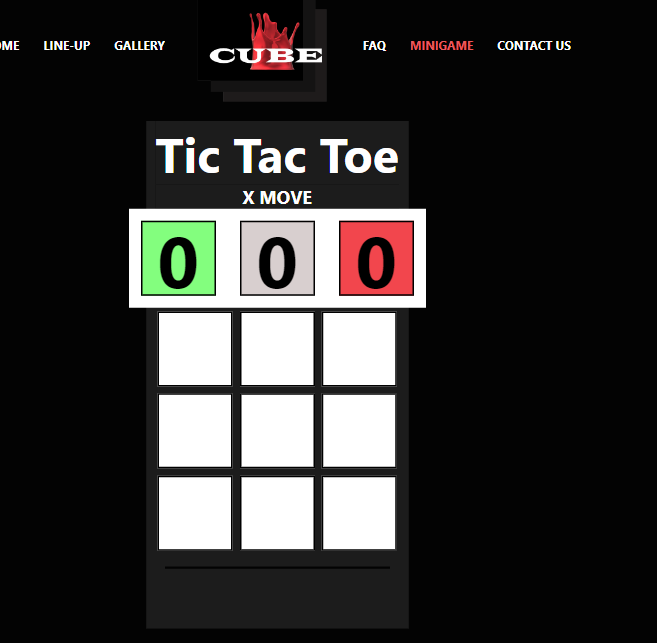
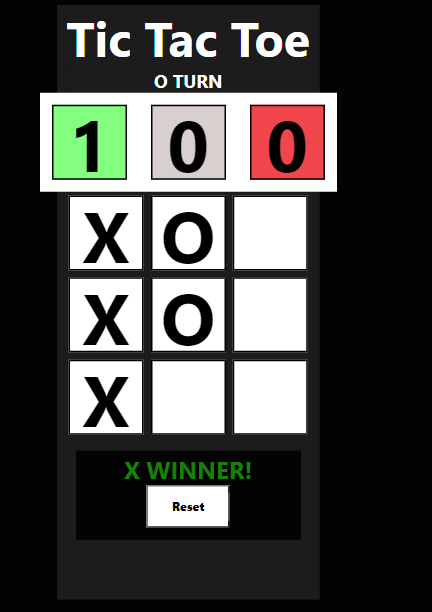


* Line-up – with a simple show/hide modal and an AJAX call to retrieve the details of selected year of the festival. All these details are retrieved from the database and built as received through the loop of the results.

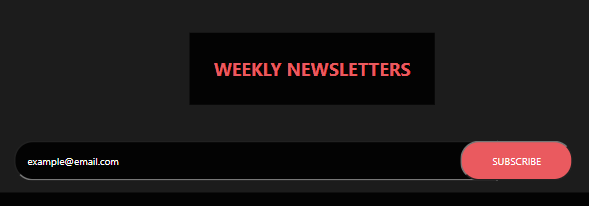
   

* Minigame – one of my proud moments where I learnt to write a simple code. Maybe in the future will write an AI code to play with. It keeps score and count of who’s turn is. Code is very basic and for sure it can be improved.

