



Module Title

Internet Technologies

Assessment Weightage & Type

40% of Coursework & Regular

Year

2025

Student Name: NIRVIK K.C.

UWE ID: 25024649

Assignment Due Date: July 10, 2025

Assignment Submission Date: July 10, 2025



Assessment Brief

Module Code	UFCFL1-15-0
Assessment Name	Component B
Module Name	Internet Technologies
Year	2025
Weighting	40%

Table of Content

Contents

INTRODUCTION	4
Travel Website	7
OBJECTIVE	12
WEB DESIGN	13
Wireframes:	15
IMPLEMENTATION	20
Programming Concepts Used:	21
Form Validation (Contact Page):	24
Classes, Methods, Functions:	25
WEB DISPLAY	28
TESTING	31
DEBUGGING	33
CHALLENGES FACED	35
CONCLUSION	36
REFERENCES	37

INTRODUCTION

A website consists of web pages written in html code, and an accessible site can be accessed by anyone in the world if it is stored or hosted on a computer that is connected to the internet called a web server.

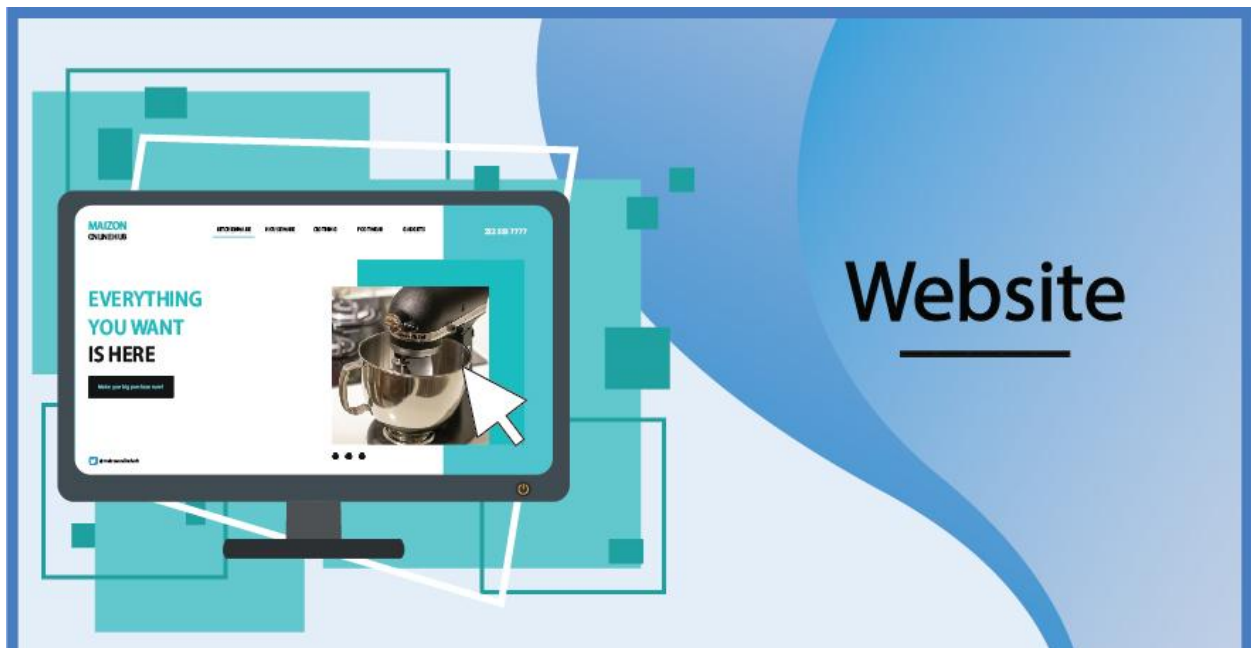


Figure 1: Meaning of Website (Template.com)

A website is like a digital space that connects web pages using links. It usually has a similar look and feel across all pages and can include stuff like pictures and videos. As the internet gets bigger, websites do all sorts of things. They give you info, let you solve problems, and help people, places, and things talk to each other – all to help the group that made the site reach its goals.

Components of Website:**a) Webhost:**

Web hosting is basically where your website located. A bunch of linked web pages only counts as a real website if it's actually hosted on a web server. The web server is just a collection of files that gets sent to people's computers when they type in your website's address.

b) Address:

Address, that's also known as the URL. When a user wants to check out a website, he/she needed to type the URL into your browser, and the web server sends you the site.

c) Homepage:

Home Page is a very common and important part of a webpage. The home page is the first page that called up when guests visit the website. The home page of a website is important because it creates the look and feel of the website and directs viewers to all the other pages on the website.

d) Design:

The design is the overall appearance and feel of the website that is the result of a correct use and integration of elements such as navigation menus, graphics, layout, navigation menus, etc.

e) Content:

All the pages that are contained within the website make up the content of the website. Good content on the web pages will make the website perform more effectively and attractively.

f) The Navigation Structure:

The navigation structure is the order of the pages, what links to what, and is usually held together by at least one navigation menu.

Travel Website

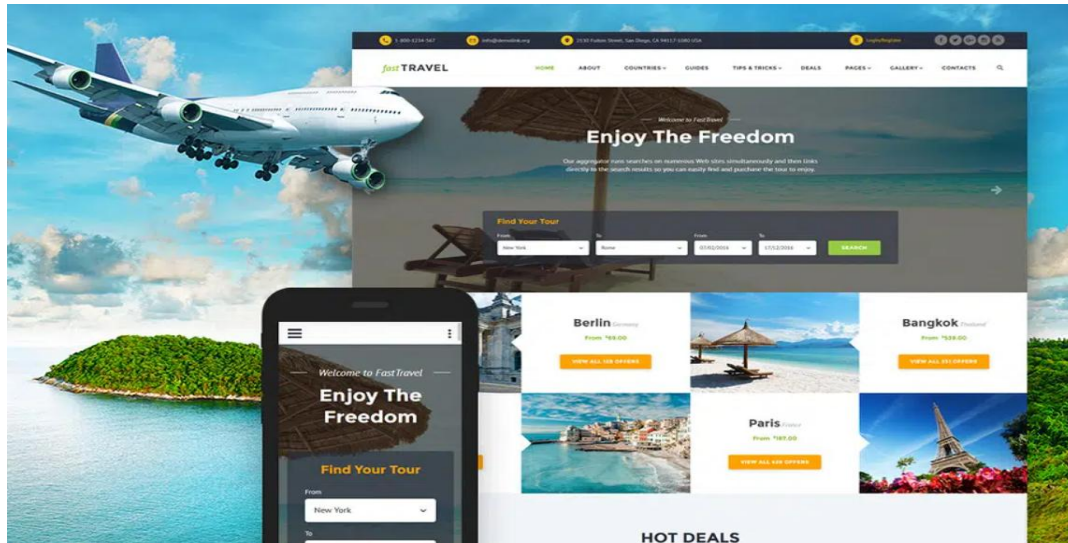


Figure 2: Responsive Travel Website (WP Travel)

A travel website is a site that offers travel information and services (e.g. booking flights, hotels, or rental cars). It is a virtual location where individuals can plan, book, and enjoy their travel. There are several types of travel websites - Online Travel Agencies (OTAs), Service Provider Websites, Fare Aggregators/Metasearch Engines, Travel Blogs and Guides, and Niche Travel Websites.

