**SUMMER TRAINIG REPORT ON**

**NEWS WEBSITE**

Submitted to Guru Gobind Singh Indraprastha University, Delhi (India) In partial fulfilment of the requirement for the award of the degree of

**Bachelor of Technology In**

**Information Technology**

### Under the guidance of

**Ms.Priyanka Kalkandha**

Assistant Professor

## Information Technology Department

### Submitted by:

**Mannu Kumar**

## Branch - IT - 1 Enrollment No. – 00415003121



**MAHARAJA SURAJMAL INSTITUTE OF TECHNOLOGY**

**C4, JANAKPURI, NEW DELHI – 110058**

# ACKNOWLEDGEMENT

In the realm of every Summer Training endeavor, big or small, success finds its roots in the collective efforts of extraordinary individuals. It is with immense gratitude that I extend my heartfelt appreciation to those whose unwavering support and guidance made my project a triumph.

I, Mannu Kumar, a student of Maharaja Surajmal Institute of Technology, New Delhi (Information Technology), want to express my deepest thanks to the Delhi Institute of Computer Technology (DICT). They not only entrusted me with the opportunity to undergo training but also showed confidence in my abilities. Special appreciation goes to Mr. Rahul Kumar, whose mentorship played a pivotal role in the success of my 'Web Development' training.

In this moment of reflection, I am honored to acknowledge the timely assistance and valuable insights provided by Mr. Rahul Kumar. His dedication ensured that resources were made available when needed, leading to the seamless completion of my training. My gratitude extends to Dr. Prabhjot Kaur, HOD of the Department of Information Technology at Maharaja Surajmal Institute of Technology, for orchestrating the summer training schedule.

A heartfelt thanks is also due to my mentor, Mr. Rahul Kumar (Instructor, DICT), whose guidance was instrumental in navigating the complexities of the project. I extend my appreciation to the entire Information Technology faculty at Maharaja Surajmal Institute of Technology, New Delhi, for their critical advice, without which this achievement would not have been possible.

Lastly, but certainly not least, I want to express profound gratitude to my family members and friends. Their unwavering support and inspiration were my constant companions throughout the preparation of this training report.

Mannu Kumar

B. Tech, 3rd Year IT Enrolment No. 00415003121

# CERTIFICATE

A certificate of a course

Description automatically generated with medium confidence

**DECLARATION**

I, Mannu Kumar, Roll No 00415003121, B. Tech (5th Semester) of the Maharaja Surajmal Institute of Technology, New Delhi hereby declare that the Training Report entitled "Summer Training on News Paper website" is an original work and data provided in the study is authentic to the best of my knowledge. This training report has not been submitted to any other institution for the fulfillment of any other degree.

Mannu Kumar

Enrolment No.: 00415003121

**Place: New Delhi**

**Date: 15th November 2023**

# ABSTRACT OF PROJECT

In an era where information knows no boundaries, our project aims to create a dynamic and user-friendly Newspaper website that serves as a comprehensive platform for global news coverage. This project aims to provide users with real-time, reliable, and personalized news updates from diverse regions worldwide. It is designed to provide users with a seamless experience of accessing news from diverse regions and cultures, all in one centralized location

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **CONTENT** | **PAGE NO** |
| Acknowledgement | I |
| Certificate | II |
| Declaration | III |
| Abstract of Project | IV |
| Table of Contents | V-VI |
| **CHAPTER** | **PAGE NO** |
| **1. INTRODUCTION** | 1 |
| 1.1 Introduction | 1 |
| 1.2 About Organization | 1 |
| **2. PROJECT OVERVIEW** | 2 |
| 2.1 Project Description | 2 |
| 2.2 Target Audience | 2 |
| **3. TECHNOLOGIES USED** | 3 - 9 |
| 3.1 HTML | 3 - 4 |
| 3.1.1 Introduction to HTML | 3 |
| 3.1.2 Example HTML Code and Explanation | 3-4 |
| 3.1.3 History of HTML | 5 |
| 3.2 CSS | 6 - 7 |
| 3.2.1 Introduction to CSS | 5 - 6 |
| 3.2.2 Example CSS Code and Explanation | 6 |
| 3.2.3 History of CSS | 7 |
| 3.2.4 Versions of CSS  3.3 JAVASCRIPT | 9 |
| 3.4.1 Introduction to Javascript | 10 |
| 3.4.2 History of Javascript | 11 |
| **4. PROJECT CODE** | 12-18 |
| 4.1 INDEX.HTML | 12 |
| 4.2 STYLE.CSS | 14 |

