

INTERNSHIP REPORT

A report submitted in partial fulfillment of the requirements for the Award of Degree of

**BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE AND DATA SCIENCE**

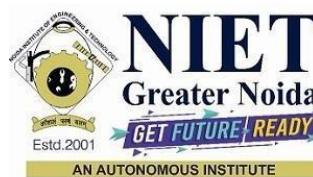
on

**“Web Development”
from
“Sync Intern's”**

**By
Mohsin Khan
Roll. No.: 2201331549010**

**Under Supervision of
Mr. Ravi Raj**

(Duration: 15th July' 2023 to 15th August' 2023)



**DEPARTMENT OF COMPUTER SCIENCE AND BUSINESS SYSTEMS
NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY,
GREATER NOIDA
(An Autonomous Institution)**

DEPARTMENT OF COMPUTER SCIENCE AND BUISNESS SYSTEMS
INSTITUTE OF ENGINEERING AND TECHNOLOGY
(An Autonomous Institution)
GREATER NOIDA



CERTIFICATE

This is to certify that the “**Internship report**” submitted by **Mohsin khan(Roll No:2201331549010)** is work done by him and submitted during 2023 – 2024 academic year, in partial fulfillment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND DATA SCIENCE**, at **Sync Intern's**.

Department Internship Coordinator
Mr. Ravi Raj
Assistant professor, CSDS

Dr. Priyanka Chandani
Head of the Department of
CSDS/AI/CSBS



Certificate

OF COMPLETION

PROUDLY PRESENTED TO

Mohsin Khansh

We are happy to certify that Mr./Ms. **Mohsin Khan** has completed one month virtual internship in **Web Development** from 15 July to 15 August, 2023.

We appreciate his/her work and Contribution.

16/08/2023

DATE



A handwritten signature in black ink that reads "Haseen".

PROGRAM
MANAGER

CERTIFICATE ID : SI/JULI2K230482

ACKNOWLEDGEMENT

First, I would like to thank **Sync Intern's** for giving me the opportunity to do an internship within the organization.

I also would like all the people that worked along with me **Sync Intern's** with their patience and openness they created an enjoyable working environment.

It is indeed with a great sense of pleasure and immense sense of gratitude that I acknowledge the help of these individuals.

I am highly indebted to Director **Dr. Vinod M. Kapse**, for the facilities provided to accomplish this internship.

I would like to thank my Head of the Department **Dr. Priyanka Chandani** for his constructive criticism throughout my internship.

I would like to thank **Mr. Ravi Raj**, internship coordinator Department of CSDS for their support and advice to get and complete internship in above said organization.

I am extremely great full to my department staff members and friends who helped me in the successful completion of this internship.

**Mohsin Khan
(2201331549010)**

ABSTRACT

Web development is the process of creating websites and web applications. It encompasses both front-end and back-end development. Front-end development focuses on the user interface and experience, utilizing technologies like HTML, CSS, and JavaScript. Back-end development deals with the server-side aspects, including databases and server scripting languages like PHP or Python. Full-stack developers are proficient in both front-end and back-end development. Responsive web design ensures that websites function well on various devices, and web development frameworks expedite the process. Security, performance optimization, and adherence to web standards are essential. Web developers use various tools and hosting solutions to bring their projects to life in this dynamic and ever-evolving field.

Organization Information:

CodSoft is an IT service and IT consultancy that specializes in creating innovative solutions for businesses. We are passionate about technology and believe in the power of software to transform the world. Our internship program is just one of the ways in which we are investing in the future of the industry. At CodSoft, we believe practical knowledge is the key to success in the tech industry. Our aim is to help students lacking basic skills by offering hands-on learning through live projects and real-world examples. Our team consists of industry experts who are dedicated to equipping interns with the knowledge and skills needed to succeed in their careers. We foster an inclusive and supportive environment that encourages learning, growth, and creativity. Its mission is to emphasize Software development is an iterative process, and the mission includes continually improving software solutions. This entails gathering feedback, identifying areas for enhancement, and implementing updates to make the software more efficient, user-friendly, and reliable. Continuous improvement also involves staying up to date with the latest technologies, tools, and industry trends.

