

**A MINI PROJECT REPORT**

*on*

**WEB DESIGN TECHNOLOGIES (24CSE361)**

**TRAVEL BLOG WITH IMAGE SLIDESHOW**

*Submitted by*

**R SHARVESH**

**1NH24CS159**

Under the guidance of

**Ms. DIVYANSHI CHHABRA**

Assistant Professor

*In partial fulfillment for the award of the degree of*

**BACHELOR OF ENGINEERING**

*in*

# COMPUTER SCIENCE AND ENGINEERING

**Academic Year: 2025-26 (ODD SEM)**



# CERTIFICATE

This is to certify that the mini project work titled “**Travel Blog with image slideshow”** is a bonafide work carried out by **R Sharvesh (1NH24CS159)** in partial fulfillment of the degree of **Bachelor of Engineering** in **Computer Science and Engineering** of the New Horizon College of Engineering during the year **2025-2026.**

## Signature of Guide Signature of HOD

**SEMESTER END EXAMINATION**

*Name of the Examiner Signature with date*

1.

2.

**ABSTRACT**

Travel begins long before the plane takes off or the road starts to unwind. It begins with curiosity—the quiet pull toward unfamiliar streets, distant horizons, and stories waiting to be discovered. This blog is a reflection of that curiosity, a visual and written journey through places shaped by culture, nature, and human connection. Each destination featured here is more than a point on a map; it is a collection of moments that linger, long after the journey ends.

The image slideshow above offers a glimpse into these experiences as they unfold. Through shifting light, textures, and movement, the photographs capture what words often struggle to describe: the stillness of early mornings, the rhythm of daily life, and the beauty found in both grand landscapes and small, overlooked details. Every image is a pause—a chance to step into a different pace of life and see the world through a traveling lens.

Beyond the visuals, this blog dives into the stories behind the images. It explores the paths taken, the people met along the way, and the emotions that surface when navigating unfamiliar places. From bustling cities to quiet corners of the world, the journey is shaped as much by chance encounters as by careful planning. There are lessons learned on winding roads, comfort found in shared meals, and inspiration drawn from cultures rich in history and tradition.

**ACKNOWLEDGEMENT**

The satisfaction and euphoria that accompany the successful completion of any task would be impossible without the mention of the people who made it possible, who’s constant guidance and encouragement crowned our efforts with success.

I have great pleasure in expressing gratitude to **Dr. Mohan Manghnani**, Chairman, New Horizon Educational Institutions, for providing necessary infrastructure and creating good environment.

I take this opportunity to express my profound gratitude to **Dr. Manjunatha,** Principal, New Horizon College of Engineering, for the constant support and encouragement.

I would like to thank **Dr. Anandhi R J**, Professor and Dean-Academics, NHCE, for her valuable guidance.

I would also like to thank **Dr. B. Rajalakshmi,** Professor and HOD, Department of Computer Science and Engineering, for the constant support.

I also express my gratitude to **Ms DIVYANSHI CHHABRA, Designation**, Department of Computer Science and Engineering, my mini project reviewer, for constantly monitoring the development of the project and setting up precise deadlines. Her valuable suggestions were the motivating factors in completing the work.