* 1. INDEX.JS 18

1. PROJECT SCREENSHOTS 20
2. CONCLUSION 22
   1. Conclusion 22
   2. Future Scope 22

REFERENCE 23

# INTRODUCTION

## Introduction

Welcome to the summer training report that documents my journey in learning and developing a News Paper website. This report serves as a reflection of the knowledge and experience I gained during my training at DICT, a reputable company that has been instrumental in shaping my understanding of web development and technology.

Throughout this report, I will provide an in-depth account of the Newspaper project, its development, and the technologies I utilized. This introduction aims to set the stage for the subsequent chapters, offering insight into the motives behind this undertaking and its significance in the context of my summer training.

## About Organization

DICT today is one of the most reputed institution in the country accredited under NIELIT Scheme and conducting courses in Computer Science. During my internship at DICT, I had the privilege of working under the guidance of experienced mentors, including Mr. Rahul Kumar. Their valuable insights, constant support, and expertise in this field played a important role in shaping my understanding of web development.

* 1. **Project Description**

# PROJECT OVERVIEW

My project is a comprehensive newspaper website designed to deliver global news coverage in a user-friendly manner. This project focuses on providing real-time and reliable news updates from diverse regions worldwide. The platform features a modern interface, multilingual support, and interactive elements to enhance user engagement. By prioritizing accessibility and personalization, This project aims to foster a connected global community where users can stay informed about the latest events shaping our world.

## Target Audience

The target audience for News Paper website includes global citizens, professionals, students, and researchers seeking up-to-date and reliable information on international affairs. The platform also caters to tech-savvy users, community engagers, and those who prioritize accessible content. In essence, This News Paper website aims to serve a diverse audience interested in staying informed about global events with a user-friendly and inclusive approach

# TECHNOLOGY USED

* 1. **HTML**

## Introduction to HTML

HTML, which stands for Hypertext Markup Language, forms the backbone of web page content and structure. It uses a system of tags and attributes to markup plain text to annotate it with structural semantics so that browsers can understand how to display it as meaningful content.

HTML documents are made of HTML elements that consist of tags enclosed in angle brackets defining the structure and content. Tags such as <h1> to <h6> denote different levels of headings while tags like

<p>, <div>, <span> represent paragraphs, divisions and spans of content.

HTML also provides tags for adding media elements like images, videos, audio as well as embedding external content like YouTube videos and Google Maps. Links between web pages can be created using anchor <a> tags.

Beyond basic text formatting, HTML allows for creating tables, lists, captions, styling text with strong and em tags, adding colors, inserting horizontal rules, and commenting code. With forms and input tags, HTML enables creating forms to capture user input.

## Example HTML Code and Explanation

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>My First HTML Page</title>

</head>

<body>

<h1>Hello, World!</h1>

<p>This is a simple HTML page.</p>

<a href="https://[www.example.com](http://www.example.com/)" target="\_blank">

<img src="https://via.placeholder.com/150" alt="Placeholder Image">

</a>

</body>

</html>

The foundational structure of an HTML document commences with the initiation of an opening <html> tag. Nested within it, the <head> tags encapsulate essential meta-information. Subsequently, the main content of the HTML page resides within the <body> tag. The closure of the HTML document is denoted by the concluding </html> tag. As an introductory guide to the HTML framework, key points elucidate the roles of fundamental HTML tags.

**<!DOCTYPE html>:** Serving as a document type declaration, this directive explicitly denotes that the document adheres to the HTML5 standard. It provides instructions to the browser on how to interpret the succeeding code.