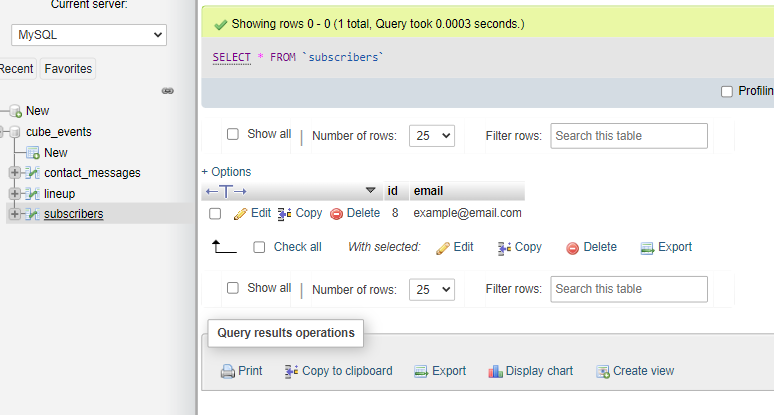
 

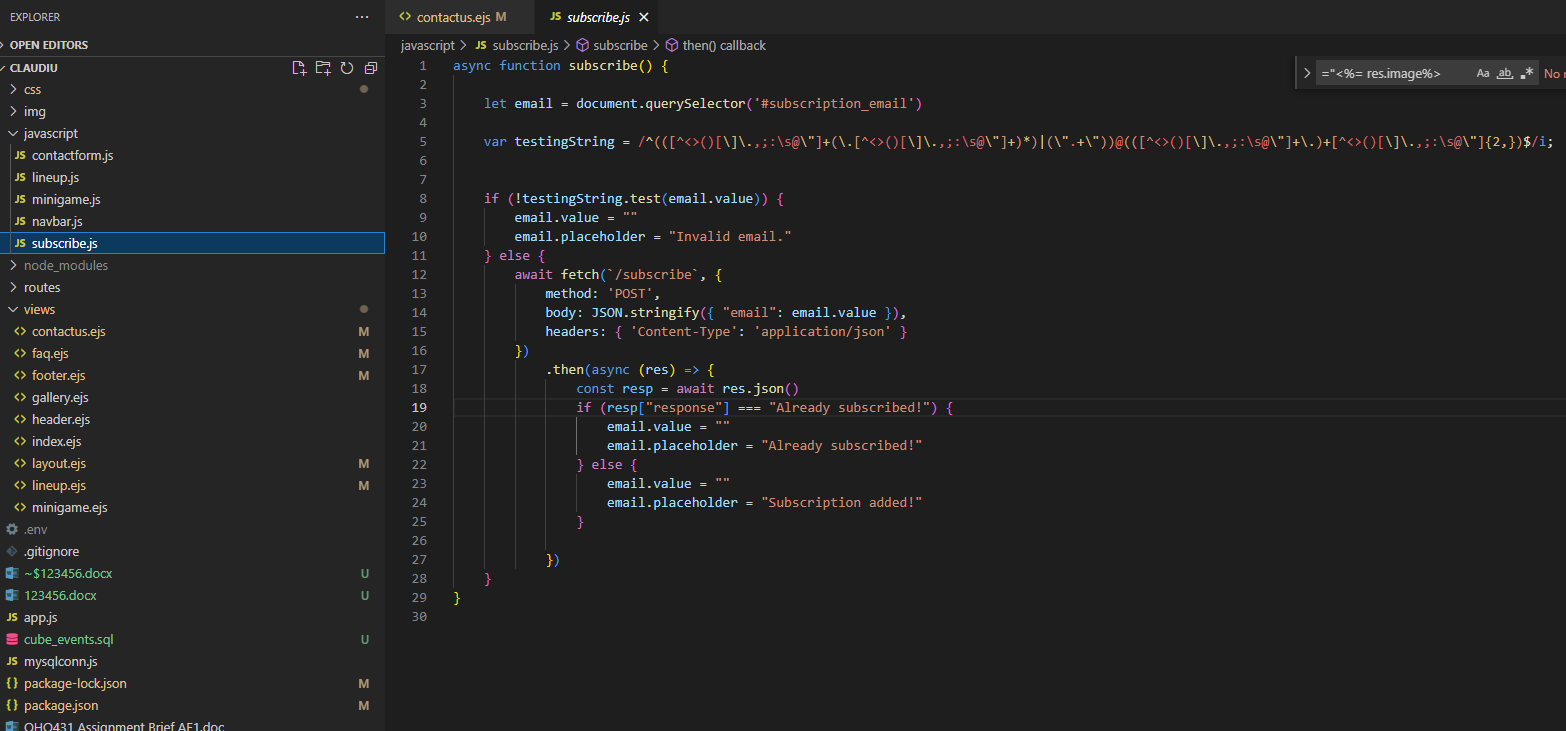
 

* And something extra, a news subscription button, a input field that is checking for errors first and then an ajax call 