**R SHARVESH**

**1NH24CS159**

## CONTENTS

**ABSTRACT I**

**ACKNOWLEDGEMENT II**

**CONTENTS III**

**LIST OF FIGURES V**

1. **INTRODUCTION 1**

1.1 PROBLEM DEFINITION 1

1.2 OBJECTIVES 2

1.3 METHODOLOGIES TO BE FOLLOWED 3

1. **FUNDAMENTALS OF THE LANGUAGES USED 4**

2.1 HTML 4

2.2 HTML TAGS 6

2.3 CSS 6

2.4 JAVASCRIPTS 7

2.5 XHTML 8

2.6 XML 9

1. **REQUIREMENT SPECIFICATION 11**

3.1 HARDWARE REQUIREMENTS 11

3.2 SOFTWARE REQUIREMENTS 11

1. **DESIGN 12**

4.1 DESIGN GOALS 12

1. **IMPLEMENTATION 13**

5.1 HOME PAGE 13

**6. RESULTS 17**

6.1 MAIN PAGE 17

**7. CONCLUSION 20**

**8. REFERENCES 21**

## LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Description** | **Page No.** |
| **5.1** | **CODE SNIPPET FOR ALBUM COVER** | **16** |
| **5.2** | **CODE SNIPPET FOR SONG INFORMATION SECTION** | **16** |
| **5.3** | **CODE SNIPPET FOR CONROLS SECTION** | **16** |
| **5.4** | **CODE SNIPPET FOR APPLYING CSS STYLING** | **17** |
| **5.5** | **CODE SNIPPET FOR CREATING PLAYER BOX** | **18** |
| **5.6** | **CODE SNIPPET FOR ALBUM COVER STYLING** | **18** |
| **5.7** | **CODE SNIPPET FOR SLIDDING EFFECT (ANIMATION)** | **19** |
| **6.1** | **RESULT FOR MUSIC PLAYLIST INTERFACE** | **26** |

### 

### CHAPTER 1

### INTRODUCTION

Travel is more than moving from one place to another; it is a journey of discovery, emotion, and personal growth. Every destination has a story to tell—through its landscapes, culture, people, and everyday moments that often go unnoticed. This travel blog, accompanied by an image slideshow, is an attempt to capture those stories visually and emotionally, allowing readers to experience the journey beyond words.

The slideshow presents a collection of moments frozen in time: breathtaking views, lively streets, peaceful corners, and small details that define the soul of each place. Each image reflects a unique experience, from spontaneous adventures to quiet reflections, showing how travel shapes our understanding of the world. Together, the visuals and narratives create a window into the journey, inviting viewers to walk the same paths, feel the same excitement, and share the same sense of wonder.

Through this blog, the aim is not only to showcase destinations but also to highlight the feelings associated with travel—the thrill of the unknown, the joy of exploration, and the comfort found in new yet familiar places. It is about stepping out of routine, embracing change, and collecting memories that last a lifetime.

* 1. **PROBLEM DEFINITION**

Many existing music interfaces are either visually cluttered, difficult to navigate, or lack personalization. Users often face challenges such as slow loading times, poor layout structure, non-responsive design on mobile devices, and confusing playback controls.  
This project aims to solve these issues by developing a clean, responsive, and easy-to-use music playlist interface that provides smooth interaction, clear navigation, and visually appealing design, ensuring a more satisfying listening experience.

**Key Problems Identified**

1. Poor visual organization leading to difficulty in browsing songs.
2. Non-responsive layouts that perform badly on mobile screens.
3. Inconsistent or confusing playback controls.
4. Lack of user-friendly playlist management features.
5. Slow or inefficient interface interactions.
6. Outdated or unattractive color schemes and UI elements**.**

**1.2**  **OBJECTIVES**

1. To design a clean and modern interface for managing and playing music.
2. To ensure responsive design across desktops, tablets, and mobile devices.
3. To implement intuitive audio controls such as play, pause, next, previous, and volume.
4. To create a well-organized layout that allows users to easily browse and manage songs.
5. To improve user experience (UX) through smooth interactions and visually pleasing design.

6. To develop the interface using HTML, CSS, and JavaScript with simple and efficient code.

* 1. **METHODOLOGIES TO BE FOLLOWED**

1. Requirement Analysis

* Identify essential features: playlist display, audio player, controls, animations.
* Decide target users and devices.

2. UI/UX Design

* Create wireframes and layouts.
* Choose color palettes, typography, icons, and spacing.
* Use design principles: alignment, contrast, balance, and readability.

3. Front-End Development

* Structure content with HTML.
* Style the interface using CSS (flexbox, grid, animations).
* Add interactivity and playback features using JavaScript.

4. Responsiveness

* Apply media queries for mobile and desktop.
* Test layout adaptability across various screen sizes.

5. Integration & Optimization

* Optimize images and audio for faster loading.
* Improve UI performance and reduce code redundancy.

**CHAPTER 2**

### FUNDAMENTALS OF THE LANGUAGES USED

The **Music Playlist Interface** on Web Design covers the foundational languages of web development: HTML, CSS, and JavaScript. These languages are essential for creating structured, styled, and interactive web pages.

* 1. **HTML (Hyper Text Markup Language)**

HTML is the standard markup language for creating web pages. It structures the content of a webpage using elements enclosed in tags. HTML (Hyper Text Markup Language) was created by Tim Berners-Lee in 1989 as part of the World Wide Web project at CERN, aimed at enabling scientists to share documents through hyperlinks. The first version of HTML was introduced in 1991, consisting of 18 basic tags like <p> and <b> .