**<html>:** Positioned as the root element, this tag encapsulates the entire HTML document. All other elements are nested within it, marking the commencement and conclusion of the HTML structure.

<head>:\*\* Hosting metadata, the <head> section includes crucial information about the document, such as its title, character encoding, and references to external resources like stylesheets or scripts.

**<meta charset="UTF-8">:** Declares the character encoding for the document as UTF-8.

<meta name="viewport" content="width=device-width, initial-scale=1.0">: Sets the viewport properties for responsive design.

**<title>:** Nested within the <head> section, the title element defines the title of the web page. The text enclosed in this tag appears in the browser's title bar or tab, providing a concise description of the open page.

**<body>:** Serving as the container for the primary content visible to users, the body element accommodates text, images, links, and other elements constituting the actual webpage content.

**<h1>:** Functioning as a heading element, the <h1> tag plays a crucial role in structuring a document and offering visual cues to readers. Typically reserved for the main title, it represents the highest-level heading.

**<p>:** Representing a paragraph of text, the <p> tag is employed to structure and format textual content. Paragraphs are demarcated by line breaks or spacing to enhance content readability.

**<a href="https://**[**www.example.com**](http://www.example.com/)**" target="\_blank">:** An anchor (<a>) element representing a hyperlink. The href attribute specifies the URL, and target="\_blank" opens the link in a new tab.

**<img src="https://via.placeholder.com/150" alt="Placeholder Image">:** An image (<img>) element. The src attribute specifies the source URL of the image, and the alt attribute provides alternative text for accessibility.

This serves as a fundamental illustration, outlining the structural components of an HTML document.While HTML complexity may increase with the addition of more elements, formatting, and interactivity, the foundational structure remains consistent, characterized by elements enclosed within tags.

## History of HTML

HTML was created by **Sir Tim Berners-Lee in late 1991** but was not officially released. It was published in 1995 as HTML 2.0. HTML 4.01 was published in late 1999 and was a major version of HTML.

HTML is a very evolving markup language and has evolved with various versions updating. Long before its revised standards and specifications are carried in, each version has allowed its user to create web pages in a much easier and prettier way and make sites very efficient.

* + - 1. **Birth of the Web (1989-1990):** The World Wide Web concept proposed by Sir Tim Berners-Lee in 1989 led to the creation of the first web page in 1990, displaying basic text.
      2. **HTML 1.0 (1991):** Tim Berners-Lee introduced the first formal HTML specification in 1991, laying the foundation with fundamental elements like headings, paragraphs, lists, and hypertext links.
      3. **HTML 2.0 (1995):** The Internet Engineering Task Force (IETF) published HTML 2.0 as an informational RFC in 1995, introducing tables for more complex layouts.
      4. **HTML 3.0 (1995):** HTML 3.0 marked a significant step forward, incorporating features like tables, applets, text flow around images, and text alignment, although browser support varied.
      5. HTML 4.0 (1997):\*\* Published in 1997, HTML 4.0 introduced style sheets, scripting, and the concept of frames, representing a substantial advancement in web design capabilities.
      6. **XHTML (2000):** XHTML, introduced in 2000, aimed to reformulate HTML 4.0 using XML syntax, emphasizing stricter syntax rules.
      7. **HTML5 (2008):** The latest major revision, HTML5, introduced in 2008, brought groundbreaking features such as native support for audio and video elements, the canvas element for drawing graphics, and improved semantic elements. It remains the current standard, reflecting ongoing collaborative efforts to shape the digital landscape.
  1. **CSS**

## Introduction to CSS

Cascading Style Sheets, commonly known as CSS, is a fundamental technology used in web development to enhance the visual presentation of HTML or XML documents. Introduced in the late 1990s, CSS enables developers to control the layout, formatting, and appearance of web pages, ensuring a consistent and visually appealing user experience.