An OTA provides all the components in one place (like Expedia, Booking.com, and Priceline), while Service Provider Websites are all the individual websites for airlines, hotels, cruise lines, or rental cars. Fare Aggregators/Metasearch Engines provide price comparisons between different OTAs and airlines. Travel Blogs and Guides includes content that focuses on the author's experiences , suggestions, knowledge of a particular destination, and travel tips.

Features of a Travel Website

Today's travel websites are user-friendly for planning trips. They allow you to search, have an online booking system, provide photos, explain services in text, have user reviews, have interactive maps, and are mobile-friendly.

They can also allow you to securely pay online, offer membership accounts, provide customer service, integrate with social media, have real-time availability and price, allow for multiple currency and language use, have travel blogs and articles, and have deal/discount sections. These websites are intended to combine all information, user reviews, interactive maps, secure online payment processing, user-friendly sharing, and support multiple currencies and languages in order to give a global audience a part in these excellent travel planning web resource.

Advantages of a Travel Website

Travel websites provide a number of advantages for travelers and businesses. They truly are beneficial for travelers, providing convenience, value, choice, customized trips for travelers, and informed decisions leaving travelers and organizations with pieces of mind.

Travel websites for businesses also assist in expanding reach and credibility and availability. They provide cost-effective marketing solutions, direct bookings, lower commissions, and increased efficiencies for travel organizations. Travel websites for services allow for direct bookings and less reliance on OTAs.

Travel websites also provide opportunities for client feedback and improvements, quicker distribution of information, engaged community service, and information analytics for data driven business decisions. In conclusion, travel websites provide both travelers and businesses with a multifaceted offering.

To sum up, travel websites have changed how to plan for travel and how we can make bookings, giving us incredible convenience, a huge range of selection, and competitive prices. These sites provide travelers with everything from detailed reviews and trip ideas to travelers to feel good about their decisions and travel with ease.

Use of Programming Language

The project showcases the development of a Student Profile Management System using html, CSS and JavaScript.

HTML (Hyper Text Markup Language)

HTML is the basis for several websites (simple web pages to complex web pages) originally developed by Berners-Lee in 1991. HTML history includes its foundation game in 1991 to its enhancements through versions from HTML 2.0, HTML 4.01 through HTML 5 (the current version, semantic tags, and various required attributes).

Framework of HTML:

The `<!DOCTYPE>` declaration shows the version of HTML used; the `<html>` element contains the overall document which has 2 main sections: head and body.

The `<head>` element contains information about the document, including title, external stylesheets, scripts, and tags, and the `<body>` element contains the content, including text, images, and video.

CSS (Cascading Style Sheets)

CSS technology, or Cascading Style Sheets, is a tool that makes HTML code more usable and aesthetically pleasing. CSS offers a variety of design and creative capabilities that enhance any web page, e.g., color, alignment, positioning, responsive, font size, design, animations, etc. CSS provides the ability to create websites that adapt to various screen sizes without the need to create separate websites for each screen size. CSS ensures that the same websites fit different screen sizes.

JavaScript

JavaScript is a powerful language. It has the power to add interactivity to static websites which have been built using HTML and CSS, applies dynamic styles to page elements, dynamically add or remove HTML elements, and so much more. For example, every time you press “Submit” on an empty form, and then you get that error message, you should know that’s JavaScript behind the scenes.

JavaScript is an important instrument in modern web development, allowing developers to build dynamic and interactive web pages. Developers can use JavaScript to create an interactive experience for users, such as animations, form validation, and many other features that make web pages interactable. JavaScript can also retrieve and manipulate data from APIs, allowing web applications to connect with third-party services and APIs. In basic terms, JavaScript is a powerful tool for engaging users, prompting them to do things on the web.

OBJECTIVE

- To create wireframes to visualize the layout and user experience.
- To implement responsive design to ensure the website is mobile-friendly and accessible across various devices.
- To create a visually appealing and intuitive UI (User Interface) that aligns with the identity of travel theme.
- To design a logical navigation hierarchy that allows users to move different pages with minimum clicks.
- To integrate design for social media icons which links to GoTravel's profiles like Facebook, Instagram, Twitter, Pinterest.
- To determine essential features in the website such as search functionality, booking options, and user testimonials.
- To ensure high-quality images and videos are available for visual appeal to the user.
- To focus on user experience by optimizing the loading speed and performance of the website.
- To develop a content plan that includes descriptions for destinations, services, and testimonials.
- To execute a comprehensive project using the concepts gathered during the course duration.