In 1995, HTML 2.0 became the first standardized version, introducing new features like forms and tables. Subsequent updates, such as HTML 3.2 in 1997, added support for CSS, while HTML 4.0 in 1999 integrated multimedia elements and scripting capabilities. During the late 1990s and early 2000s, XHTML emerged as a stricter version of HTML but faced adoption challenges. HTML5, introduced in 2014, revolutionized the web by adding semantic elements, multimedia capabilities, and cross-platform support, cementing its role as the cornerstone of modern web development.

**Key Features**:

• **Structure**: Defines elements like headings, paragraphs, lists, and images.

• **Links**: Connects to other documents using hyperlinks.

• **Forms**: Captures user input.

**2.2 HTML TAGS**

* <html>: Defines the root of the HTML document.
* <head>: Contains metadata (like the title, links to stylesheets, and scripts) which is not displayed on the web page itself.
* <title>: Defines the title of the document, which appears in the browser's title bar or tab.
* <body>: Defines the main content of the HTML document that is visible to the user.

**Text and Formatting Tags**

These tags are used to format text content.

* <h1> to <h6>: Define HTML headings, with <h1> being the most important and <h6> the least.
* <p>: Defines a paragraph.
* <br>: Inserts a single line break (self-closing tag).
* <b>: Defines bold text (stylistic only).
* <strong>: Defines important text (semantic, usually displayed in bold).
* <i>: Defines a part of text in an alternate voice or mood (stylistic, usually displayed in italics).
* <em>: Defines emphasized text (semantic, usually displayed in italics).

**Links and Images**

These tags are essential for navigation and multimedia.

* <a>: Defines a hyperlink, used to link to other pages or resources.
* <img>: Embeds an image (self-closing tag).

**Lists Tags**

These tags are used to create lists of items.

* <ul>: Defines an unordered (bulleted) list.
* <ol>: Defines an ordered (numbered or lettered) list.
* <li>: Defines a list item within <ul> or <ol> tags.

**Semantic and Layout Tags**

HTML5 introduced semantic tags that clearly define the structure and meaning of the content.

* <header>: Defines a container for introductory content or a set of navigational links.
* <nav>: Defines navigation links.
* <main>: Specifies the main content of a document.
* <section>: Defines a section in a document (e.g., chapters, headers, footers).
* <article>: Defines independent, self-contained content.
* <footer>: Defines a footer for a document or section.

**Forms and Tables**

These tags are used for user input and data presentation.

* <form>: Defines an HTML form for user input.
* <input>: Creates form inputs (e.g., text fields, checkboxes, radio buttons).
* <table>: Defines a table.
* <tr>: Defines a table row.
* <th>: Defines a table header cell.
* <td>: Represents a standard table data cell.

.

**2.3 CSS**

CSS (Cascading Style Sheets) was introduced in 1996 by the World Wide Web Consortium(W3C) as a solution to separate the content of web pages from their presentation. The concept was first proposed by HAkon Wium Lie in 1994, aiming to provide web developers with a way to control the appearance of web pages without altering their HTML structure. The first specification, CSS1, included basic styling features such as font customization, text alignment, and margins. In 1998, CSS2 expanded its capabilities with features like positioning, media types, and table styling. However, inconsistent browser support slowed adoption during the early 2000s. CSS3, introduced as a modular update, brought transformative features like animations, transitions, and responsive design through media queries. Today, CSS remains a fundamental technology in web development, empowering developers to create visually engaging and responsive websites. It defines the presentation of HTML documents, including layout, colors, fonts, and spacing, ensuring a seamless user experience across devices.

**Key Features**:

• **Selectors:** Targets HTML elements (e.g., p, #id, .class).

• **Box Model**: Defines padding, border, and margin for layout.

**• Flexibility**: Enables responsive design and animations.

**2.4 JAVASCRIPT**

JavaScript is a high-level, lightweight, and versatile programming language primarily used to create interactive and dynamic web applications. It enables developers to add features such as animations, form validations, and real-time content updates. Initially developed by Brendan Eich at Netscape in 1995, it was originally called Mocha, later renamed LiveScript, and eventually JavaScript to align with the popularity of Java at the time. JavaScript gained widespread adoption with the introduction of ECMAScript standards in 1997, which ensured consistency across implementations. Over the years, JavaScript has evolved significantly, adding modern features like asynchronous programming, modularization, and extensive libraries, making it a cornerstone of modern web development alongside HTML and CSS.