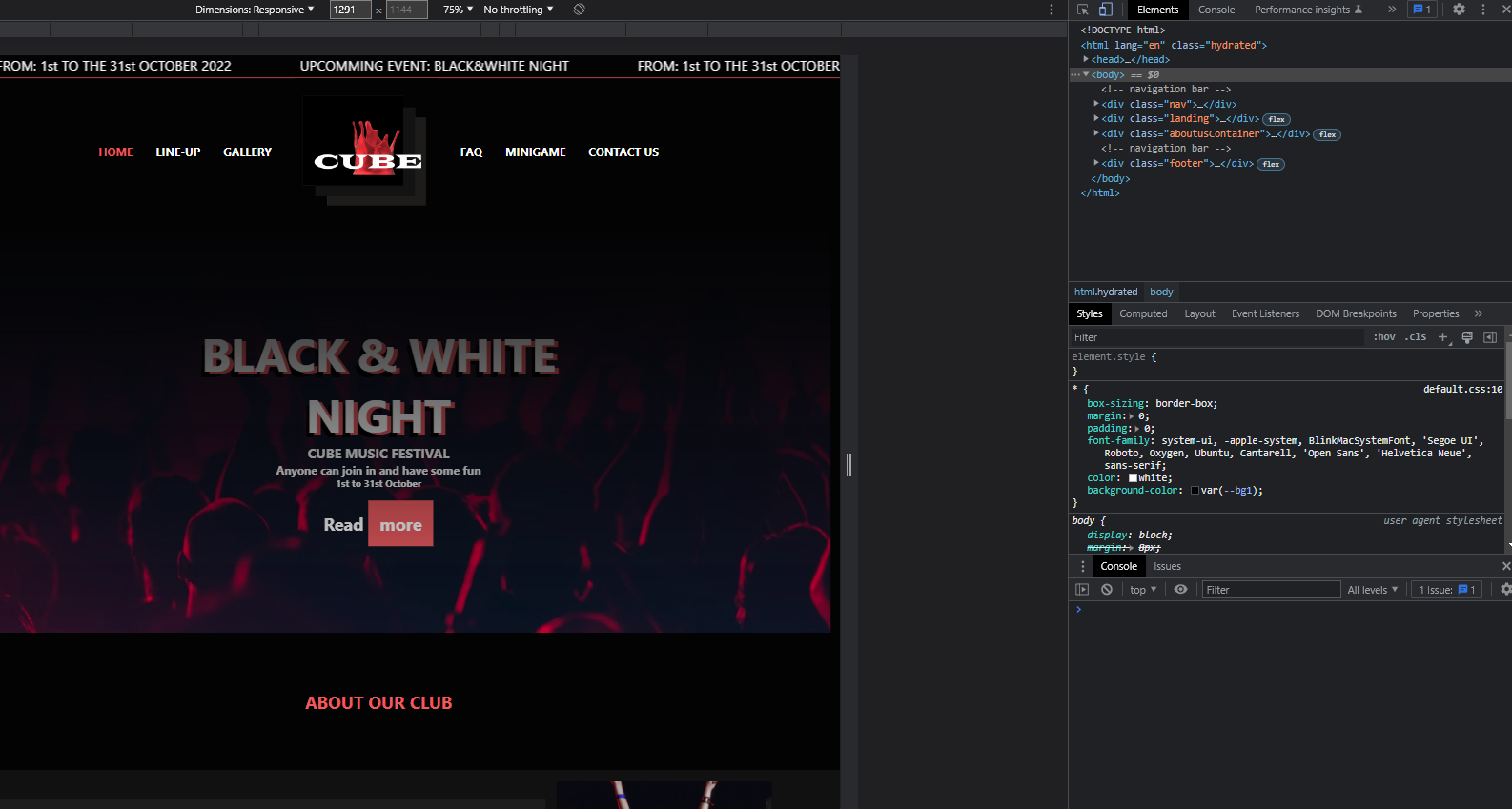


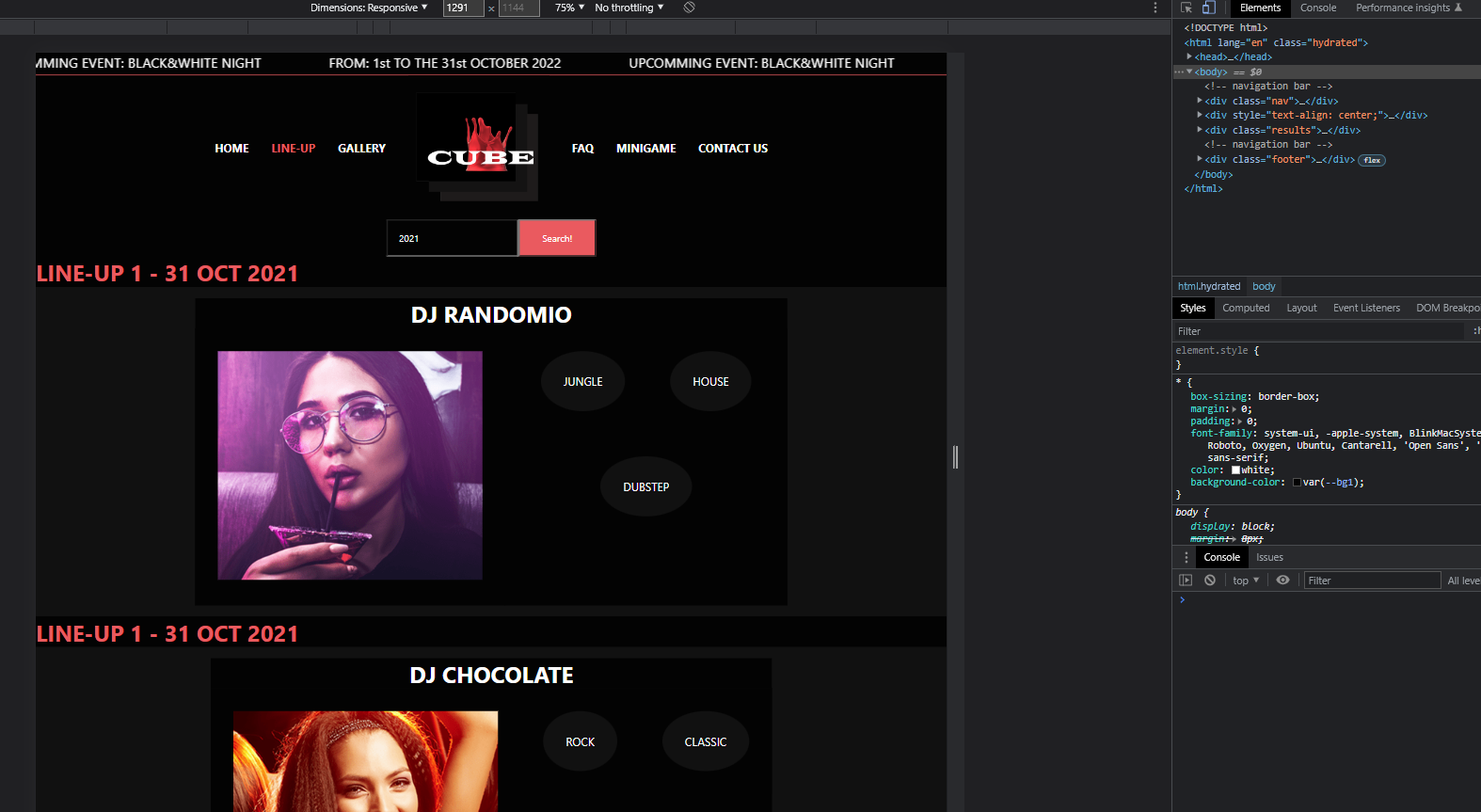