WEB DESIGN

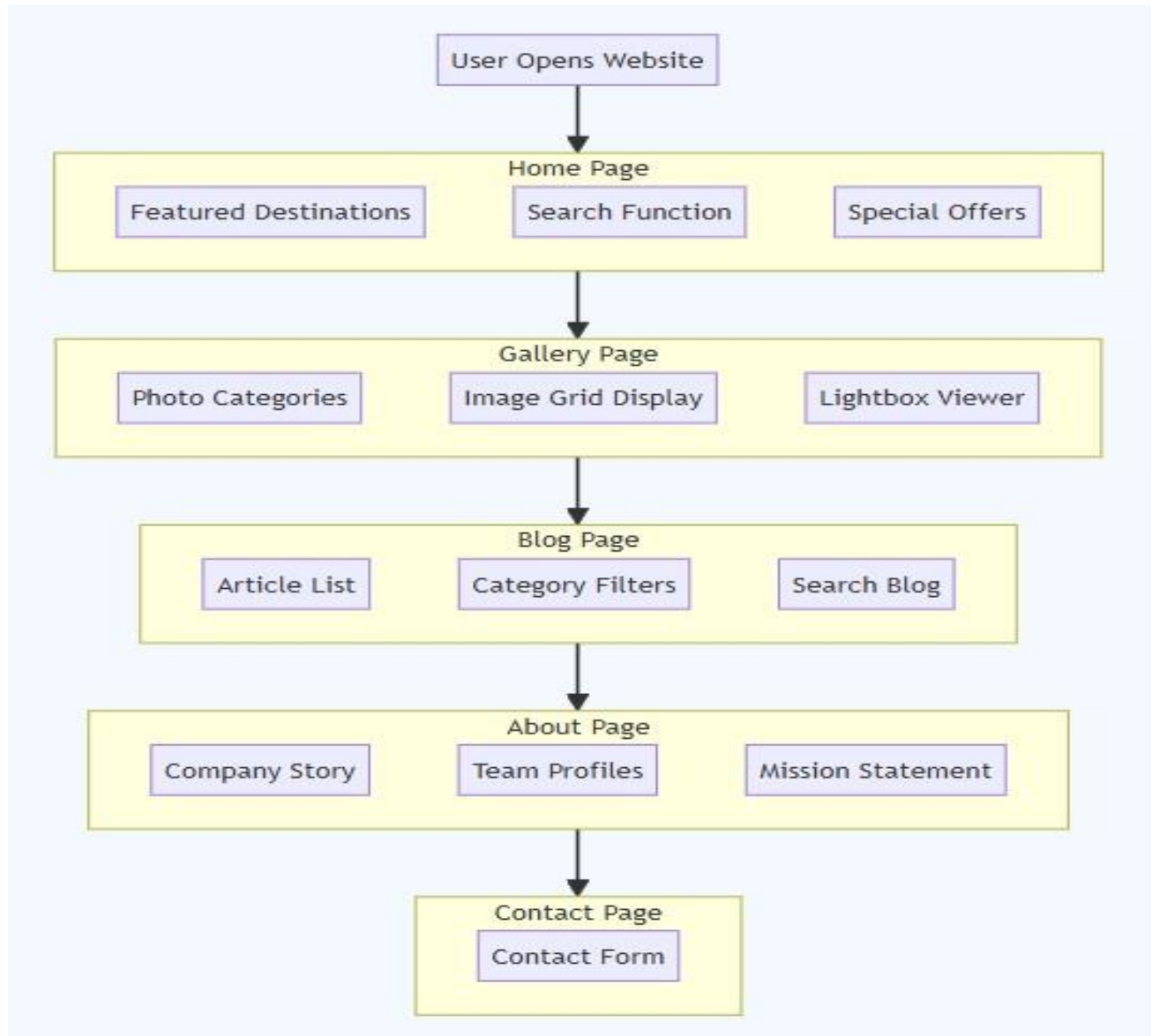


Figure 3: Web Design for Travel Website (HIX AI)

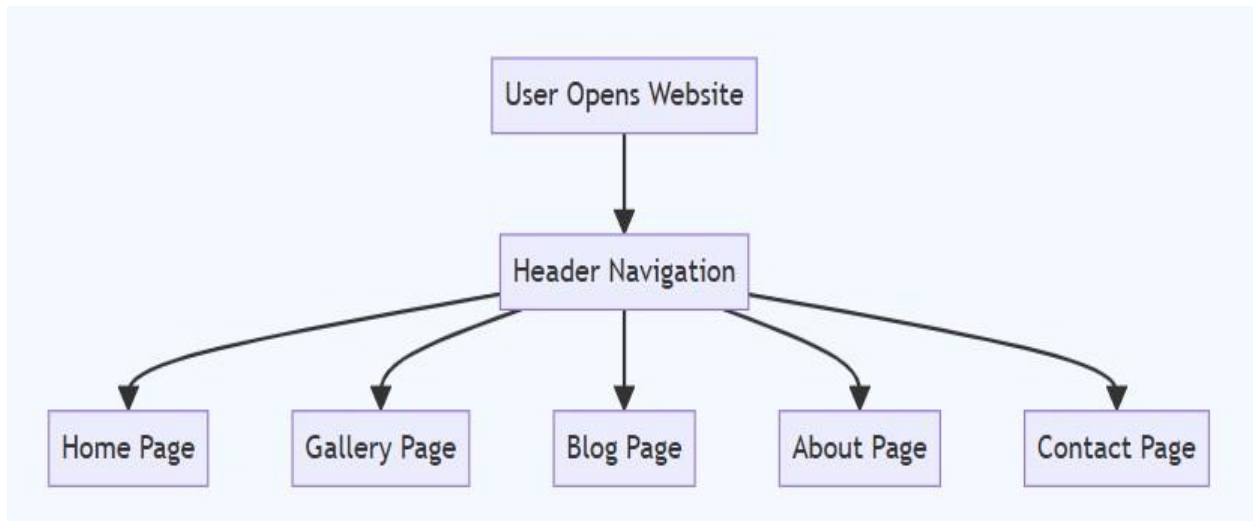


Figure 4: User Navigation Menu Design (HIX AI)

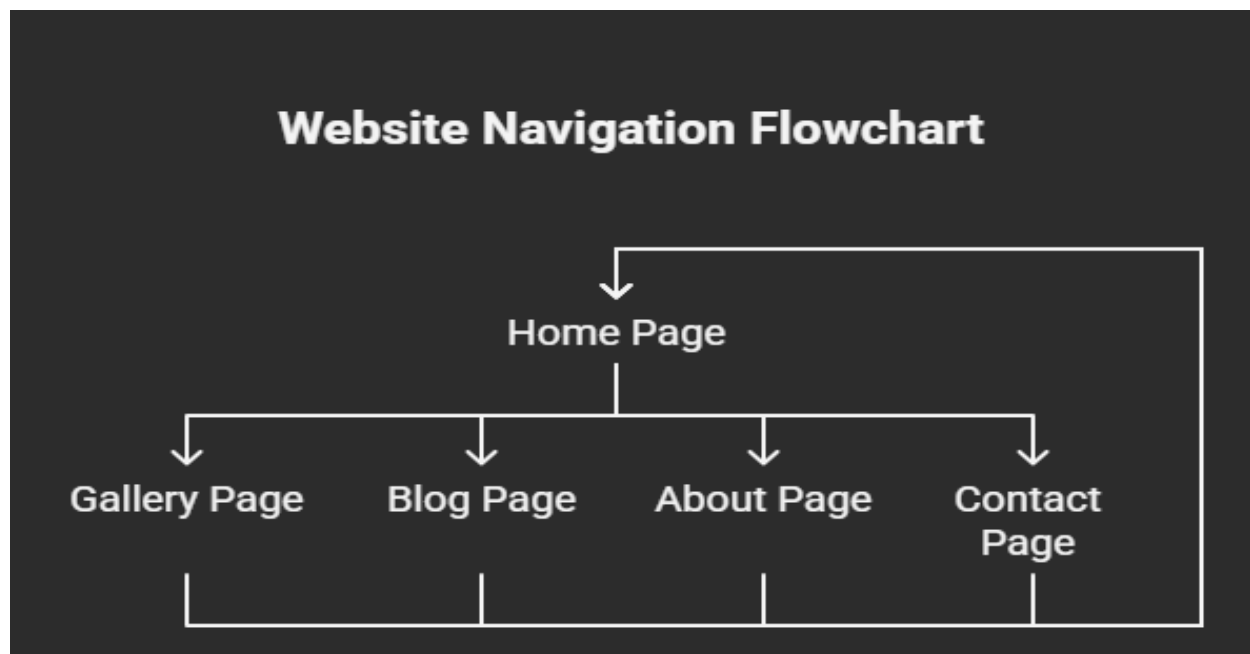


Figure 5: Flowchart for flow of entire program for whole website (Napkin AI)

Wireframes:



Figure 6: Wireframe for Home Page (index.html)

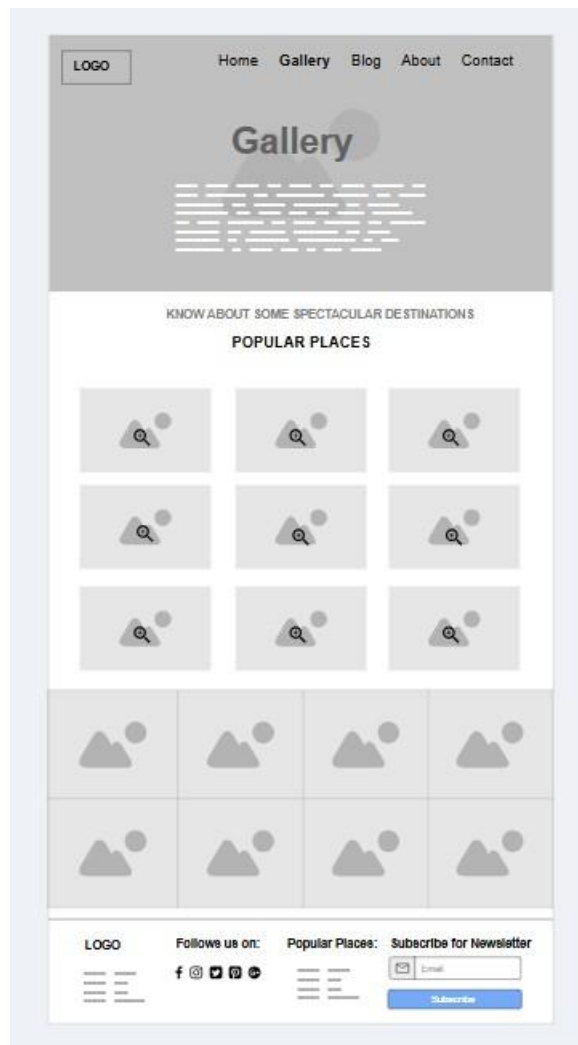


Figure 7: Wireframe for Gallery page (gallery.html)

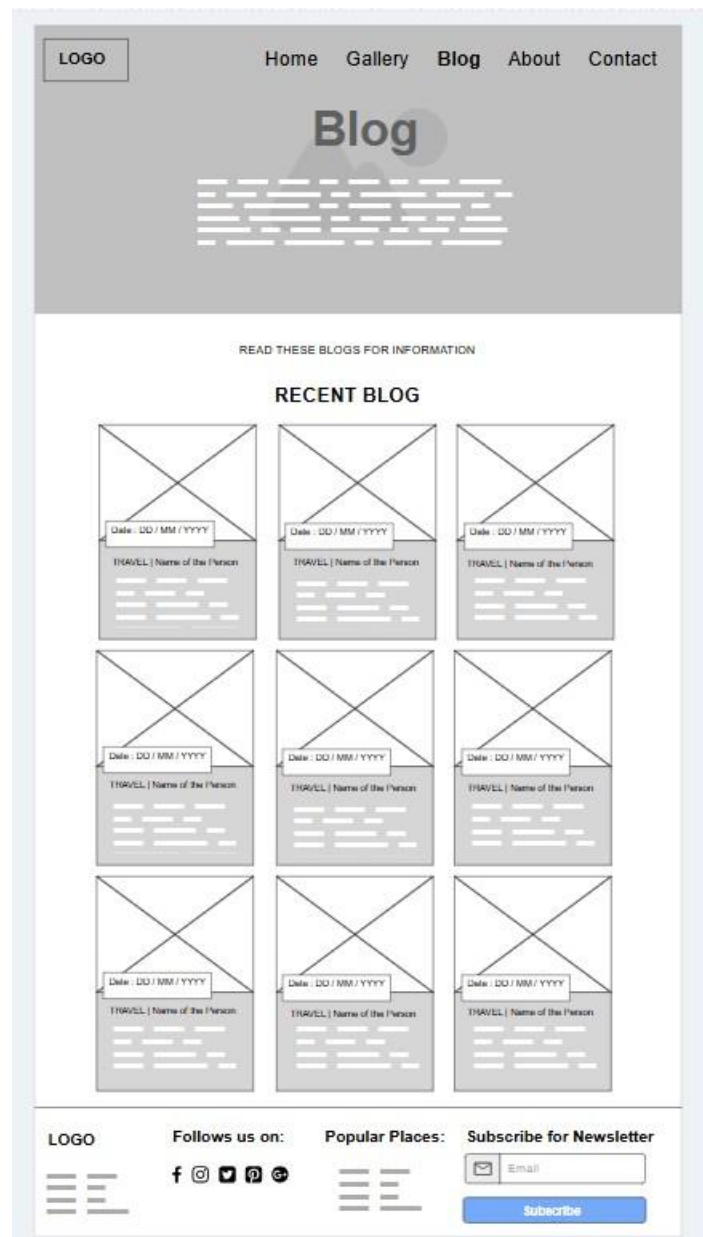
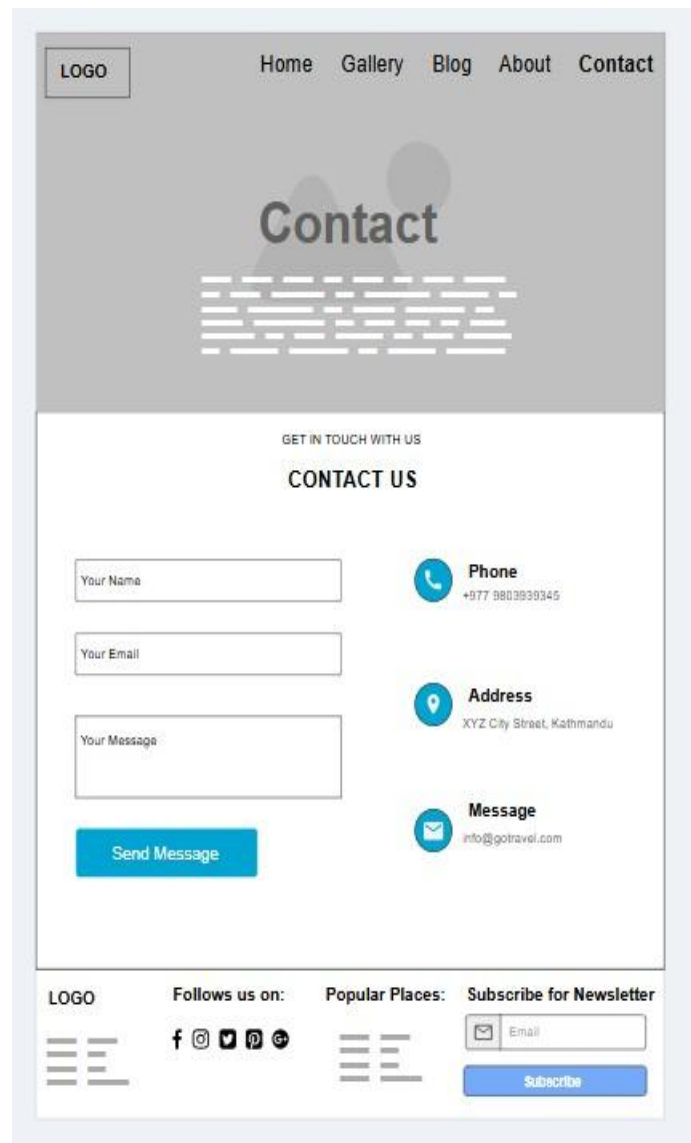