**Key Features and examples:**

• **Basics:** Understanding variables (var, let, const), data types, operators, and expressions.

**• Functions**: Writing reusable blocks of code to perform specific tasks.

**• DOM Manipulation**: Selecting and modifying elements dynamically.

**• Event Handling**: Responding to user actions like clicks and keypresses.

**• Form Validation**: Ensuring proper user input.

**2.5 XHTML**

XHTML or Extensible HyperText Markup Language is a mix of HTML and XML, very similar to [HTML](https://www.geeksforgeeks.org/html/html-introduction/) but stricter. It's like a rulebook for creating web pages that browsers easily understand. Unlike HTML, you have to be careful and follow the rules exactly. Most browsers support it. Just think of it as a more precise way to write web code. It was developed by the World Wide Web Consortium (W3C) and helps web developers transition from HTML to XML. With XHTML, developers can enter the XML world with all its features while still ensuring backward and future compatibility of the content. The XHTML family includes three document types; the first is XHTML 1.0, which was recommended by W3C on January 26, 2000. The second is XHTML 1.1, which was recommended by W3C on May 31, 2001.The third is XHTML5, a standard used for developing an XML adaptation of the HTML5 specification. An XHTML document must have an XHTML <!DOCTYPE> declaration.

**The elements of XHTML** :

* **<html>**: The root element that encloses the entire document and must contain an xmlns attribute.
* **<head>**: Contains metadata about the document that does not appear in the main browser window.
  + **<title>**: Specifies the title of the document, displayed in the browser's title bar or tab.
  + **<meta />**: Provides meta-information like character set and keywords.
  + **<link />**: Links to external resources, such as stylesheets (CSS).
  + **<script>**: Links to or contains executable scripts, such as JavaScript.
  + **<body>** (or <frameset> for frameset DTD): Contains the visible content of the web page.

**2.6 XML**

Extensible Markup Language (XML) is a type of markup language that establishes a set of guidelines for encoding texts in a way that is both machine- and human-readable. For storing and transferring data on the web and in many other applications, XML is widely used. XML steps in as a versatile tool for encoding and organizing data in a way that both humans and machines can comprehend.

XML emerged in the late 1990s as a revolutionary concept in the evolving landscape of the internet. Before XML, HTML served as the predominant language for web content, but it lacked the flexibility needed for complex data representation. XML arrived as a solution, offering a standardized format for expressing diverse types of data in a hierarchical structure.

**CHAPTER 3**

**REQUIREMENT SPECIFICATION**

**3.1 HARDWARE REQUIREMENTS**

1. Processor:

• Minimum: Dual-core processor (e.g., Intel Core i3 or equivalent).

• Recommended: Quad-core processor or higher for multitasking.

2. RAM:

• Minimum: 2 GB (sufficient for lightweight operations).

3. Storage:

• A minimum of 100 MB of free space (to store the HTML file, assets, and browser cache).

4. Display:

• Screen resolution of at least 1024x768 to accommodate the quiz UI.

• A modern display supporting standard web colors.

5.Input Devices:

• Keyboard and mouse (or touch input for tablets and mobile devices).

6. Internet Connection:

• Required since some animations in the document are procured from sources in the internet.

**3.2 SOFTWARE REQUIREMENTS**

1. Operating System:

• Windows (7 or higher), macOS (10.12 or higher), or a modern Linux distribution.

• For mobile: Android 5.0+ or iOS 10+.

2. Web Browser: • Modern web browsers like:

• Google Chrome (latest version preferred).

• Mozilla Firefox (latest version preferred).

• Microsoft Edge (latest version).

• Safari (for macOS/iOS).

• Ensure the browser supports ES6 JavaScript and CSS3 features.

3. Text Editor/IDE (for development or modification):

• Notepad, Visual Studio Code, Sublime Text, Atom, or any basic text editor.

4. Web Server (Optional):

• For local hosting (if not opening directly in a browser):

• Use a lightweight server like Python's built-in HTTP server (python -m http.server) or tools like XAMPP.

**CHAPTER 4**

### DESIGN

#### 4.1 DESIGN GOALS

**1. User-Friendly Navigation**

* Provide an intuitive layout that allows users to easily browse, search, and select songs.
* Ensure clear visibility of playlist items, play controls, and navigation menus.