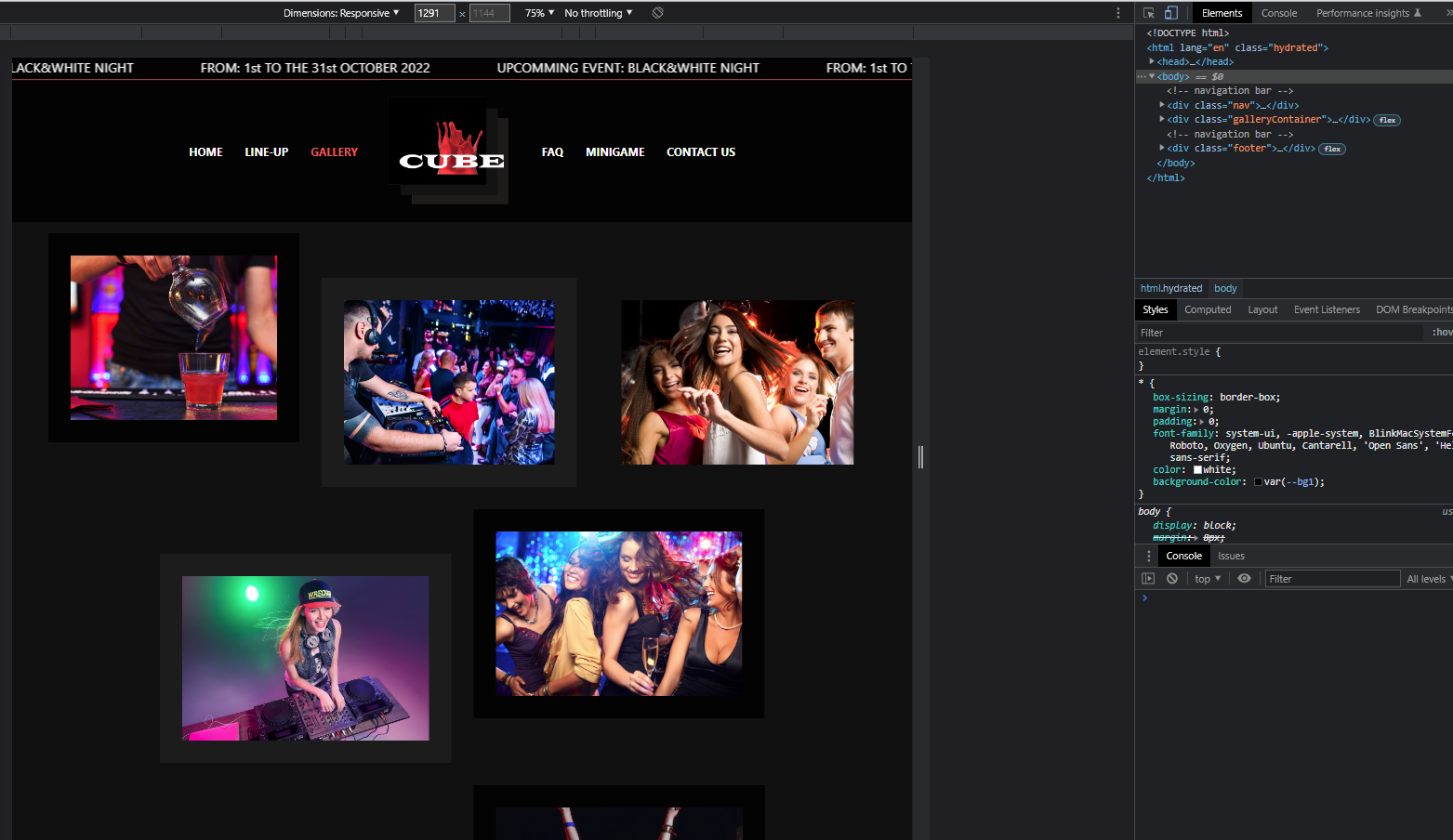




The website works 100% flawless, with no error codes in the developer console and is 100% validated with w3c validation tool.