## Key Components of CSS:

* + - 1. **Selectors:** Selectors define which HTML elements the style rules will apply to. They can target specific elements, classes, IDs, or even attributes.
      2. **Properties:** CSS properties determine the visual aspects of the selected elements, such as color, font, size, spacing, and positioning.
      3. **Values:** Values are assigned to properties and specify the desired characteristics. For example, a color property might have a value like "blue" or "#008080" (hexadecimal code).

#### Basic Syntax:

selector { property: value;

}

#### How CSS Works:

1. Inheritance: Styles are inherited from parent elements to their children, creating a hierarchical structure. This helps maintain consistency across a website.
2. Cascading: If multiple styles conflict, the cascade determines which style takes precedence. This involves factors like specificity, importance, and order of appearance.

#### Benefits of CSS:

1. Separation of Concerns: CSS separates the structure (HTML) from the presentation (CSS), promoting cleaner code and easier maintenance.
2. Consistency: CSS ensures a consistent look and feel throughout a website, making it easier to manage and update.
3. Responsive Design: CSS facilitates responsive web design, allowing layouts to adapt to different screen sizes and devices.
4. Efficiency: By styling multiple elements simultaneously, CSS reduces redundancy and improves efficiency in web development.

In summary, CSS is a powerful tool that enhances the aesthetics and layout of web pages, contributing to a more engaging and user-friendly online experience.

## Example CSS Code and Explanation

body {

background-color: #f0f0f0; font-family: Arial, sans-serif; margin: 0;

padding: 0;

}

h1 {

color: #333; font-size: 36px;

text-align: center;

}

div.content p { line-height: 1.5;

}

a {

text-decoration: none;

}

a:hover {

color: darkorange;

}

* + - 1. **body:** This selector targets the <body> element of the webpage, setting various styles for the entire page.
      2. **{}:**The curly braces enclose a declaration block that contains the styles to be applied to the selected element.
      3. **background-color: #f0f0f0;:**This property changes the background color of the entire page to a light gray (#f0f0f0).
      4. **font-family: Arial, sans-serif;:**This property sets the default font family to Arial, with a fallback to sans-serif fonts if Arial isn't available.
      5. **margin: 0; and padding: 0;:**These properties remove the default margin and padding from the

<body> element.

* + - 1. **h1:**This selector targets all <h1> elements on the page.
      2. **color: #333;:**This property changes the text color of <h1> elements to a dark gray (#333).
      3. **font-size: 36px;:**This sets the font size of <h1> elements to 36 pixels.
      4. **text-align: center;:**This centers the text within <h1> elements.
      5. **div.content p:**This selector targets all <p> elements inside a <div> with the class "content".
      6. **line-height: 1.5;:**This adjusts the line height of the <p> elements for better readability.
      7. **a:**This selector targets all links on the page.
      8. **text-decoration: none;:**This removes the underline from links to make them appear as plain text.

14 **a:hover:**This selector changes the link color when hovered over by the mouse.

## History of CSS

Along with HTML, CSS is also standardized by W3C that controls the web standards. CSS is free to use, independent and it is also an open-source standard. And users can use HTML and CSS together with the W3C.

1. **Genesis (1996-1997):** CSS was first proposed by Håkon Wium Lie and Bert Bos in 1994. It aimed to separate document structure (HTML) from presentation styles. CSS1 was officially introduced in 1996 and provided basic styling capabilities.
2. **CSS2 (1998-2011):** CSS2, introduced in 1998, brought significant enhancements, including positioning, z-index, media types, and support for internationalization. However, its adoption was gradual due to inconsistent browser support.
3. **W3C Working Groups (1997-2008):** CSS underwent continuous development through W3C (World Wide Web Consortium) working groups, addressing issues, introducing modules, and refining specifications.
4. **CSS2.1 (2011):** CSS2.1, finalized in 2011, aimed to consolidate and clarify CSS2, serving as a stable reference for implementations. Browser vendors focused on improving CSS2.1 compatibility.
5. **CSS3 (2001-present):** CSS3 was introduced as a modularized approach to CSS, allowing independent development of features. Selectors, colors, fonts, and animations are among the many modules, enabling richer designs.
6. **Flexbox and Grid Layout (2012-2016):** CSS saw substantial advancements with the introduction of Flexbox (2012) and Grid Layout (2016), providing powerful layout capabilities and easing the complexities of responsive design.
7. **CSS4 (2021):** While not a formal version, the term CSS4 is informally used to describe ongoing developments in CSS. Features like Container Queries, Custom Property Setters, and more aim to enhance CSS capabilities further.
8. **Responsive Design (2010s):** With the rise of mobile devices, responsive design principles became crucial. Media queries in CSS3 allowed developers to create layouts suitable for various screen sizes.
9. **Preprocessors (2000s-present):** Tools like Sass and Less gained popularity, offering features like variables, nesting, and functions, enhancing the maintainability and extensibility of stylesheets.
10. **Current Trends (2020s):** CSS continues to evolve with emphasis on performance, accessibility, and new layout models. The focus is on addressing challenges in modern web development.