Figure 8: Wireframe for Blog Page (blog.html)



Figure 9: Wireframe for About Page (about.html)



The wireframe illustrates a contact page layout. At the top, a navigation bar includes a 'LOGO' placeholder and links for 'Home', 'Gallery', 'Blog', 'About', and 'Contact'. The main header features a large 'Contact' title and a background graphic of two stylized figures. Below this, a section titled 'GET IN TOUCH WITH US' and 'CONTACT US' contains a contact form and contact details. The form has fields for 'Your Name', 'Your Email', and 'Your Message', along with a 'Send Message' button. To the right, contact information is listed: 'Phone' (+977 9803939345), 'Address' (XYZ City Street, Kathmandu), and 'Message' (info@gotravel.com). The footer includes a 'LOGO' placeholder, social media links ('Follows us on:'), a 'Popular Places:' section, and a 'Subscribe for Newsletter' form with an email input and a 'Subscribe' button.

LOGO Home Gallery Blog About Contact

Contact

GET IN TOUCH WITH US

CONTACT US

Your Name

Your Email

Your Message

Send Message

Phone
+977 9803939345

Address
XYZ City Street, Kathmandu

Message
info@gotravel.com

LOGO Follows us on: Popular Places: Subscribe for Newsletter

f @ t p g+

Email

Subscribe

Figure 10: Wireframe for Contact Page (contact.html)

IMPLEMENTATION

HTML, CSS, and JavaScript can be executed using integrated environment (IDE) for writing and execution of code. So, I have using Visual Studio Code for writing all three languages.

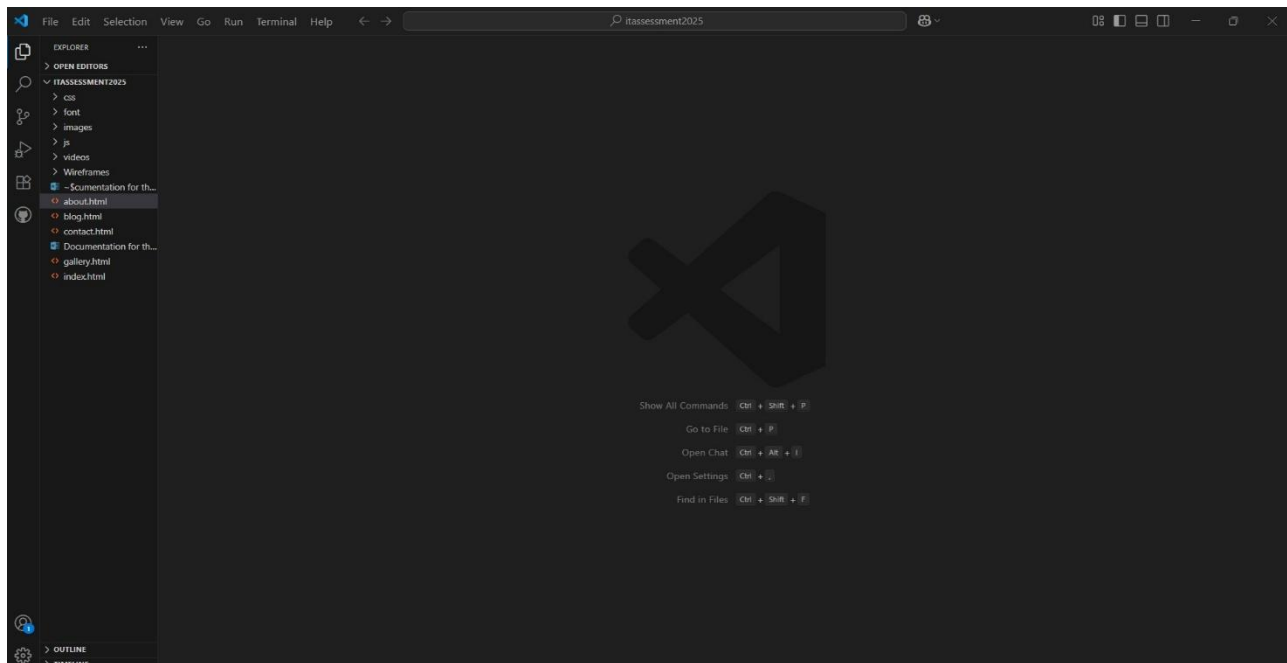


Figure 11: Visual Studio Code for writing HTML, CSS, and JavaScript and organizing other files

Programming Concepts Used:

HTML

Structure and Content:

HTML is used to define the basic structure and content of each web pages in the website (like ‘index.html’, ‘about.html’, ‘gallery.html’, ‘blog.html’, ‘contact.html’).

Elements:

Elements like ‘<header>’, ‘<nav>’, ‘<section>’, ‘<footer>’, ‘<div>’, ‘<h1>’ to ‘<h6>’, ‘<p>’, ‘’, ‘’, ‘<a>’, ‘’, ‘<form>’, ‘<input>’, ‘<textarea>’, ‘<button>’, and ‘<video>’ are used to organize content semantically.

Forms:

The ‘contact.htm’ and ‘index.html’ files include ‘<form>’ elements with various input types (‘text’, ‘email’, ‘date’, ‘number’, ‘submit’, ‘textarea’) for user interaction.

CSS(Cascading Style Sheets):

Layout:

Flexbox(‘display: flex’) is utilized for creating responsive layouts, especially for navigation, headers, and content rows. For e.g., ‘navbar’, ‘.container. flex’, ‘header-form’, ‘about-row’, ‘.facts-row’, ‘.blog-row’, ‘.gallery-row’, ‘popular-row’, ‘.services-row’, ‘test-row’, ‘.footer-row’).

Typography:

Font families ('font/fonts.css'), sizes, weights, and colors are defined.

Colors and Backgrounds:

Setting the backgrounds colors (aqua, and white), text colors (grey, white, black, and blue) and hover effects.

Spacing:

Margins('my-2' and 'py-4') and padding are used for spacing elements.

Responsiveness:

Implementation of 'responsiveness.css' for media queries to adapt the layout and styling for different screens and making sure that the website is mobile-friendly.

Utility Classes:

Classes like 'flex', 'shadow', 'py-4', 'my-2', 'sm-title', 'lg-title', 'text', 'btn', and 'form-control' are used to apply styles across multiple elements.