**2. Attractive & Modern UI**

* Use visually appealing colors, typography, and spacing.
* Maintain a balance between aesthetics and simplicity to improve usability.

**3. Responsive Design**

* Interface should automatically adapt to various screen sizes – desktop, tablet, and mobile.
* Maintain consistent functionality and readability across devices.

**4. Smooth Playback Controls**

* Provide seamless play, pause, next, previous, repeat, and shuffle options.
* Ensure smooth transitions between tracks without delays.

**5. Efficient Content Organization**

* Display playlists in a well-structured manner.
* Use categories, filters, or sorting features for easy navigation of songs.

**6. Accessibility**

* Ensure the interface is usable for all users, including those with disabilities.
* Add features like keyboard navigation, readable fonts, sufficient contrast, and alternative text.

**7. Fast Loading & Performance Optimization**

* Reduce load time by optimizing images, code, and audio files.
* Aim for smooth interactions without lag.

**8. Personalization**

* Allow users to create, edit, and manage custom playlists.
* Optionally include user preferences like theme mode (dark/light) or favorite tracks.

**9. Consistent Design Language**

* Maintain uniform colors, icons, buttons, and spacing throughout the interface.
* Follow web design standards for familiarity and ease of use.

**10. Security & Data Protection**

* Ensure secure handling of user-generated playlists and account data (if login is used).
* Protect media access from unauthorized usage.

**CODE**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0"/>

<title>WanderScape | Travel Blog</title>

<!-- Google Font -->

<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;600&display=swap" rel="stylesheet">

<style>

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: 'Poppins', sans-serif;

}

body {

background: #f5f7fa;

color: #333;

line-height: 1.6;

}

header {

position: fixed;

width: 100%;

background: rgba(0,0,0,0.6);

color: #fff;

padding: 15px 40px;

z-index: 1000;

backdrop-filter: blur(8px);

}

header h1 {

font-weight: 600;

letter-spacing: 2px;

}

/\* SLIDESHOW \*/

.slideshow {

position: relative;

height: 100vh;

overflow: hidden;

}

.slide {

position: absolute;

width: 100%;

height: 100%;

opacity: 0;

transform: scale(1.1);

transition: opacity 1.2s ease, transform 1.2s ease;

}

.slide.active {

opacity: 1;

transform: scale(1);

}

.slide img {

width: 100%;

height: 100%;

object-fit: cover;

}

.overlay {

position: absolute;

inset: 0;

background: linear-gradient(to top, rgba(0,0,0,0.7), transparent);

display: flex;

align-items: flex-end;

padding: 60px;

color: #fff;

}

.overlay h2 {

font-size: 48px;

animation: fadeUp 1.2s ease forwards;

}

@keyframes fadeUp {

from { opacity: 0; transform: translateY(30px); }

to { opacity: 1; transform: translateY(0); }

}

/\* NAV BUTTONS \*/

.nav {

position: absolute;

top: 50%;

width: 100%;

display: flex;

justify-content: space-between;

padding: 0 30px;

transform: translateY(-50%);

}

.nav button {

background: rgba(255,255,255,0.3);

border: none;

color: #fff;

font-size: 28px;

padding: 12px 18px;

cursor: pointer;

border-radius: 50%;

transition: background 0.3s;

}

.nav button:hover {

background: rgba(255,255,255,0.7);

color: #000;

}

/\* DOTS \*/

.dots {

position: absolute;

bottom: 20px;

width: 100%;

text-align: center;

}

.dot {

display: inline-block;

width: 12px;

height: 12px;

margin: 5px;

background: #bbb;

border-radius: 50%;

cursor: pointer;

transition: background 0.3s;

}

.dot.active {

background: #fff;

}

/\* CONTENT \*/

section {

max-width: 1000px;

margin: 80px auto;

padding: 0 20px;

animation: fadeUp 1.2s ease;

}

section h3 {

font-size: 32px;

margin-bottom: 20px;

}

footer {

text-align: center;

padding: 30px;

background: #111;

color: #ccc;