CSS's evolution reflects the dynamic nature of web technologies, adapting to the changing needs of designers and developers across different eras.

## Versions of CSS

CSS (Cascading Style Sheets) has evolved through different versions, each introducing new features and improvements. Here are the major versions of CSS:

* + - 1. **CSS1 (1996):** The first version, CSS1, was introduced to separate the content from the presentation. It included basic styling properties for fonts, colors, and text alignment.
      2. **CSS2 (1998):** CSS2 expanded on CSS1, introducing new features like positioning, z-index, and support for media types. However, its adoption was slow due to inconsistent browser support.
      3. **CSS2.1 (2011):** CSS2.1 aimed to clarify and consolidate the CSS2 specification. It provided a stable reference for browser implementations and improved compatibility.
      4. **CSS3 (2001-present):** CSS3 represents a modularized approach, where different modules are developed independently. It introduces a wide range of features, including selectors, colors, backgrounds, borders, text effects, and animations.

Notable CSS3 modules include:

Selectors Level 3 (2001)

Color Module Level 3 (2011)

Backgrounds and Borders Module Level 3 (2012) Text Module Level 3 (2019)

Animations (2011) and Transitions (2009)

* + - 1. **CSS4 (Informal Term):** CSS4 is not an official version but is informally used to refer to ongoing developments beyond CSS3. It includes potential features like Container Queries, Custom Property Setters, and other enhancements.

CSS is now often discussed as an evolving set of modules rather than distinct versions. The term "CSS4" is used colloquially to express the continued evolution of CSS beyond the CSS3 specification. Each module may advance independently, allowing for more frequent updates and additions to the language.

* 1. **JAVASCRIPT**

## Introduction to Javascript

JavaScript is a high-level, versatile programming language primarily utilized for web development. Its primary role is to add interactivity and dynamic behavior to web pages, enhancing the user experience. JavaScript is known for its ability to run directly in web browsers, allowing developers to create responsive and interactive websites. Beyond the browser, it has gained popularity on the server-side with technologies like Node.js, enabling the development of scalable and efficient server applications.

One of the distinctive features of JavaScript is its dynamic typing, where variables are not explicitly typed. This flexibility contributes to its ease of use. JavaScript follows a prototype-based object- oriented programming paradigm, using prototypes rather than classes for creating objects. The language is also event-driven, allowing developers to respond to user actions or system events.

In terms of syntax, JavaScript shares similarities with other C-based languages like Java and C++. It utilizes functions as first-class citizens, meaning they can be passed as arguments and returned from other functions. Variable declaration is done using var, let, or const.

A basic example illustrates the syntax and structure of JavaScript, including variable declaration, function definition, and function invocation. JavaScript's capability extends to manipulating the Document Object Model (DOM) in web browsers, enabling developers to dynamically update content and styles on web pages.

Asynchronous programming is a crucial aspect of JavaScript, and it supports mechanisms such as callbacks, Promises, and the async/await syntax. This is particularly valuable for tasks like fetching data from a server without blocking the execution of other code.