Programs and opportunities:

Learning web development is a valuable and rewarding endeavor in today's digital age. It equips individuals with the skills to create and maintain websites and web applications, which have become indispensable in various professional fields and personal pursuits. The outcome of learning web development is multifaceted, providing numerous benefits. Web development skills are in high demand in the job market. With businesses and organizations of all sizes seeking a strong online presence, web developers are essential for creating user-friendly websites and interactive applications. This demand often translates into lucrative career opportunities and job security. Learning web development fosters creativity and problem-solving. It empowers individuals to bring their ideas to life on the internet, whether through personal blogs, e-commerce sites, or innovative web applications. This creative outlet can be immensely satisfying and opens doors to entrepreneurship and personal branding. Web development is a continuously evolving field, offering opportunities for ongoing learning and growth. As technologies and frameworks advance, web developers can stay engaged and adapt to new trends and best practices. This constant learning ensures that your skills remain relevant and up to date. Web development skills can be highly transferable. The ability to code, work with databases, and understand web architecture can extend to other software development areas. This flexibility allows individuals to explore different domains within the tech industry, such as mobile app development, game development, or even data science.

Methodologies:

We follow a structured methodology for our projects which starts from designing the solution to the implementation phase. Well planned Project reduces the time to deliver the project and any additional ad-hoc costs to our clients, hence we dedicate most of our time understanding our client's business and gather requirements. This ground-up approach helps us deliver not only the solution to our clients but also add value to your investments.

Key parts of the report:

Under each division we further provide specific industry solutions on focused domains with cutting edge technologies.

Benefits of the Company/Institution through our report:

Under each division we further provide specific industry solution on focused domains with cutting edge technologies. We emphasize building relationships with our clients by delivering projects on time and within budget.

INDEX

S.no	CONTENTS	Page no
1.	Introduction.....	1
	1.1 Modules	2
2.	Software requirements specifications	4
3.	Technology	5
	3.1 HTML	5
	3.2 CSS	6
	3.3 JAVA SCRIPT.....	7
	3.4 HTML 5	7
4.	Coding.....	10
5.	Screenshots	11
6.	Conclusion	16
7.	Bibliography	17

Learning Objectives/Internship Objectives

- Internships are generally thought of to be reserved for college students looking to gain experience in a particular field. However, a wide array of people can benefit from Training Internships to receive real world experience and develop their skills.
- An objective for this position should emphasize the skills you already possess in the area and your interest in learning more.
- Internships are utilized in a few different career fields, including architecture, engineering, healthcare, economics, advertising and many more.
- Some internships are used to allow individuals to perform scientific research while others are specifically designed to allow people to gain first-hand experience working.
- Utilizing internships is a great way to build your resume and develop skills that can be emphasized in your resume for future jobs. When you are applying for a Training Internship, make sure to highlight any special skills or talents that can make you stand apart from the rest of the applicants so that you have an improved chance of landing the position.

WEEKLY OVERVIEW OF INTERNSHIP ACTIVITIES

1st WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	8/05/17	Monday	Introduction of HTML framework
	9/05/17	Tuesday	Features of HTML
	10/05/17	Wednesday	About HTML structural elements
	11/05/17	Thursday	About text, content, list, and table elements
	12/05/17	Friday	About hyperlink, form, semantic elements with HTML attributes
	13/05/17	Saturday	Understanding different types of HTML entities and comments

2nd WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	15/05/17	Monday	Introduction to CSS
	16/05/17	Tuesday	About selectors, properties, values, and CSS Box specificity
	17/05/17	Wednesday	About cascading, specificity, and media queries
	18/05/17	Thursday	About Pseudo-classes and Pseudo-elements
	19/05/17	Friday	About CSS preprocessors
	20/05/17	Saturday	CSS frameworks and libraries

3 rd WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	22/05/17	Monday	Introduction to JavaScript
	23/05/17	Tuesday	About its syntax, variables, data types, and operators
	24/05/17	Wednesday	About control structures, functions, OOP, and events
	25/05/17	Thursday	About DOM, asynchronous JavaScript, and XML
	26/05/17	Friday	About ES6, modern features, libraries, and frameworks
	27/05/17	Saturday	JSON, module systems and error handling

4 th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	29/05/17	Monday	Practicing HTML
	30/05/17	Tuesday	Hand-on coding on HTML 5
	31/05/17	Wednesday	Practicing CSS
	01/06/17	Thursday	Practicing JAVASCRIPT
	02/06/17	Friday	Debugging and developing basic projects
	03/06/17	Saturday	Real Projects session