External Stylesheets:

Styles are linked from external '.css' files ('normalize.css', 'utility.css', 'style.css', 'responsiv.css' for better organization and maintainability.

Font Awesome:

An external CDN link to include Font Awesome icons stored in 'fonts.css' ('<i class= "fas fa-bars">', '<i class= "fab fa-facebook-f"></i>', etc.) for visual enhancement.

JavaScript:

JavaScript interacts with the Document Object Model (DOM) to dynamically change the content and the style of HTML elements.

Navbar Toggle:

The 'script.js' handles the functionality for showing and hiding the mobile navigation menu using 'navbar-show-btn' and 'nav-close-btn' IDs.

Image Modal (Gallery Page):

Event listeners are attached to the gallery item (.gallery-item') to open a modal ('#img-modal-box') when clicked on. The 'showImageContent' function sets the 'src' property of the image in the modal to match the image's index that was clicked. Both 'next-btn' and 'prev-btn' move between the images shown in the modal and adjusts the 'imgIndex' and calls 'showImageContent'. Image modal 'modal-close-btn closes' the modal.

Event Handling:

The 'addEventListener' is used to react to user interactions, such as mouse clicks ('click') or form submission ('submit').

Form Validation (Contact Page):

It prevents the default form submission (`e.preventDefault()`). It checks each of the name, email, and message fields for empty values. It validates email format using a regular expression (`validateEmail` function). It presents error messages in a dedicated `'error-message'` div. An `'alert'` is shown if the form is successfully submitted (the form is not submitted to any server).

Error Handling:**Client-Side-Form Validation:**

The `'contact.html'` implements client-side error handling for its contact form:

- a) It checks for empty fields (name, email, message).
- b) It validates the email format.
- c) In the case of a validation failure, it returns a red error that is shown inside the `'div'` with `'id="error-message"'`.
- d) The `'return;'` statement will stop execution of the submission logic if an error has been found.

There is not server-side error handling as this is a static HTML/CSS/JS website, there is no server-side code, and thus no server-side error handling for form submissions.

Classes, Methods, Functions:

Classes (CSS):

(e.g., ‘navbar’, ‘container’, ‘flex’, ‘site-brand’, ‘nav-item’, ‘nav-link’, ‘header-sm’, ‘header-title’, ‘header-form’, ‘form-control’, ‘tn’, ‘title-wrap’, ‘sm-title’, ‘lg-title’, ‘about-row’, ‘about-left’, ‘about-right’, ‘text’, ‘facts-row’, ‘facts-item’, ‘facts-icon’, ‘facts-info’, ‘blog-row’, ‘blog-item’, ‘shadow’, ‘blog-date’, ‘blog-item-top’, ‘blog-item-bottom’, ‘gallery-row’, ‘gallery-item’, ‘zoom-icon’, ‘popular-row’, ‘popular-item’, ‘rating’, ‘services-row’, ‘services-item’, ‘services-icon’, ‘test-row’, ‘test-item’, ‘test-item-info’, ‘video-wrapper’, ‘footer-row’, ‘footer-item’, ‘social-links’, ‘subscribe-form’, ‘error-message’).

Functions (JavaScript):

‘showImageContent(index)’: (In ‘gallery.html’)

This function takes the index passed to it and updates the src of the image in the modal accordingly.

validateEmail(email): (In contact.html)

This function takes an email string and checks if it is in a valid email format using a regular expression, and returns either true or false.

Anonymous Functions (Event Listeners):

The event listeners use anonymous functions often (e.g.,
 contactForm.addEventListener('submit', (e) => {...}); ,
 ‘nextBtn.addEventListener('click', () => {...});’).

Methods(JavaScript):

`document.querySelector()`:

Used to select the first element matching a specified css selector (e.g. `document.querySelector('.contact-form')`, `document.querySelector('video-wrapper video')`).

`document.querySelectorAll()`:

Used to select all elements matching a specified css selector (e.g. `document.querySelectorAll('gallery-item')`).

`document.getElementById()`:

Used to select an element by its id (e.g. `document.getElementById('error-message')`, `document.getElementById('play-btn')`).

`element.addEventListener()`:

Attaches an event handler to an element (e.g. `contactForm.addEventListener('submit', ...)`).

`element.addEeventListerner()`:

Attaches an event handler to an element (e.g., `contactForm.addEventListerner('submit', ...)`).

`e.preventDefault()`: (event object method) Prevents the default action of an event, e.g. submitting a form.

`element.style.display` = "block" / "none": Directly manipulate the css display property of an element to display or hide it.

`‘element.textContent’= ‘...’`: Gets or sets the text content of an element.

`‘element.value.trim()’`: Gets the value of an input and trim whitespace from the ends.

`‘String.prototype.toLowerCase()’`: Converts a string into lowercase.

`‘RegExp.prototype.test()’`: Tests for a match in a string.

`‘video.play()’` / `‘video.pause()’`:

Methods of HTML 5 `‘<video>’` element to control playback.

`‘parseInt()’`

Parses a string argument and return an integer.

`‘string.prototype.match()’`:

Retrieves the result of matching a string against a regular expression.