## History of Javascript

JavaScript, often abbreviated as JS, has a fascinating history that dates back to the early days of the internet. Here's a concise overview:

* + - 1. **Birth at Netscape (1995):** JavaScript was created by Brendan Eich in 1995 while he was working at Netscape Communications Corporation. Originally named Mocha, it was later briefly known as LiveScript before finally being named JavaScript.

The goal was to create a scripting language for the Netscape Navigator browser that could enable dynamic content and interactions on web pages.

* + - 1. **Collaboration with ECMA (1997):** To standardize JavaScript and ensure its compatibility across different browsers, Netscape collaborated with ECMA International (European Computer Manufacturers Association).

The standardized version of JavaScript was named ECMAScript. The first edition of ECMAScript was published in 1997.

* + - 1. **Microsoft's JScript (1996-1997):** Microsoft introduced its own version of JavaScript called JScript for Internet Explorer. While JScript was similar, it had some differences from Netscape's implementation.

This led to competition and some compatibility issues between browsers.

Browser Wars and DOM (1996-2001):JavaScript became a key player in the "Browser Wars" between Netscape Navigator and Microsoft Internet Explorer. Each browser developed its own Document Object Model (DOM), causing disparities in how JavaScript interacted with HTML.

Despite these challenges, developers started using JavaScript to create more dynamic and interactive websites.

* + - 1. **The Rise of AJAX (Early 2000s):** As web applications became more sophisticated, JavaScript's role evolved. AJAX (Asynchronous JavaScript and XML) emerged, allowing web pages to update content asynchronously without requiring a full page reload.

This period marked the beginning of JavaScript's transition from a simple scripting language to a powerful tool for building web applications.

* + - 1. **JavaScript on the Server (Node.js - 2009):** In 2009, Ryan Dahl introduced Node.js, a server-side JavaScript runtime. This allowed developers to use JavaScript for server-side scripting, opening up new possibilities for building scalable and real-time applications.
      2. **Modern JavaScript (ES6 and Beyond):**ECMAScript continued to evolve, introducing new features and syntax enhancements. ECMAScript 6 (ES6), published in 2015, was a significant update that brought arrow functions, classes, and other modern language features.

Subsequent versions of ECMAScript have continued to enhance the language with features like async/await and new data structures.

* + - 1. **JavaScript Frameworks and Libraries:**The popularity of JavaScript soared with the rise of frameworks and libraries like Angular, React, and Vue.js. These tools simplified the process of building complex web applications.
      2. **JavaScript Everywhere (Present):** Today, JavaScript is a ubiquitous programming language. It runs on both the client and server sides, making it a crucial technology for full-stack development. The JavaScript ecosystem is vast, encompassing a diverse range of tools, libraries, and frameworks. JavaScript's journey from a browser scripting language to a versatile, full-stack language has been marked by continuous innovation and community collaboration. It remains an integral part of web development, driving the creation of modern, interactive, and dynamic online experiences.

# PROJECT CODE

**NEWS.HTML**

****

**A screenshot of a computer code

Description automatically generated**

**A screen shot of a computer

Description automatically generated**

**NEWS.CSS**

**A screenshot of a computer

Description automatically generated**

A screen shot of a computer program

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screen shot of a computer code