5 th WEEK	DATE	DAY	NAME OF THE TOPIC/MODULE COMPLETED
	05/06/17	Monday	Design & Analysis
	06/06/17	Tuesday	Coding
	07/06/17	Wednesday	Testing

1. INTRODUCTION

Web development is a multifaceted field that encompasses the process of creating and maintaining websites and web applications. It's a dynamic and ever-evolving domain that plays a pivotal role in the digital age. Web developers need a strong understanding of various technologies, including HTML, CSS, JavaScript, and more. They work on both the front-end, where user interfaces are designed and developed, and the back end, where server-side logic, databases, and application functionality are handled. Full-stack developers bridge the gap between these two realms. The demand for web development skills is substantial as businesses and organizations rely on the internet to reach their audiences and conduct business. Therefore, web developers need to stay up to date with the latest technologies, trends, and best practices.

However, web development comes with its share of challenges. Keeping up with the fast-paced nature of the industry can be demanding, as new technologies and frameworks continually emerge. Security is a critical concern, as websites and web applications are vulnerable to various cyber threats. Ensuring cross-browser compatibility and responsive design, especially with the increasing variety of devices and screen sizes, can be a complex task. Accessibility, which involves making web content usable for people with disabilities, is another challenge, as it requires adherence to established guidelines and standards. Additionally, optimizing web performance and handling scalability issues are ongoing concerns to provide a smooth and efficient user experience.

To address the needs and requirements of web development, developers must have a strong foundation in HTML, CSS, and JavaScript. They often use front-end libraries and frameworks like React, Angular, or Vue.js to streamline the development of interactive user interfaces. Back-end development necessitates proficiency in server-side languages such as Node.js, Python, Ruby, or PHP, along with databases like MySQL or MongoDB. Collaboration and version control with tools like Git are essential for teamwork. In addition to technical skills, web developers benefit from good communication and problem-solving abilities, as they often work closely with clients and team members to understand project requirements and find solutions to challenges. Continuous learning and adaptation to the ever-changing landscape of web technologies are vital for success in this field.

HTML, CSS, and JavaScript are the foundational trio of web development that collectively offer distinct advantages in crafting dynamic and visually appealing websites and web applications. HTML (Hypertext Markup Language) provides the structural backbone for web content, ensuring logical organization and accessibility. CSS (Cascading Style Sheets) enhances user experience by enabling precise styling, layout control, and responsive design, elevating aesthetics, and usability. JavaScript, a versatile and client-side scripting language, empowers interactivity and dynamic behavior, from form validation to real-time updates, delivering engaging and responsive web experiences. Together, these technologies create a seamless, user-friendly, and feature-rich web environment that underpins the digital world's vast array of content and applications.

The landscape of web development continues to evolve with the introduction of new features and trends. Web Assembly (Wasm) is gaining prominence, allowing for high-performance code execution in web browsers, enabling complex tasks to be handled with impressive speed. The JAM stack architecture is redefining web development by decoupling the fronted from the backend, emphasizing improved performance and security.

2. SOFTWARE REQUIREMENTS SPECIFICATIONS

Web development involves creating websites and web applications using various technologies. The software requirements for web development can vary depending on the specific technologies and languages you are using, but here are some common software requirements and tools you may need:

1. *Text Editor or Integrated Development Environment (IDE)*:

- Text editors like Visual Studio Code, Sublime Text, or Atom are popular for writing HTML, CSS, and JavaScript.
- IDEs like WebStorm, Visual Studio, or IntelliJ IDEA provide a more comprehensive environment with features for web development.

2. *Web Browsers*:

- You'll need multiple web browsers for testing and debugging your web applications. Common choices include Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari.

3. *Version Control*:

- Git and a Git client (e.g., GitKraken, Sourcetree) for tracking changes and collaborating on projects. Platforms like GitHub, GitLab, or Bitbucket for code hosting and collaboration.

4. *Package Managers*:

- Package managers like npm (Node Package Manager) for JavaScript, Yarn, or Composer for PHP help manage and install project dependencies.

5. *Local Development Server*:

- A local web server environment is essential for testing and debugging web applications. Popular choices include Apache, Nginx, XAMPP, WAMP, or tools like MAMP for macOS.

6. *Database Management Tools*:

- Database management software such as phpMyAdmin for MySQL or tools like SQL Server Management Studio for Microsoft SQL Server.