}

</style>

</head>

<body>

<header>

<h1>WanderScape</h1>

</header>

<div class="slideshow">

<div class="slide active">

<img src="https://images.unsplash.com/photo-1500530855697-b586d89ba3ee" />

<div class="overlay"><h2>Chasing Horizons</h2></div>

</div>

<div class="slide">

<img src="https://images.unsplash.com/photo-1507525428034-b723cf961d3e" />

<div class="overlay"><h2>Ocean Escapes</h2></div>

</div>

<div class="slide">

<img src="https://images.unsplash.com/photo-1501785888041-af3ef285b470" />

<div class="overlay"><h2>Into the Wild</h2></div>

</div>

<div class="nav">

<button onclick="prevSlide()">❮</button>

<button onclick="nextSlide()">❯</button>

</div>

<div class="dots"></div>

</div>

<section>

<h3>About the Journey</h3>

<p>

Travel is more than destinations—it's moments, people, and stories that reshape how we see the world.

This blog captures those moments through visuals and reflections, inviting you to explore places where

culture, nature, and curiosity meet.

</p>

</section>

<footer>

© 2025 WanderScape | Travel & Stories

</footer>

<script>

const slides = document.querySelectorAll('.slide');

const dotsContainer = document.querySelector('.dots');

let index = 0;

slides.forEach((\_, i) => {

const dot = document.createElement('span');

dot.classList.add('dot');

dot.onclick = () => showSlide(i);

dotsContainer.appendChild(dot);

});

const dots = document.querySelectorAll('.dot');

dots[0].classList.add('active');

function showSlide(i) {

slides[index].classList.remove('active');

dots[index].classList.remove('active');

index = i;

slides[index].classList.add('active');

dots[index].classList.add('active');

}

function nextSlide() {

showSlide((index + 1) % slides.length);

}

function prevSlide() {

showSlide((index - 1 + slides.length) % slides.length);

}

setInterval(nextSlide, 6000);