Description automatically generated

**NEWS.JS**

A screenshot of a computer code

Description automatically generated

A screenshot of a computer code

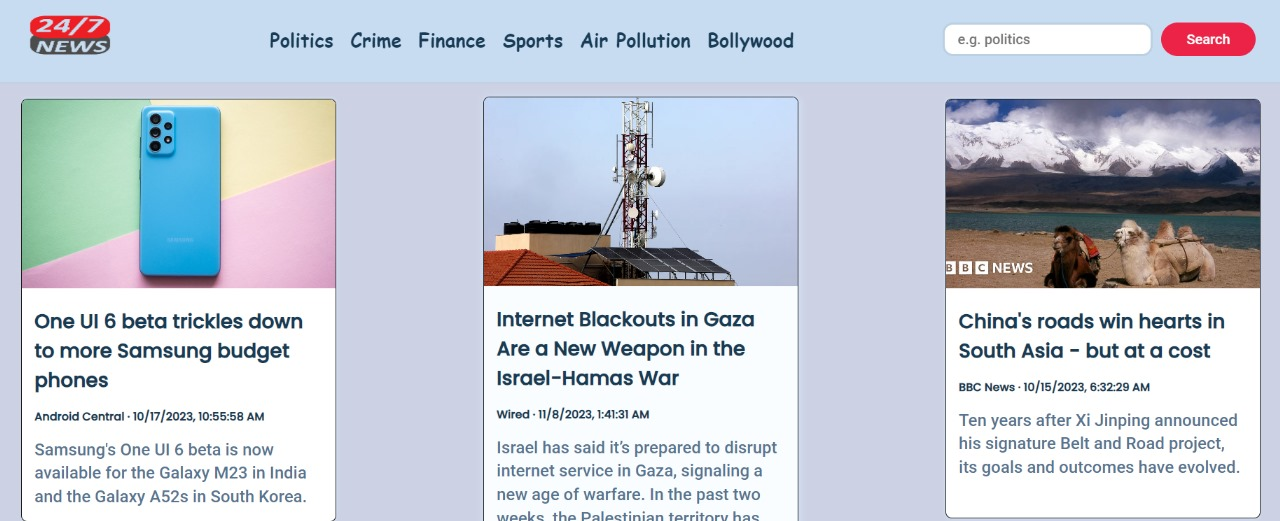
Description automatically generated

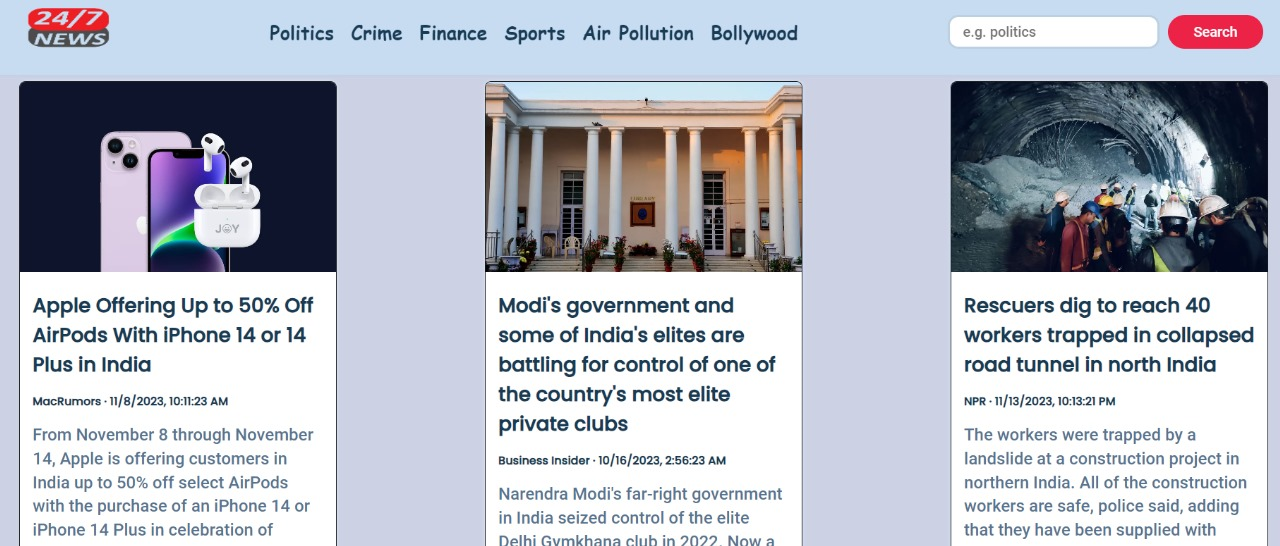
A computer code with text

Description automatically generated

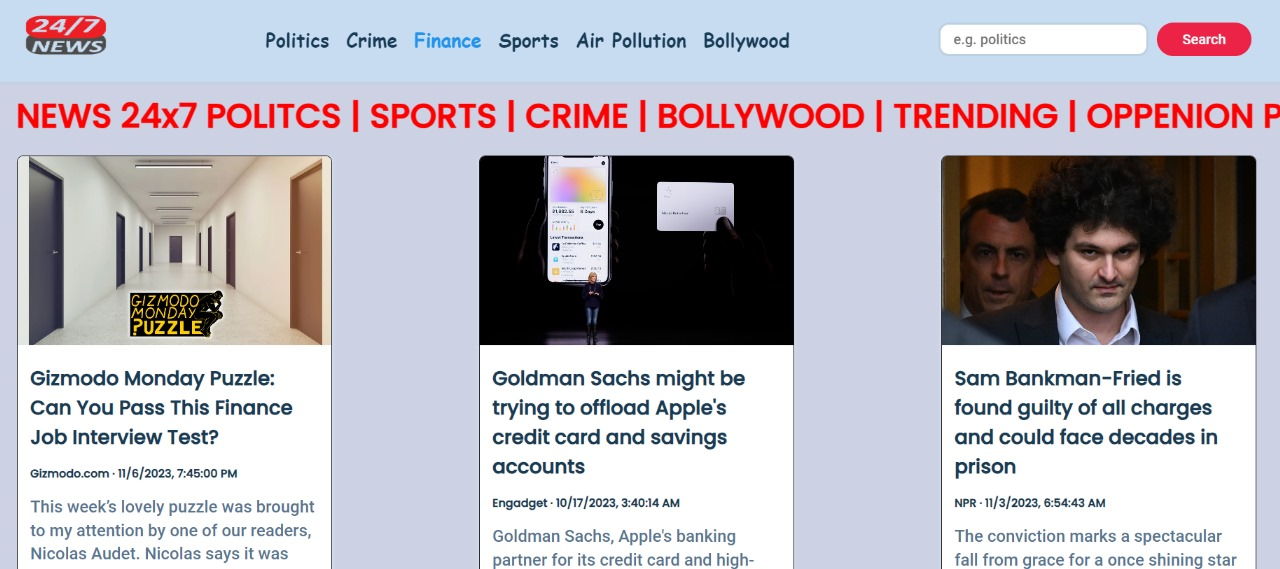
# PROJECT SCREENSHOT

**HOME PAGE**





**FINANCE**



**POLITICS**

A screenshot of two men

Description automatically generated

# CONCLUSION

In conclusion, the development of the News API project has been a rewarding and enlightening experience. This project, implemented using HTML, CSS, and JavaScript, not only deepened my technical skills but also provided valuable insights into the world of web development, user interfaces, and data integration.

**Technical Skills Enhancement**

The project allowed me to leverage HTML for structuring the content, CSS for styling and layout, and JavaScript for dynamic and interactive features. Through this, I honed my abilities in creating responsive and visually appealing web pages that deliver a seamless user experience.

**Data Integration with News API**

The integration of the News API played a pivotal role in providing up-to-date and relevant news content to the users. Leveraging JavaScript, I dynamically fetched and displayed news articles based on user preferences, creating a dynamic and engaging browsing experience.

**Future Development and Learning Opportunities**

While the News API project has reached a functional state, there are ample opportunities for future development. Potential enhancements could include refining the user interface, implementing user authentication for personalized news recommendations, or incorporating additional features to enrich the overall user experience.

In summary, the News API project has been a valuable journey in integrating technical skills, data sources, and user interface design. It stands as a testament to the capabilities of HTML, CSS, and JavaScript in creating functional and interactive web applications. As technology advances, this project serves as a stepping stone towards more sophisticated endeavors.

# REFERENCE

* + **HTML**: <https://www.w3schools.com/html/html_intro.asp>
  + **CSS**: <https://developer.mozilla.org/en-US/docs/Web/CSS>
  + **JAVASCRIPT**: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
  + **Historical Reference**: https://[www.wikipedia.org/](http://www.wikipedia.org/)