7. *Front-End Libraries and Frameworks*:

- Depending on your project requirements, you may need libraries or frameworks like React, Angular, Vue.js, Bootstrap, or Foundation to streamline front-end development.

8. *Back-End Technologies*:

- If you're working on the back end, you'll need appropriate server-side technologies, such as Node.js, Ruby on Rails, Django, or .NET, along with related tools and frameworks.

9. *Text and Image Editors*:

- Graphic design and image-editing software like Adobe Photoshop, Adobe Illustrator, or GIMP for creating and editing web graphics.

- Text editors for writing code or content.

10. *Task Runners and Build Tools*:

- Task runners like Gulp and Grunt automate repetitive tasks, while build tools like Webpack or Babel help optimize code.

11. *API Development Tools*:

- If you're developing APIs, tools like Postman can help test and document your APIs.

12. *Content Management Systems (CMS)*:

- Depending on your project, you might need to install and configure a CMS like WordPress, Drupal, or Joomla.

13. *Command-Line Interface (CLI)*:

- Familiarity with a command-line interface is helpful for various development tasks, including version control and task automation.

14. *Web Performance Testing Tools*:

- Tools like Google Lighthouse, GTmetrix, or PageSpeed Insights can help assess and improve the performance of your web applications.

15. *Debugging Tools*:

- Browser developer tools and debugging extensions like Redux DevTools or React DevTools for front-end debugging.

- IDE-integrated debugging tools for server-side debugging.

These are some of the software requirements and tools commonly used in web development. The specific tools you need may vary based on your project, your chosen programming languages, and your personal preferences.

3. TECHNOLOGY

3.1 HTML

HTML, or Hypertext Markup Language, is the standard markup language used to create web pages. It is the fundamental building block of the World Wide Web and plays a central role in web development. HTML defines the structure and content of a web page by using a system of elements and tags. HTML is a markup language, not a programming language. It uses tags to define the structure of a web page, rather than writing code to perform operations. HTML documents are composed of elements enclosed in tags. Elements are the building blocks of a web page, and tags define how those elements are displayed. HTML provides the structural framework for a web page.

Common elements include headings (`

`, ``, etc.), paragraphs (` `), lists (` `, ` `, ` - `), tables (` | | |---| | `), and more. HTML is used to display text content on a web page. This includes text formatting (bold, italic, etc.), links (` <a>`), and text-based elements. HTML allows you to display images using the `` element, with attributes like `src` for the image source and `alt` for alternative text. Links are created using the `<a>` element, allowing users to navigate to other web pages, resources, or email addresses. HTML provides elements for creating forms, including text fields, radio buttons, checkboxes, and submit buttons. Forms are used for user input and data submission. | |---|

HTML5 introduced semantic elements like ``, ``, ``, and ``, which add meaning and structure to the content, making it more accessible and SEO-friendly. HTML elements often have attributes that provide additional information or modify their behavior. For example, the `class` and `id` attributes are used for styling and JavaScript integration. HTML documents should adhere to web standards and be validated to ensure they display correctly in different browsers. Web accessibility is a crucial aspect of HTML, and developers are encouraged to create web content that is usable by people with disabilities. This includes using proper semantic elements and attributes.

While HTML defines the structure and content of a web page, CSS is used to control the presentation and styling. HTML and CSS often work together to create visually appealing web pages. HTML can be enhanced with JavaScript to add interactivity and dynamic behavior to web pages. JavaScript can be embedded directly within HTML or linked as external scripts. Understanding HTML is a fundamental skill for anyone involved in web development, whether you're a beginner or an experienced developer. HTML is the backbone of web development, providing the foundation upon which content and structure are built for websites and web applications.

HTML, as the backbone of web development, offers numerous advantages, such as its simplicity and widespread support, enabling content structuring and accessibility across various browsers and devices. It fosters web standardization, easing content sharing, and is complemented by CSS and JavaScript for enhanced styling and interactivity. However, HTML has limitations, including its primarily static nature, which necessitates the integration of CSS and JavaScript for dynamic and visually appealing web content. It may also pose challenges for complex layouts and may require additional technologies for server-side processing, potentially resulting in longer load times for more intricate web applications. Nevertheless, HTML's foundational role in web development remains invaluable while requiring synergy with other technologies for optimal results.

