Kiki’s Diner Milestone 2

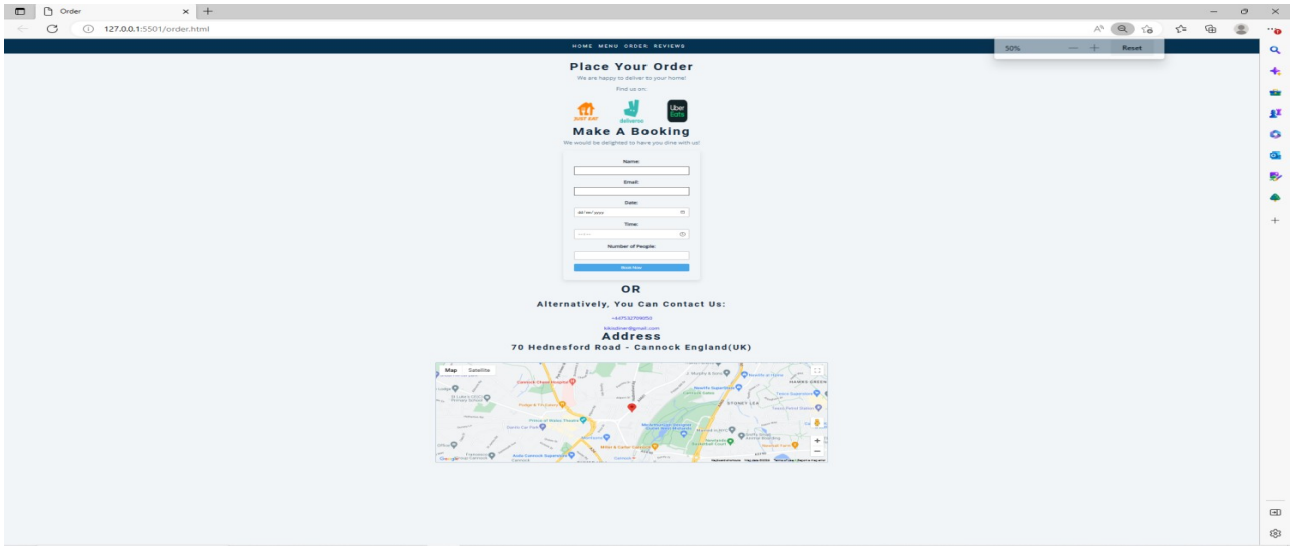


Table of Contents

1. [UX User Experience]

2. [Design]

3. [Technologies Used]

4. [Deployment]

5. [Testing]

6. [Credits]

UX User Experience

2.1 Project Goals

The goal of this project is to create a user-friendly and visually appealing website for Kiki's Diner, a fictional restaurant. The website aims to provide an enhanced user experience by offering a dynamic menu, customer reviews, a booking form, and easy navigation.

2.2 Development and Business Goals

- Develop a responsive website to ensure compatibility across various devices.

- Create an interactive menu section to showcase the restaurant's dishes and prices effectively.
- Implement a customer review system to display authentic feedback and build trust among users.
- Provide a user-friendly booking form to facilitate easy online reservations.
- Optimize the website for search engines to improve online visibility and attract potential customers.

2.3 User Stories

As I user I want to..

1.Be able to gather information about the establishment.