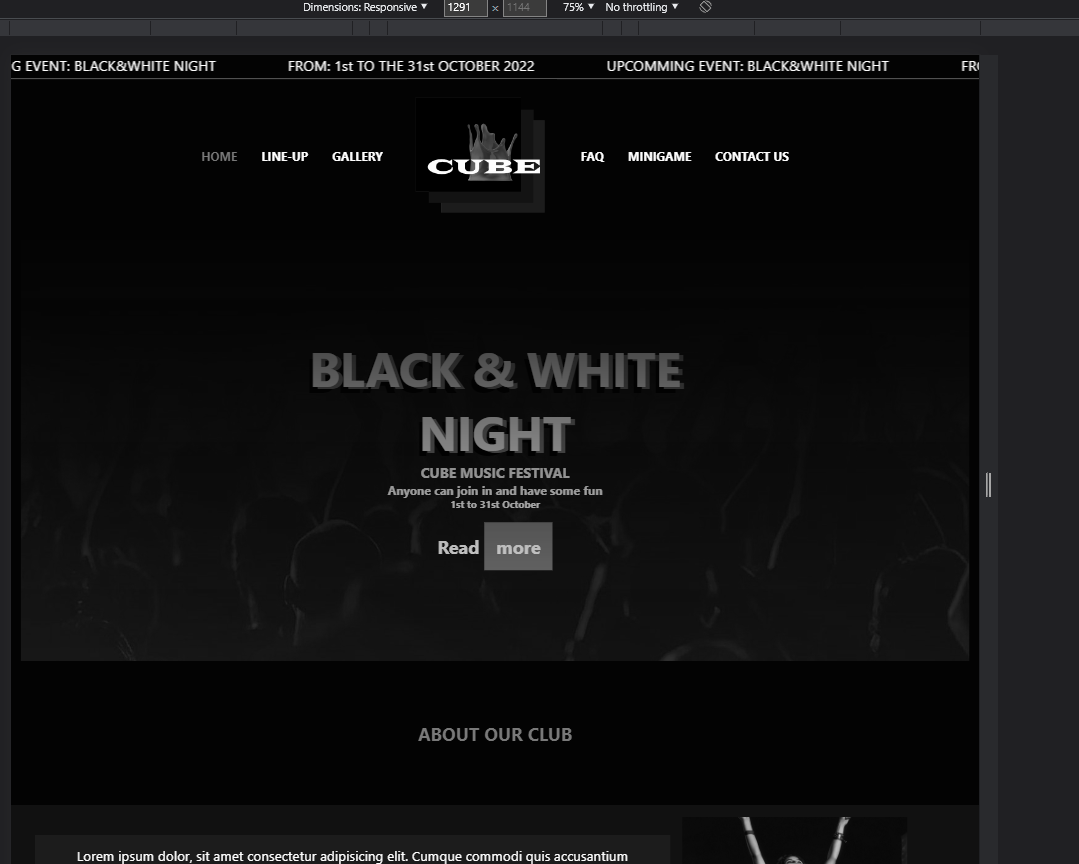
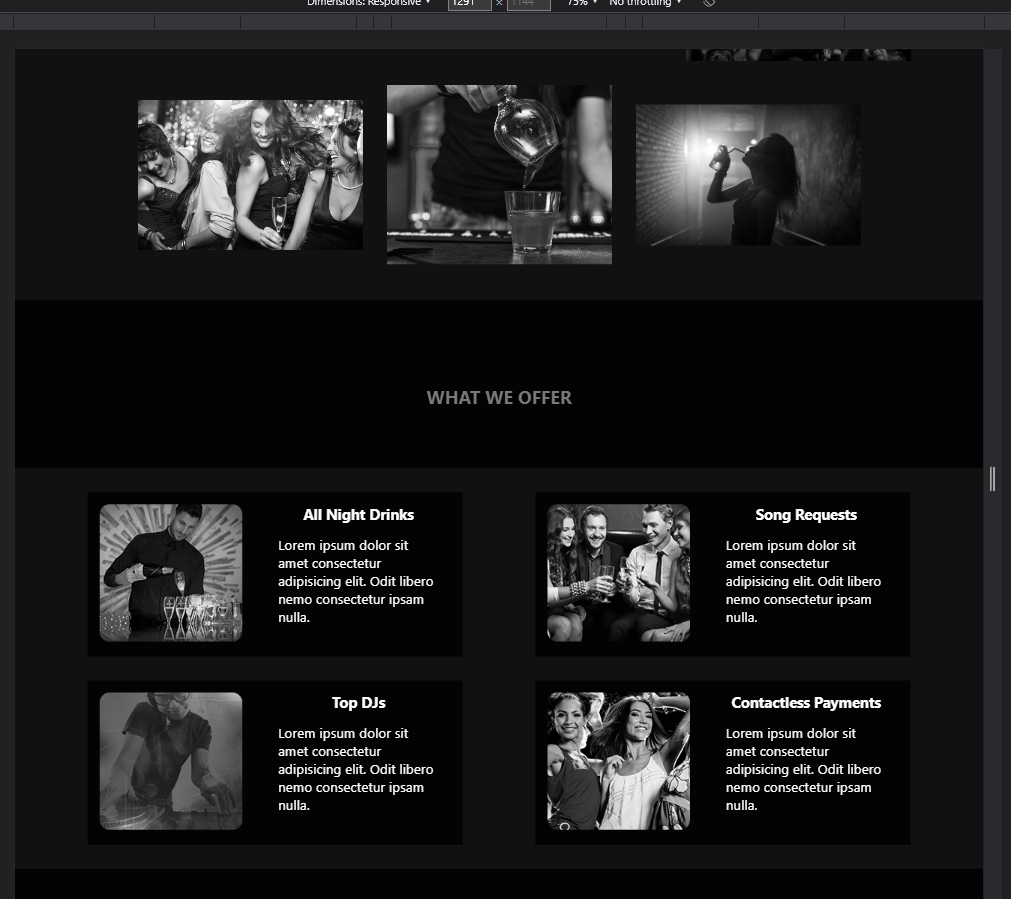
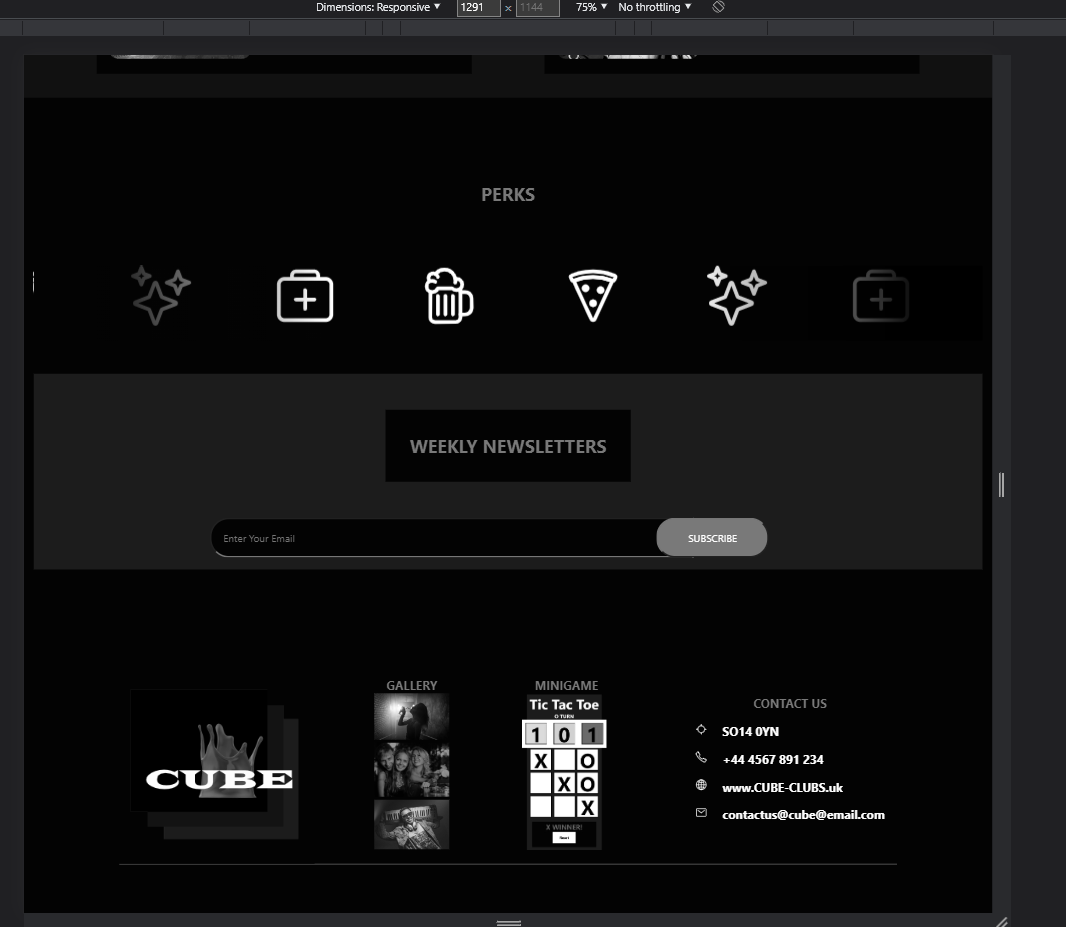
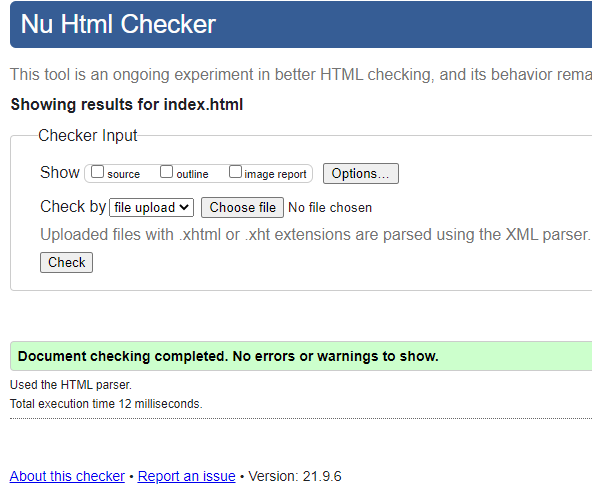
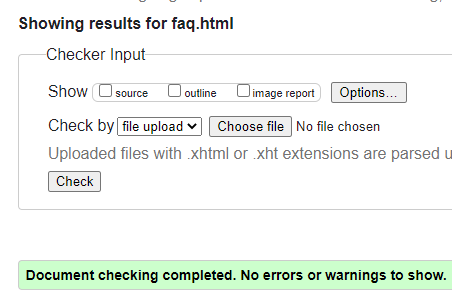
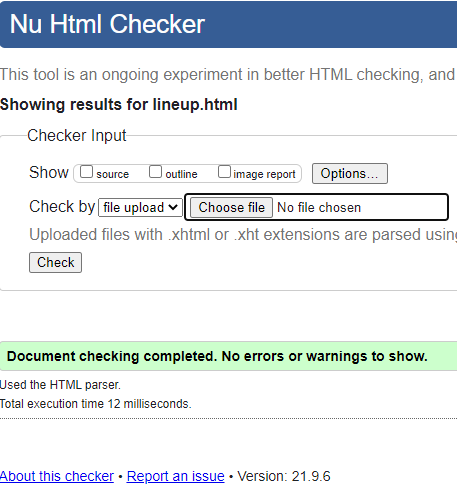






# Accessibility

The website has some accessibility features. My colour palette is suitable for colour blind people, it is fully navigable with the help of only the keyboard and all my images have an equivalent alternate text. For example, to people who cannot see and use a screen reader that reads aloud the information on a page, including the alt text for the visual image.

# Testing

The website has been tested on many different devices, as you can see in the Appendix B

Also been tested on multiple browsers with supporting evidence in Appendix A.

# References and bibliography

1. What is JavaScript, last accessed 17/09/2022, <https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript>
2. Javascript tutorial, last accessed 15/09/2022, <https://www.w3schools.com/js/>
3. What is HTML, last accessed 15/09/2022, <https://www.w3schools.com/whatis/whatis_html.asp>
4. CSS introduction, last accessed 15/09/2022, https://www.w3schools.com/css/css\_intro.asp

5. Logo Creator, https://www.jimdo.com/website/logo-creator

6. About NodeJS, last accessed 15/09/2022, https://www.w3schools.com/nodejs/nodejs\_intro.asp

# Appendix

## A. Validating different pages conform to W3C standards