3.2 CSS

Cascading Style Sheets (CSS) is a powerful language in web development that plays a central role in defining the visual appearance of web pages. CSS is a key component for designing and formatting web content, allowing developers to control elements' colors, fonts, spacing, layout, and other visual properties. It works hand in hand with HTML, which focuses on structuring content, to create the full user experience on the web.

Selectors and properties are the building blocks of CSS. Selectors target HTML elements, while properties define how those elements should be styled. For instance, you can use selectors like `p` to style paragraphs, or you can employ class selectors like `.my-class` to apply styles to specific elements. CSS properties encompass a wide range of styling options, such as setting the text color, defining font sizes, and controlling margins and padding, allowing developers to tailor the appearance of web content to their preferences.

The CSS box model is a fundamental concept, defining how elements are displayed. It outlines the content, padding, border, and margin for each element, shaping the layout of web pages. Additionally, CSS introduces the concept of "cascading," signifying that styles can be inherited, overwritten, or combined from multiple sources. Specificity rules determine the order of precedence for styles, resolving conflicts when different sources provide conflicting instructions.

CSS also excels in enabling responsive design. With techniques like media queries, CSS allows developers to adapt their designs to different screen sizes and devices, ensuring that web content is accessible and looks appealing on a variety of platforms. Furthermore, CSS frameworks and libraries, such as Bootstrap, Foundation, and Materialize, simplify web development by providing pre-designed components and responsive grids for consistent and efficient design.

In recent years, CSS has evolved with features like animations and transitions. CSS3 introduced the capability to create animations and transitions directly within CSS, offering effects such as fading, sliding, and scaling without the need for JavaScript. CSS has also benefited from the introduction of layout models like Flexbox and Grid, which simplify the creation of complex and responsive web layouts, reducing the need for complex positioning.

Cross-browser compatibility is a crucial aspect of CSS, ensuring that web designs maintain consistency across various browsers. It plays an essential role in achieving a uniform look and feel for web content regardless of the user's choice of browser. CSS also contributes to accessibility by improving the usability of web content for people with disabilities. Properly crafted CSS can enhance the accessibility of web pages and is often used in conjunction with HTML5 semantic elements to further support accessibility.

CSS continues to evolve with new features and capabilities that enhance web design and development. Recent additions include CSS Grid Layout, offering precise control over webpage layouts, and CSS Flexbox for creating flexible and responsive designs with ease. CSS Custom Properties (Variables) allow for dynamic styling and theming, while CSS Scroll Snap allows smooth scrolling and snapping to defined points. CSS Scroll Behavior enhances the scrolling experience by providing smooth animations and control over scroll behavior. CSS-in-JS solutions, like Styled-Components, bridge the gap between JavaScript and CSS for more dynamic and modular styling. CSS Grid and Flexbox continue to be prominent in creating complex and responsive layouts. These features, among others, contribute to a more dynamic and user-friendly web design process.

In summary, CSS is a versatile and indispensable tool for web designers and developers. It determines the visual aspects of web content, enabling the creation of visually appealing, responsive, and user-friendly web experiences when combined with HTML and JavaScript. Mastery of CSS is fundamental for crafting engaging and effective websites and web applications in the field of web development.

3.3 JavaScript

JavaScript, often referred to as ECMAScript, is a highly versatile and widely used programming language primarily employed for web development, although its applications extend to various other domains. JavaScript is most used as a client-side scripting language, running within web browsers and enabling web developers to imbue web pages with interactivity, dynamic behavior, and real-time features. It plays a pivotal role in enhancing user experiences on the web. JavaScript adheres to the ECMAScript standard, which defines its syntax and features. Periodic updates, such as ES6 (ECMAScript 2015), have introduced modern language features like arrow functions, classes, and template literals, keeping the language relevant and powerful.

JavaScript employs C-style syntax, combining statements, variables, data types, and operators to create structured and functional code. Punctuation and keywords are used to define the logic within JavaScript. The language supports a variety of data types, including numbers, strings, Booleans, objects, arrays, and more. Variables are declared using `var`, `let`, or `const`, and functions are a fundamental aspect of JavaScript, allowing code to be organized into reusable blocks that can take parameters, perform actions, and return values. JavaScript encompasses mathematical, comparison, and logical operators for calculations, decision-making, and condition evaluation. Control structures like `if` statements, `for` and `while` loops, and `switch` statements facilitate decision-making and repetitive tasks.