2. Inform myself on the services the establishment provides, and its location.

3. The option to book a table online to minimize waiting times and ensure a seamless dining experience.

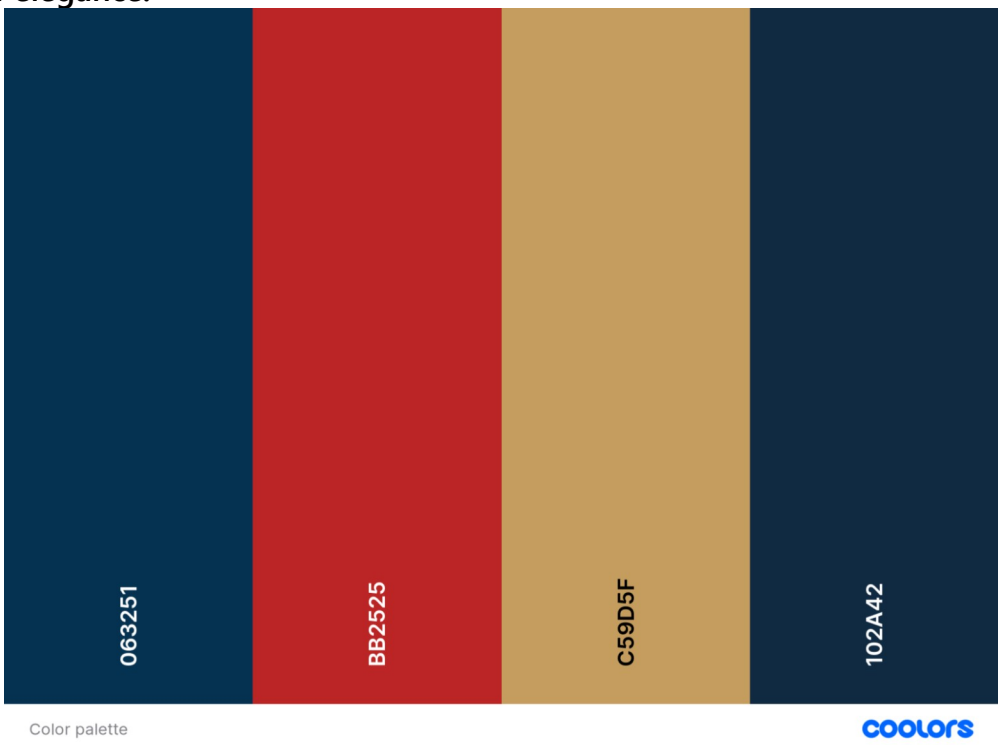
4. Have easy access to the restaurant's social media links to stay updated on the latest news and promotions.

5. As a first-time visitor, I expect the website to load quickly and have an intuitive navigation system for effortless browsing.

Design

3.1 Colour Scheme

The website's design features a warm and inviting colour scheme to reflect the restaurant's theme. The primary colours used are [], complemented by gold accents for a touch of elegance.



3.2 Typography

The chosen typography combines [Roboto from FontAwesome(sans serif backup] to maintain readability and aesthetics throughout the site.

3.3 Icons and Images

The website incorporates [Delivery services icons and Social links icons] for intuitive navigation and visual appeal. High-quality images of the restaurant's dishes and ambiance have been curated to entice visitors.

3.4 Features

- Interactive menu section with detailed descriptions and prices.
- Customer review system for displaying user feedback.
- Easy-to-use booking form for online reservations.
- Social media links for staying updated on promotions.
- Fast-loading and intuitive navigation system.

3.5 Accessibility

The website is designed with accessibility in mind, adhering to web accessibility standards to ensure inclusivity for all users, including those with disabilities. Features such as alt text for images and proper semantic structure have been implemented.

Technologies Used

4.1 Languages Used

- HTML
- CSS
- JavaScript

4.2 Frameworks, Libraries, and Programs Used

- Font Awesome 5.14.0
- Vanila JS (no libraries/frameworks)
- VSCode

-Deployment

5.1 Local Deployment

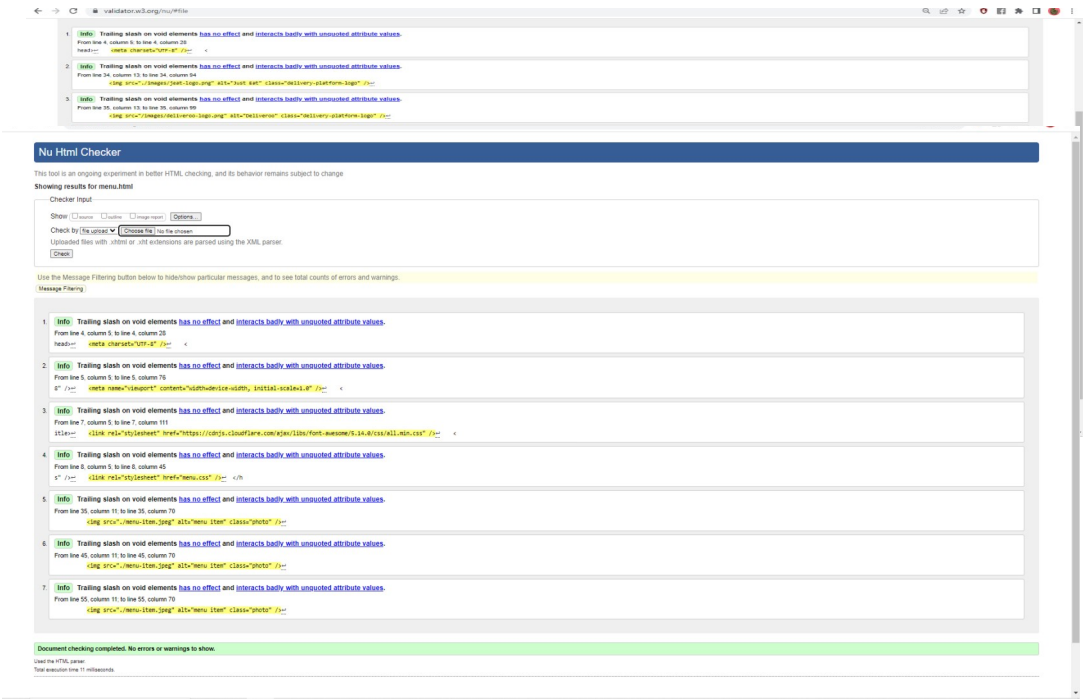
To deploy the website locally, follow these steps:

1. Clone the repository from [https://github.com/SynthSlav/Milestone2.1.git].
2. Open the index.html file in your preferred web browser.

Testing

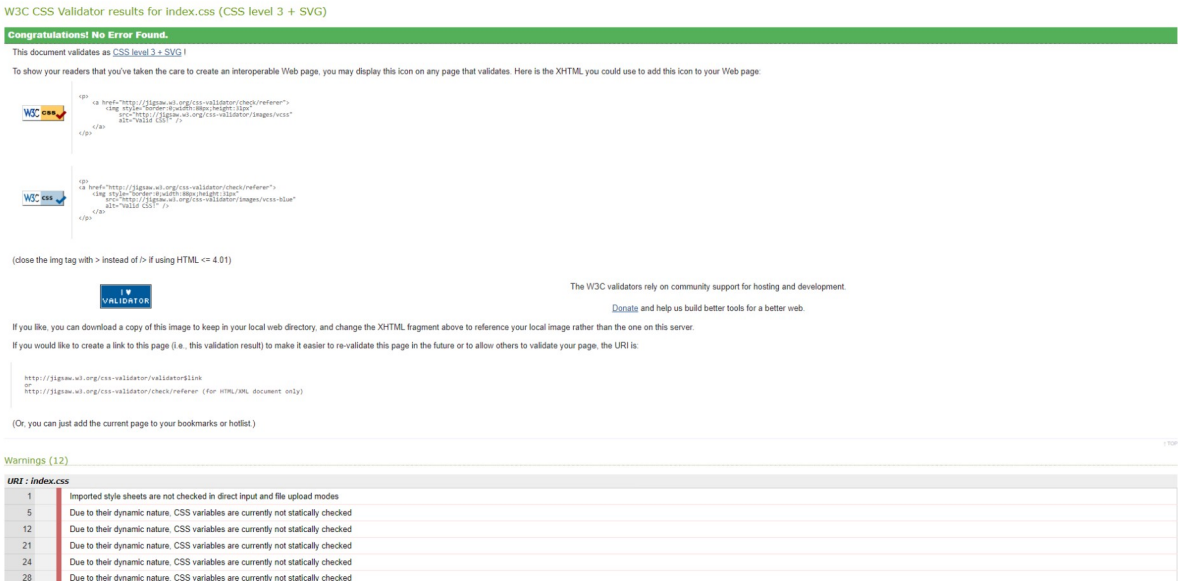
6.1 Automated Testing

6.1.1 W3 Nu Validator



The website's HTML has been validated using the W3 Nu Validator to ensure proper syntax and adherence to web standards.

6.1.2 W3 CSS Validation Service


W3C CSS Validator results for index.css (CSS level 3 + SVG)

Congratulations! No Error Found.

This document validates as CSS level 3 + SVG !

To show your readers that you've taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the XHTML you could use to add this icon to your Web page:

```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="Valid CSS!" />
    </a>
</p>
```

```
<p>
  <a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="Valid CSS!" />
  </a>
</p>
```

(close the img tag with > instead of /> if using HTML <= 4.01)

The W3C validators rely on community support for hosting and development.

Donate and help us build better tools for a better web.

If you like, you can download a copy of this image to keep in your local web directory, and change the XHTML fragment above to reference your local image rather than the one on this server.

