Report on Travel Website Project

# 1. Introduction

The purpose of this project was to create a travel website using HTML and CSS, with a focus on user experience, accessibility, and responsive web design. The site aims to provide users with an easy-to-navigate interface for exploring travel destinations, booking services, and accessing travel-related information. By using semantic HTML and modern CSS techniques, the site ensures an accessible and responsive experience across different devices and screen sizes.

# 2. Case Study 1: Real-World Website Overview

For the first case study, I analyzed the popular travel booking website Expedia.com. Expedia offers a user-friendly layout with clear navigation, ensuring easy access to travel services like flights, hotels, and car rentals. The use of dropdown menus, search bars, and buttons makes it intuitive for users. The site also incorporates accessibility features, such as alt text for images, descriptive labels for forms, and keyboard navigation support, which helps users with disabilities access the site easily.

# 3. Key HTML Elements for Accessibility

In building my travel website, I followed several best practices to ensure accessibility for all users, including those with disabilities:

• Alt attributes for images: Each image has descriptive alt text to ensure that screen readers can interpret visual content.

• Semantic HTML: I used elements like <header>, <nav>, <main>, and <footer> to structure the page meaningfully for screen readers and assistive technologies.

• Form labels: Each input field in forms is properly labeled using the <label> element to associate text descriptions with form controls.

• ARIA roles: Where necessary, I used ARIA roles and attributes to provide additional context to assistive technologies.

# 4. CSS Enhancements for Visual Impairments

To make my website more accessible to visually impaired users, I applied several CSS techniques:

• High contrast colors: I ensured that the text contrasts sufficiently against the background to make it easier for users with low vision.

• Scalable fonts: I utilized relative units like 'em' and '%' instead of fixed 'px' values for text size, making it easier for users to adjust font sizes.

• Readable fonts: Sans-serif fonts were used, with a minimum size of 16px to ensure readability.

• CSS Media Queries: I implemented media queries to support high-contrast themes for users with visual impairments.

# 5. Case Study 2: Responsive Web Design

The second case study involves Airbnb.com, which is known for its excellent responsive design. The site adapts seamlessly to different screen sizes, from mobile phones to large desktop monitors. On smaller screens, the layout shifts to a single-column design, and the navigation collapses into a hamburger menu. The use of flexible images, fluid grids, and media queries ensures a smooth and consistent user experience across all devices.

# 6. Website Overview

My travel website consists of several key pages:  
• Homepage: Introduces users to the website with a featured image slider showcasing top travel destinations.  
• Destinations Page: Offers details on various travel destinations, including images, descriptions, and suggested activities.  
• Booking Page: Allows users to book trips, complete with a user-friendly form where they can select their travel preferences and dates.  
• Contact Page: Provides contact information and an interactive form for users to ask questions or request support.

# 7. CSS Techniques for Responsive Design

I used several CSS techniques to ensure that my website responds well to different screen sizes:

• Media queries: Applied @media rules to adjust layout based on screen size, ensuring mobile and tablet support.

• Flexible grids and fluid layouts: Used a percentage-based grid system to allow the website to scale naturally.

• Responsive images: Ensured images resize correctly with the 'max-width: 100%' rule.

# 8. Testing Across Devices

I tested my travel website on various devices, including smartphones, tablets, and desktop computers, as well as across multiple browsers like Google Chrome, Firefox, and Safari. This testing ensured that the website looked good and functioned properly on all devices. I used browser developer tools and online emulators to simulate different screen sizes and test compatibility.

# 9. Conclusion

In conclusion, this project helped me learn how to create an accessible and responsive travel website using HTML and CSS. I applied modern techniques like media queries, fluid grids, and accessibility features to ensure that the site works well for all users, regardless of their device or ability. The project provided valuable experience in testing and refining web design across different platforms.

# 10. References

• "Web Accessibility Basics," W3C. Available at: https://www.w3.org/WAI/fundamentals/accessibility-intro/

• "Responsive Web Design Basics," Google Developers. Available at: https://developers.google.com/web/fundamentals/design-and-ux/responsive/

• "CSS Techniques for Accessibility," Mozilla Developer Network. Available at: https://developer.mozilla.org/en-US/docs/Learn/Accessibility/CSS