Being an object-oriented language, JavaScript enables the creation and manipulation of objects, supports constructors, prototypes, classes, and methods. Events play a crucial role in JavaScript, as they allow web pages to respond to user interactions and browser events. Events like click, mouseover, and keypress are commonly used. JavaScript is particularly adept at manipulating the Document Object Model (DOM) of web pages. This allows developers to dynamically change the content and structure of web pages by selecting, modifying, and creating HTML elements and attributes.

Asynchronous programming is a core feature of JavaScript, handling tasks like data fetching and server communication through callbacks, promises, and the `async/await` mechanism to ensure web applications remain responsive. JavaScript's capabilities also extend to AJAX (Asynchronous JavaScript and XML), which enables asynchronous requests to the server, facilitating data fetching and sending without the need to reload the entire web page. Technologies like XMLHttpRequest and the modern Fetch API make AJAX requests accessible. With ES6, JavaScript introduced modules, which promote code organization and reusability by enabling the export and import of functions, classes, and objects. This modular approach streamlines development and enhances maintainability.

JavaScript is commonly used in conjunction with libraries and frameworks like jQuery, React, Angular, and Vue.js, which provide a structured approach to development and add advanced features to web applications. The language's adaptability and wide ecosystem make it suitable for a wide range of projects, from server-side development with Node.js to game development and various other applications. Its adaptability, extensive community support, and wide-ranging applications cement its position as a fundamental language in modern software development.

3.4 HTML 5

HTML5, the fifth and latest version of Hypertext Markup Language, represents a significant evolution in web development compared to its predecessor, HTML. HTML5 introduces numerous new elements, features, and capabilities that enhance the creation of web content and applications. It offers advanced multimedia support through the `<video>` and `<audio>` elements, enabling seamless integration of video and audio content without the need for third-party plugins like Flash. HTML5 also introduces the `<canvas>` element, providing a platform for dynamic graphics and animations, thereby reducing the reliance on external technologies.

Another noteworthy addition is the support for geolocation, allowing web applications to access a user's location, enabling functionalities like mapping services and location-based features. Additionally, HTML5 includes a robust set of form elements and attributes that simplify data input, validation, and customization. The `<input>` element, for instance, offers new input types like email and date, making it easier to collect specific types of data.

Moreover, HTML5 fosters improved semantics with elements such as `<header>`, `<nav>`, and `<section>`, enhancing content structure and search engine optimization. The introduction of the `<article>` and `<aside>` elements provides more meaningful markup for articles and side content. HTML5 also empowers web developers to build more accessible websites through the integration of elements like `<figure>` and `<figcaption>` for image and caption relationships.

Furthermore, HTML5 supports offline web applications using the Application Cache feature, making it possible for users to access web content even without an internet connection. Local storage mechanisms like localStorage and sessionStorage allow web applications to store data on the client side, reducing the reliance on server-side resources.

HTML5 brings substantial advantages to modern web development, offering native support for multimedia, dynamic graphics with the `<canvas>` element, and improved semantics with semantic tags, enhancing both the developer and user experience. The addition of advanced form elements, geolocation support, and improved accessibility features further enrich web content. Offline capabilities through the Application Cache and local storage options provide enhanced user engagement and performance optimization. However, HTML5 may face challenges such as browser compatibility issues, a learning curve for developers, security concerns when implementing advanced features, and the complexity of managing multimedia licensing and accessibility, emphasizing the importance of careful implementation and consideration of these factors in web development projects.

In summary, HTML5 brings substantial enhancements to web development compared to HTML, including improved multimedia support, canvas for graphics, geolocation capabilities, advanced form elements, enhanced semantics, accessibility features, offline functionality, and local storage options. These features collectively contribute to a richer and more responsive web experience, making HTML5 the preferred choice for modern web development.