If you would like to create a link to this page (i.e., this validation result) to make it easier to re-validate this page in the future or to allow others to validate your page, the URI is:

```
http://jigsaw.w3.org/css-validator/validator$link
or
http://jigsaw.w3.org/css-validator/check/referer (for HTML/XML document only)
```

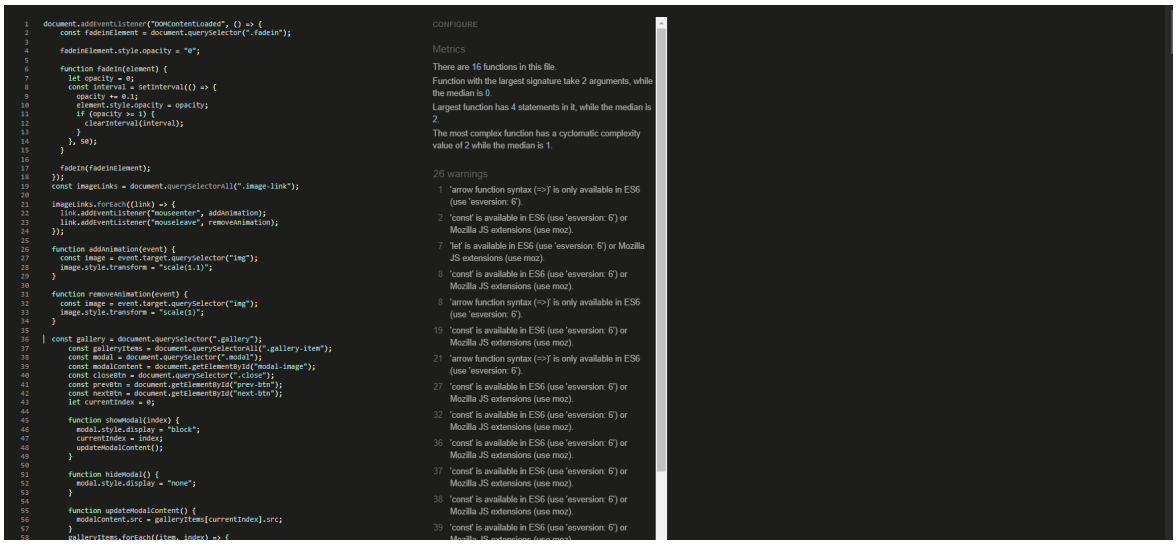
(Or, you can just add the current page to your bookmarks or hotlist.)

Warnings (12)

URI : index.css	
1	Imported style sheets are not checked in direct input and file upload modes
5	Due to their dynamic nature, CSS variables are currently not statically checked
12	Due to their dynamic nature, CSS variables are currently not statically checked
21	Due to their dynamic nature, CSS variables are currently not statically checked
24	Due to their dynamic nature, CSS variables are currently not statically checked
28	Due to their dynamic nature, CSS variables are currently not statically checked

The website's CSS has been validated using the W3 CSS Validation Service to ensure error-free code and compliance with CSS specifications.

6.1.4 JS Hint



```
document.addEventListener("DOMContentLoaded", () => {
    const fadeinElement = document.querySelector(".fadein");

    fadeinElement.style.opacity = "0";

    function fadeIn(element) {
      let opacity = 0;
      const interval = setInterval(() => {
        opacity += 0.1;
        element.style.opacity = opacity;
        if (opacity >= 1) {
          clearInterval(interval);
        }
      }, 50);
    }

    fadeIn(fadeinElement);
  });
  const imageLinks = document.querySelectorAll(".image-link");

  imageLinks.forEach((link) => {
    link.addEventListener("mouseenter", addAnimation);
    link.addEventListener("mouseleave", removeAnimation);
  });

  function addAnimation(event) {
    const image = event.target.querySelector("img");
    image.style.transform = "scale(1.1)";
  }

  function removeAnimation(event) {
    const image = event.target.querySelector("img");
    image.style.transform = "scale(1)";
  }

const gallery = document.querySelector(".gallery");
    const galleryItems = document.querySelectorAll(".gallery-item");
    const modal = document.querySelector(".modal");
    const modalContent = document.getElementById("modal-image");
    const closeBtn = document.querySelector(".close");
    const prevBtn = document.getElementById("prev-btn");
    const nextBtn = document.getElementById("next-btn");
    let currentIndex = 0;

    function showModal(index) {
      modal.style.display = "block";
      currentIndex = index;
      updateModalContent();
    }

    function hideModal() {
      modal.style.display = "none";
    }

    function updateModalContent() {
      modalContent.src = galleryItems[currentIndex].src;
    }
    galleryItems.forEach((item, index) => {
```