WEB DISPLAY

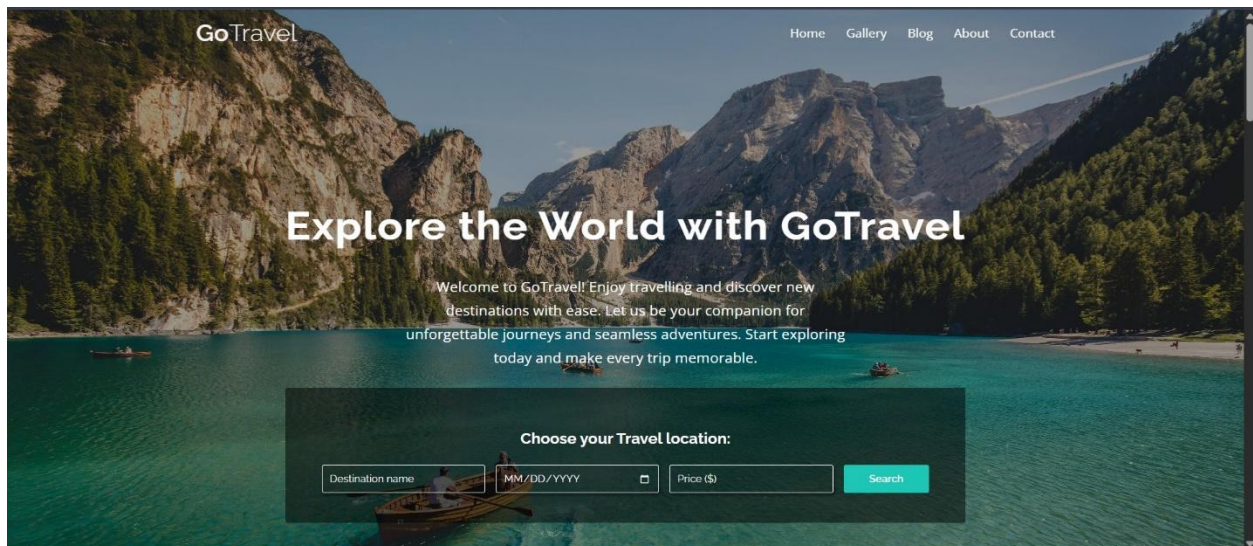


Figure 12: Home Page View

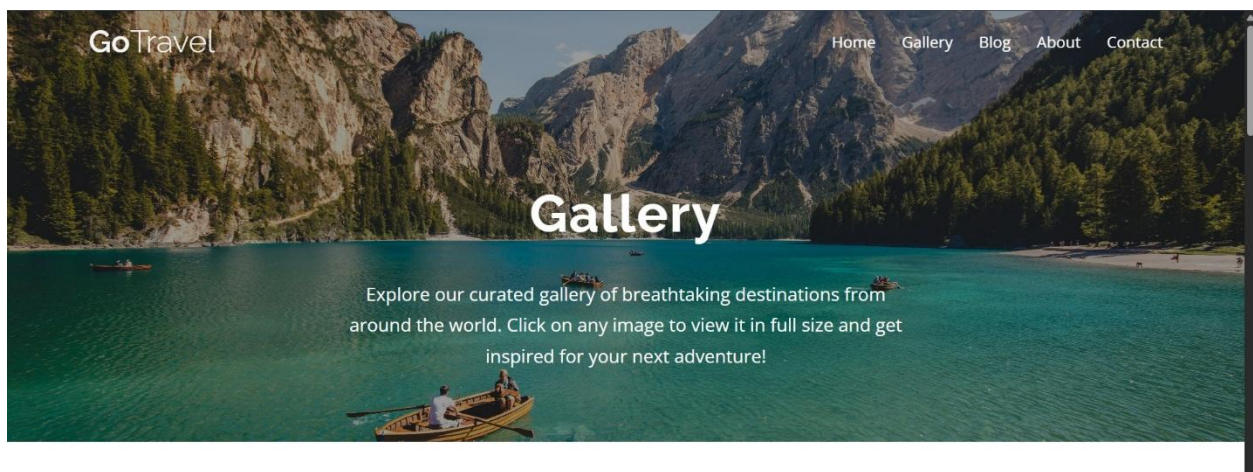


Figure 13: Gallery Page View

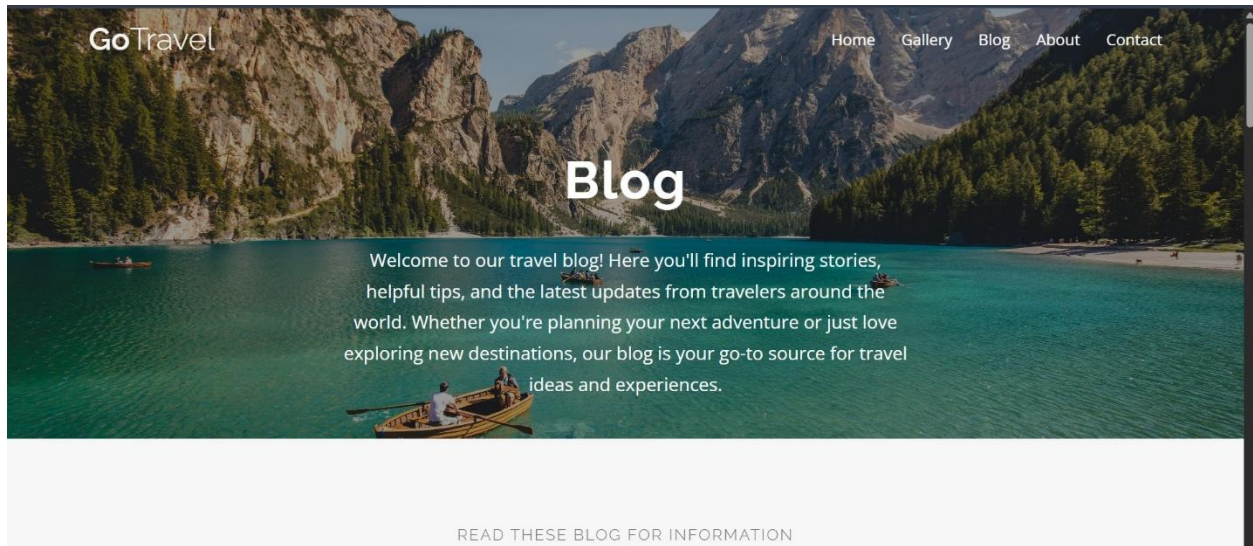


Figure 14: Blog Page View

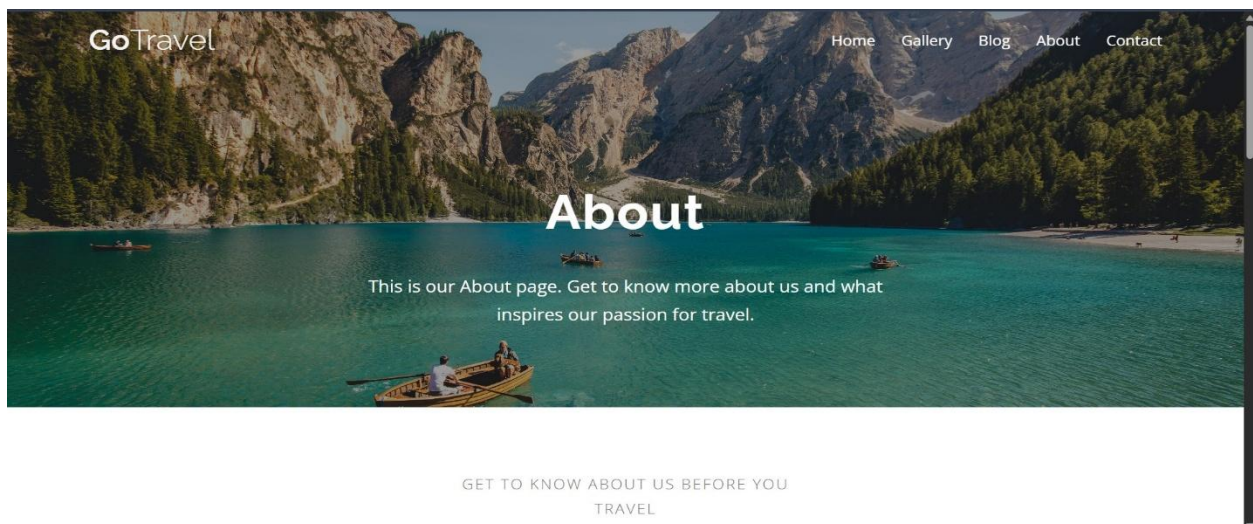


Figure 15 : About Page View

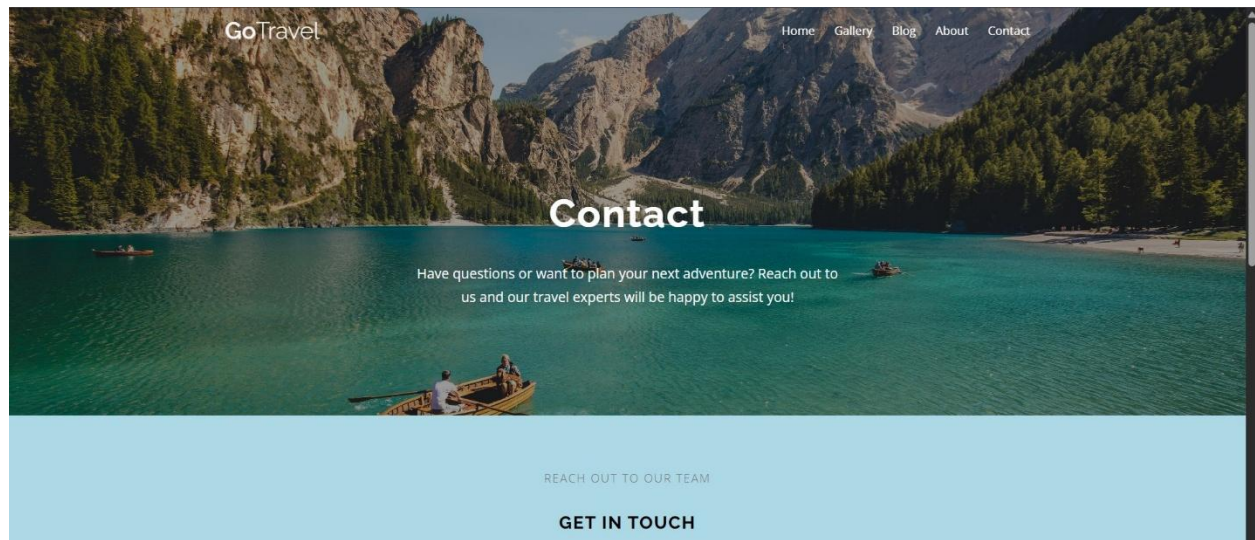


Figure 16: Contact Page View

TESTING

Testing

Manual Testing:

For this kind of website, the primary manner of testing would be manual testing:

Browser Compatibility:

Check how the website renders and has functionality in different web browsers (Chrome, Firefox, Edge, Safari).

Responsiveness:

Resize the browser window or use developer tools to emulate the different sizes of devices, and ensure the responsive.css styles apply correctly.

Navigation:

Click on all the navigation links in the navbar and ensure they navigate to the correct pages.

Interactive Items:

Testing the navbar toggle in mobile views.

Testing the image modal in the gallery of images:

Opening, closing, and navigating Next/Previous. Testing the play/pause button for the video.

Testing contact form validation:

submitting with missing empty fields, an invalid email, and complete valid data, to observe error messages and success alerts.

Content Review:

Check all the text for typos, and confirm that images load.

No Automated Testing Frameworks:

In this case, given the simplicity of the project, I have no evidence of automated testing frameworks (Jest, Cypress, Selenium, etc.).

DEBUGGING

Browser Developer Tools:

The most common and effective debugging tool for this or similar projects is the browser built-in developer tools (e.g., the Chrome DevTools or Firefox Developer Tools):

Elements Tab:

Inspect the HTML structure of the page, and investigate the CSS styles that are applied.

Console Tab:

Check for JavaScript errors and check the `console.log()`'s which will report variable values, and check for network requests.

Sources Tab:

Use breakpoints in the JavaScript code to be able to step through execution, and look at variable states.

Network Tab: Check resources are loading (images, CSS, JS files) and check their request response status codes.

Responsive Design Mode:

Emulate different screen sizes and orientations.

`console.log()`:

Developers will be likely to use or have `console.log()` statements placed in the javascript to "print", or output variable values or to verify certain code paths have occurred.

Errors:

the client-side form validation provides in real time feedback to the user about their input errors.

CHALLENGES FACED

- Difficulty in planning and web design.
- Understanding flow of the program.
- Creating an intuitive and visually appealing design that enhances user experience can be challenging.
- Ensuring the website is responsive and works well on various devices (desktops, tablets, smartphones) can be complex.