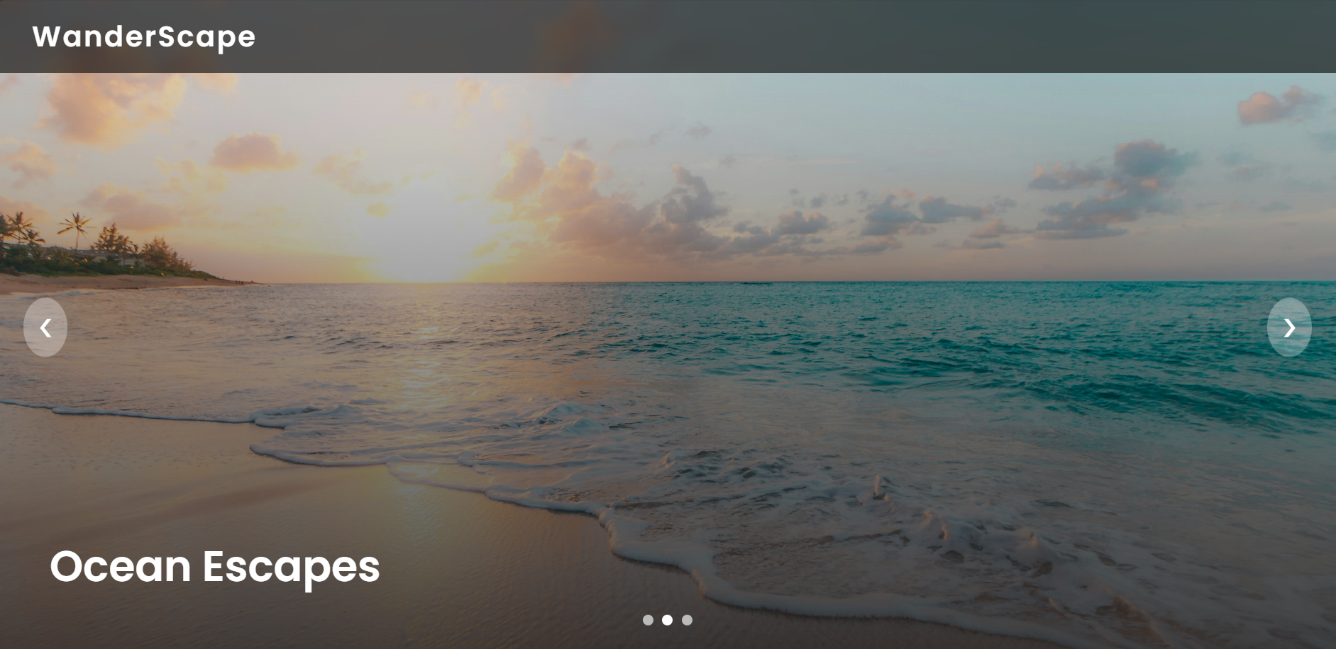
</script></body>

</html>

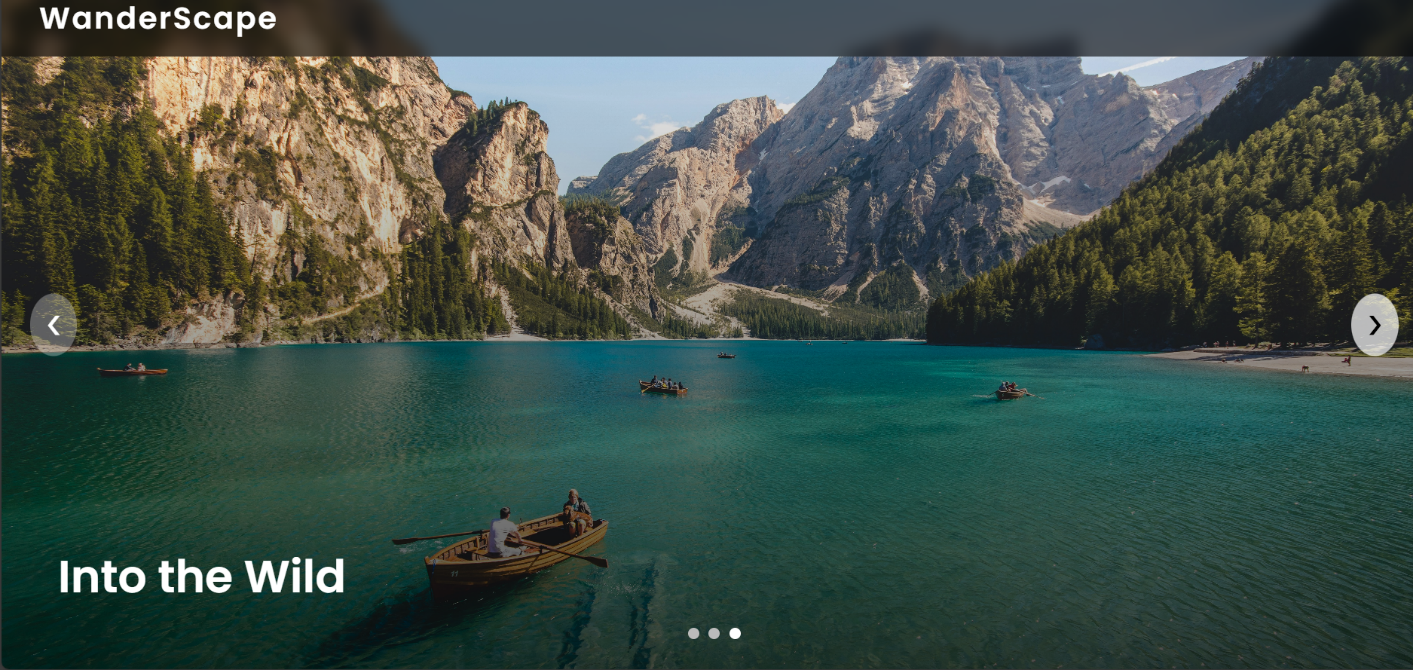
**CHAPTER 5**

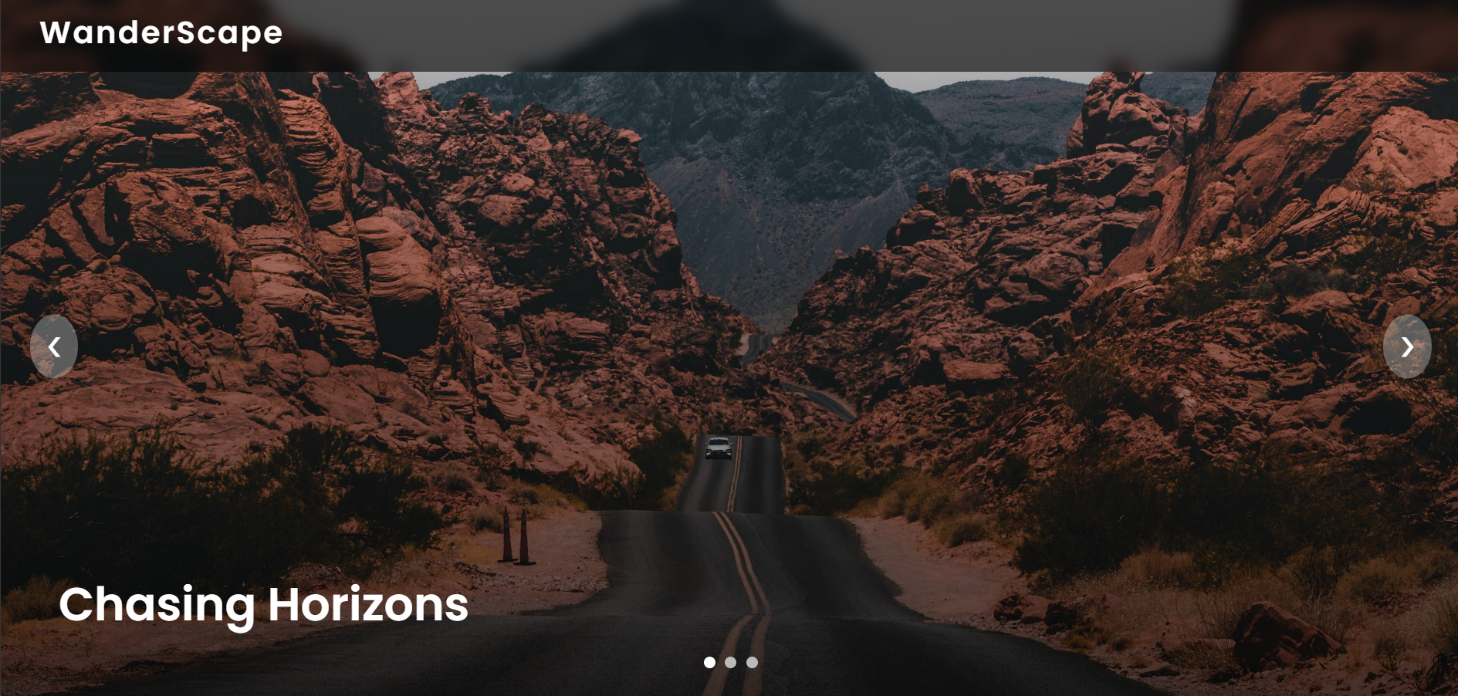
### RESULTS

A modern dark-themed music player UI with glowing album art, playback buttons, and a clean playlist list.



6.1 RESULT OF MUSIC PLAYLIST INTERFACE





**CHAPTER 6**

**CONCLUSION**

As the final images fade through the slideshow, what remains is more than a collection of beautiful places—it is a story of moments, emotions, and quiet transformations that travel brings into our lives. Each photograph captures a fragment of the journey: the golden light of early mornings, the vibrant chaos of local streets, the calm of hidden landscapes, and the warmth of people met along the way. Together, they remind us that travel is not only about destinations, but about how those destinations make us feel.

This journey reinforced the idea that the world is both vast and deeply connected. From unfamiliar paths to shared smiles, every experience added a new perspective and challenged the comfort of routine. The slideshow serves as a visual diary, preserving memories that words alone cannot fully express. A single image can recall the scent of the air, the rhythm of footsteps on unknown roads, or the sense of wonder felt when seeing something for the first time.

Travel also teaches patience, adaptability, and gratitude. Missed turns often lead to unexpected discoveries, and slow moments become the most meaningful ones. Through this journey, the importance of being present became clear—pausing to observe, listen, and truly experience a place rather than rushing through it. These lessons linger long after the bags are unpacked.

As this blog comes to an end, let it be an invitation rather than a farewell. An invitation to explore beyond familiar boundaries, to document your own stories, and to see the world with curious eyes. Whether you travel far or stay close to home, there is always something new waiting to be discovered. May these images inspire you to chase experiences, embrace the unknown, and create memories worth remembering—because every journey, no matter how big or small, has the power to leave a lasting mark.

**REFERENCES**

* + https://devdevout.com/css/css-animated-backgrounds
  + <https://www.w3schools.com/tags/tag_html.asp>
  + https://github.com/topics/web-development-project
  + <https://www.w3schools.com/js/js_object_property.asp>
  + https://developer.mozilla.org/en-US/docs/Learn/CSS