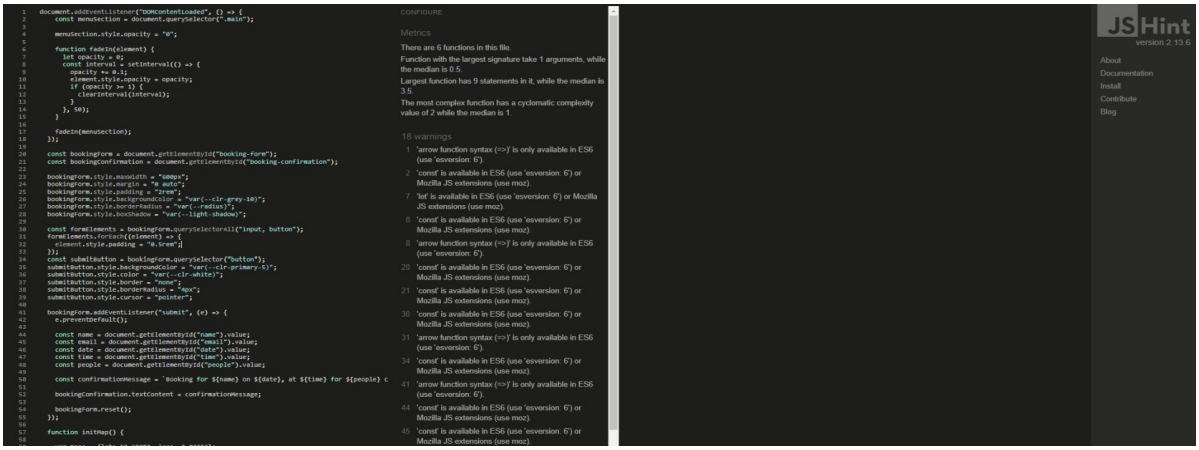
CONFIGURE

Metrics

There are 16 functions in this file.
Function with the largest signature take 2 arguments, while the median is 0.
Largest function has 4 statements in it, while the median is 2.
The most complex function has a cyclomatic complexity value of 2 while the median is 1.

26 warnings

1 'arrow function syntax (=>)' is only available in ES6 (use 'esversion: 6').
2 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
7 'let' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
8 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
8 'arrow function syntax (=>)' is only available in ES6 (use 'esversion: 6').
19 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
21 'arrow function syntax (=>)' is only available in ES6 (use 'esversion: 6').
27 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
32 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
36 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
37 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
38 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
39 'const' is available in ES6 (use 'esversion: 6') or



```
document.addEventListener("DOMContentLoaded", () => {
    const menuSection = document.querySelector(".main");

    menuSection.style.opacity = "0";

    function fadeIn(element) {
      let opacity = 0;
      const interval = setInterval(() => {
        opacity += 0.1;
        element.style.opacity = opacity;
        if (opacity >= 1) {
          clearInterval(interval);
        }
      }, 50);
    }

    fadeIn(menuSection);
  });

  const bookingForm = document.getElementById("booking-form");
  const bookingConfirmation = document.getElementById("booking-confirmation");

  bookingForm.style.maxWidth = "600px";
  bookingForm.style.margin = "0 auto";
  bookingForm.style.padding = "2rem";
  bookingForm.style.backgroundColor = "var(--clr-grey-10)";
  bookingForm.style.borderRadius = "var(--radius)";
  bookingForm.style.boxShadow = "var(--light-shadow)";

  const formElements = bookingForm.querySelectorAll("input, button");
  formElements.forEach((element) => {
    element.style.padding = "0.5rem";
  });
  const submitButton = bookingForm.querySelector("button");
  submitButton.style.backgroundColor = "var(--clr-primary-5)";
  submitButton.style.color = "var(--clr-white)";
  submitButton.style.border = "none";
  submitButton.style.borderRadius = "4px";
  submitButton.style.cursor = "pointer";

  bookingForm.addEventListener("submit", (e) => {
    e.preventDefault();

    const name = document.getElementById("name").value;
    const email = document.getElementById("email").value;
    const date = document.getElementById("date").value;
    const time = document.getElementById("time").value;
    const people = document.getElementById("people").value;

    const confirmationMessage = `Booking for ${name} on ${date}, at ${time} for ${people} c

    bookingConfirmation.textContent = confirmationMessage;

    bookingForm.reset();
  });

  function initMap() {
```

CONFIGURE

Metrics

There are 6 functions in this file.
Function with the largest signature take 1 arguments, while the median is 0.5.
Largest function has 9 statements in it, while the median is 3.5.
The most complex function has a cyclomatic complexity value of 2 while the median is 1.