- Generating high-quality, engaging content (text, images, videos) that resonates with the target audience.
- Ensuring the website loads quickly and efficiently can be challenging with large images.

CONCLUSION

The process of development within the travel website included building wireframes to map out the designs of each page, which included navigation, content areas, social media, and anything else that was going to be interactive for the user. This stage in the process was crucial for identifying usability problems and ensuring that we were meeting the needs of our target audience. Once the wireframes were developed and I was able to progress forward to developing and creating easy navigation, accessibility, and a user-friendly interface.

The wireframes were converted into working web pages using HTML, CSS and JavaScript. HTML provided structure for the website, aiding in accessibility and SEO. CSS was used to style the website to match the travel agency's branding. SASS features, Flexbox, and media queries were used to create a responsive layout with CSS. JavaScript was used in this stage of development to create interactivity and loads of functionality, including image modals, form validation in Contact Page, and content updates.

This travel agency web design focuses on aesthetics and functionality. The website displays travel packages and services. The design includes colorful imagery, interesting typography, and engages the user with intuitive navigation. This project illustrates how a mindful approach to design and development can assist in building a successful real-world project leading to the growth in the particular field.

REFERENCES

Anon. (no date) *Scaler Topics* [online]. Available from:

<https://www.scaler.com/topics/javascript/> [Accessed 4 July 2025].

Anon. (2023) *template.net*. 10 January 2023 [online]. Available from:

<https://www.template.net/graphic-design/website/> [Accessed 4 July 2025].

Chhetri, Y. (2023) Top 16 Must-have key features for travel websites (2024) *WP Travel*. 13 March 2023 [online]. Available from: <https://wptravel.io/top-features-for-travel-agency-website/> [Accessed 4 July 2025].

Anon. (2021) *GeeksforGeeks*. 5 July 2021 [online]. Available from:

<https://www.geeksforgeeks.org/blogs/what-is-a-website/> [Accessed 5 July 2025].

Anon. (2023) *GeeksforGeeks*. 4 December 2023 [online]. Available from:

<https://www.geeksforgeeks.org/html/html-tutorial/> [Accessed 5 July 2025].

Anon. (2023) *GeeksforGeeks*. 11 December 2023 [online]. Available from:

<https://www.geeksforgeeks.org/css/css-tutorial/> [Accessed 5 July 2025].

Anon. (no date) *Napkin.ai* [online]. Available from:

<https://app.napkin.ai/page/CgoiCHByb2Qtb25lEiwKBFBhZ2UaJDJjMTlhMzM3LTg5NTgtNGY3Yy05M2IwLWI2YWQxZWQ5YTdkZg> [Accessed 5 July 2025].