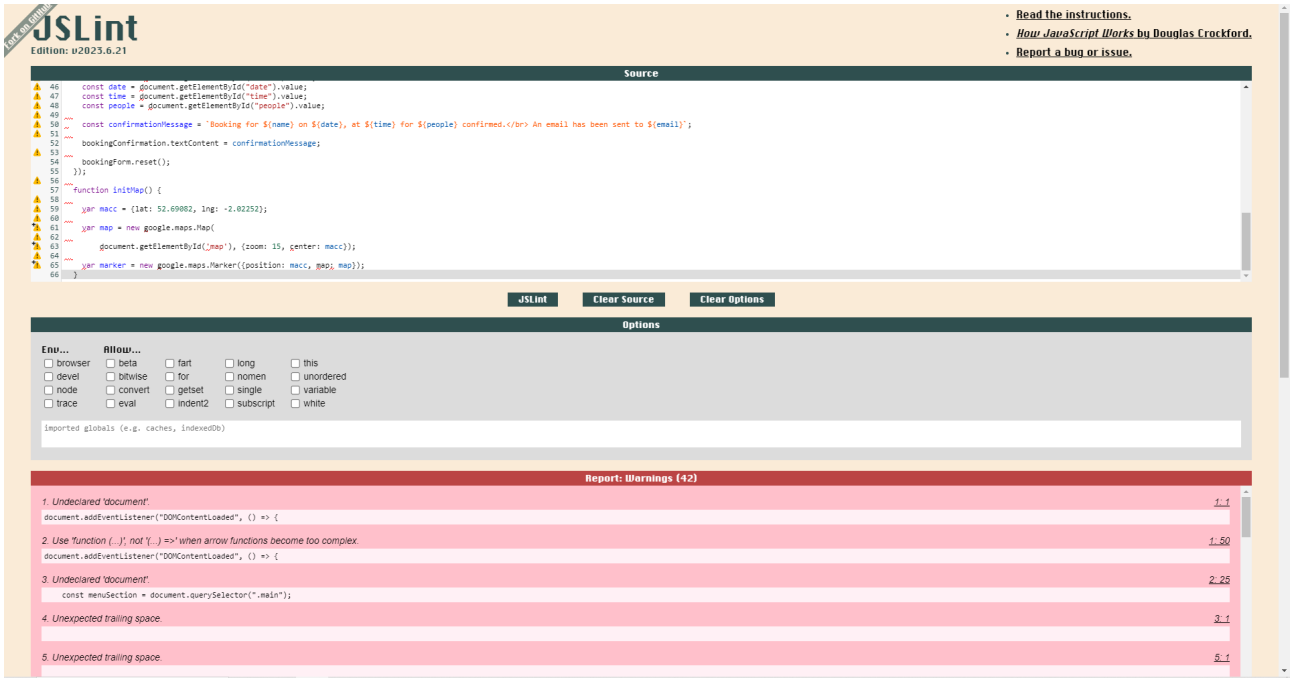
18 warnings

1 'arrow function syntax (=>)' is only available in ES6 (use 'esversion: 6').
2 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
7 'let' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
8 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
8 'arrow function syntax (=>)' is only available in ES6 (use 'esversion: 6').
20 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
21 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
30 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
31 'arrow function syntax (=>)' is only available in ES6 (use 'esversion: 6').
34 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
41 'arrow function syntax (=>)' is only available in ES6 (use 'esversion: 6').
44 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).
45 'const' is available in ES6 (use 'esversion: 6') or Mozilla JS extensions (use moz).

JSHint
version 2.13.6
About
Documentation
Install
Contribute
Blog

JavaScript code has been validated using JS Hint to identify and correct any potential errors.

JSLint



There are warnings but the JS code on all files seems to run fine.
6.2 Manual Testing

Extensive manual testing has been conducted to verify the website's functionality, responsiveness, and user experience. There was issues with the Google Maps API, which after a lot of trial and error managed to sort it out.

6.3 Bugs

All identified bugs and issues have been documented and addressed during the testing phase. The only one that might persist might be the Google Maps API which requires the restrictions to be added into the key propperties.

Credits

7.1 Code Sections

John Smilga for the Menu Template alongside the colour palette for certain ellements.

7.2 Media

- Stock Photos (google.com)
- [Insert Image Source/Author]

7.4 Acknowledgements
-Google (for lots of info, stock photos, color pallets)
-Coolor

-VSCode
-My friend Assel and Spase for supporting through the project.