CODING:

Product Landing Pages

I have created a Product Landing Pages website which is presenting about me, my certifications, etc. I have tried my best to create a functional and error free website. It took eight hours of coding.

```
197 <div>
198   
199 </div>
200 </div>
201 </div>
202 <!--===== WORK =====-->
203 <section class="work_section" id="work">
204   <h2 class="section-title">Work</h2>
205
206   <div class="work_container bd-grid">
207     <a href="#" class="work_img">
208       
209     </a>
210     <a href="#" class="work_img">
211       
212     </a>
213     <a href="#" class="work_img">
214       
215     </a>
216     <a href="#" class="work_img">
217       
218     </a>
219     <a href="#" class="work_img">
220       
221     </a>
222     <a href="#" class="work_img">
223       
224     </a>
225   </div>
226 </section>
227
228 <!--===== CONTACT =====-->
229 <section class="contact_section" id="contact">
230   <h2 class="section-title">Contact</h2>
231
232   <div class="contact_container bd-grid">
233     <form action="#" class="contact_form">
234       <input type="text" placeholder="Name" class="contact_input">
235       <input type="mail" placeholder="Email" class="contact_input">
236       <textarea name="id" cols="8" rows="10" class="contact_input"></textarea>
237       <input type="button" value="Send" class="contact_button button">
238     </form>
239   </div>
240 </section>
241 </main>
242
243 <!--===== FOOTER =====-->
244 <footer class="footer">
245   <p class="Footer_title">Dhairya</p>
246   <p class="Footer_title">jairbhati@gmail.com</p>
247   <p class="Footer_title">AIT*BFI</p>
248   <div class="Footer_social">
249     <a href="#" class="Footer_icon"><i class="bx bx-facebook"></i></a>
250     <a href="#" class="Footer_icon"><i class="bx bx-instagram"></i></a>
251     <a href="#" class="Footer_icon"><i class="bx bx-twitter"></i></a>
252   </div>
253   <p class="Footer_copy">©169; AIT*BFI. All rights reserved.</p>
254 </footer>
255
256 <!--===== SCROLL REVEAL =====-->
257 <script src="https://unpkg.com/scrollreveal"></script>
258
259 <!--===== MAIN JS =====-->
260 <script src="assets/js/main.js"></script>
261 </body>
262 </html>
```


Quize Application:

I have created a Quize Application. A landing page is a standalone web page, created specifically for a marketing or advertising campaign.

```
<!DOCTYPE Html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="description" content="Beams coffee" />
    <meta href="https://cdn.jsdelivr.net/npm/bootstrap@0.5/css/bootstrap.min.css" rel="stylesheet">

    <link href="css/main.css" type="text/css" rel="stylesheet"/>
    <link rel="stylesheet" href="assets/css/styles.css">
    <title>Never trust anyone who doesn't drink coffee...</title>
  </head>
  <!--===== HEADER =====-->
  <header class="l-header">
    <nav menu="bd-grid">
      <div>
        <a href="#" class="nav__logo">AIT*BFI Products</a>
      </div>

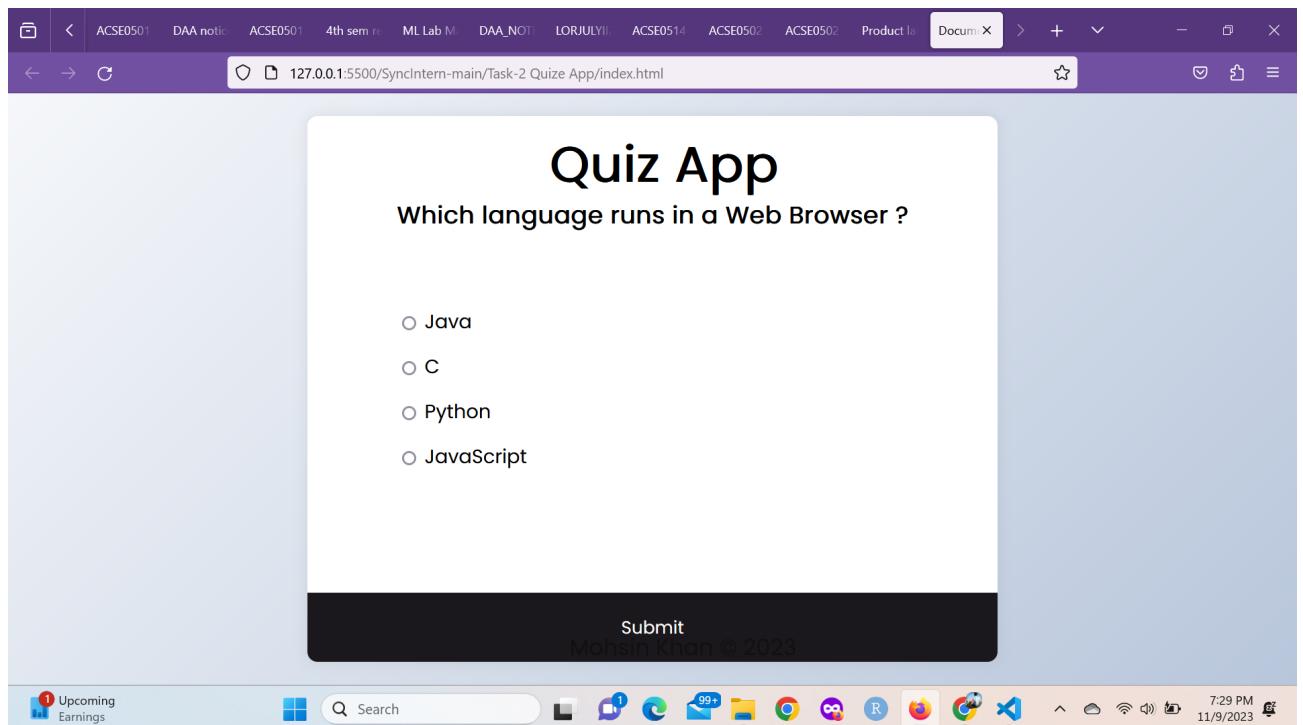
      <div class="nav__toggle" id="nav-toggle">
        <i class="bx bx-menu"></i>
      </div>

      <div class="nav__menu" id="nav-menu">
        <div class="nav__close" id="nav-close">
          <i class="bx bx-x"></i>
        </div>

        <ul class="nav__list">
          <li class="nav__item"><a href="#">Home</a><span>nav_link active</span></li>
          <li class="nav__item"><a href="#">About</a><span>nav_link</span></li>
          <li class="nav__item"><a href="#">Skills</a><span>nav_link</span></li>
          <li class="nav__item"><a href="#">Contact</a><span>nav_link</span></li>
        </ul>
      </div>
    </nav>
  </header>

  <main class="l-main">
    <!--===== HOME =====-->
    <section home id="home">
      <div class="home__grid bd-grid">
        <div class="home__img">
          
          
          
          
          
          
        </div>
        <div class="home__data">
          <h1 style="color: #rgb(136, 14, 67)">Cup Beans<br>Take a Chars<br>Together</h1>
          <p class="home__description">Let's us help you in discovering the best<br>coffee drink for this beautiful day. <br>or of the week.</p>
          <a href="#" class="home__button">Get Started</a>
        </div>
      </div>
    </section>
  </main>

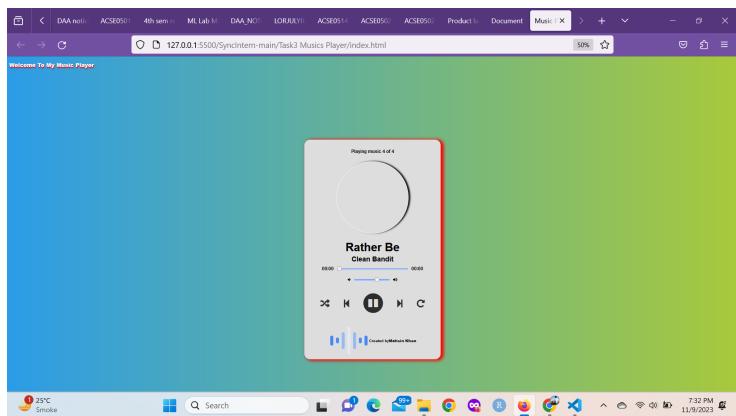
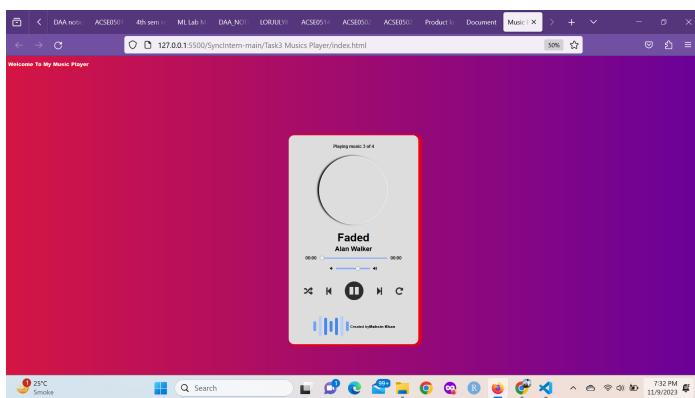
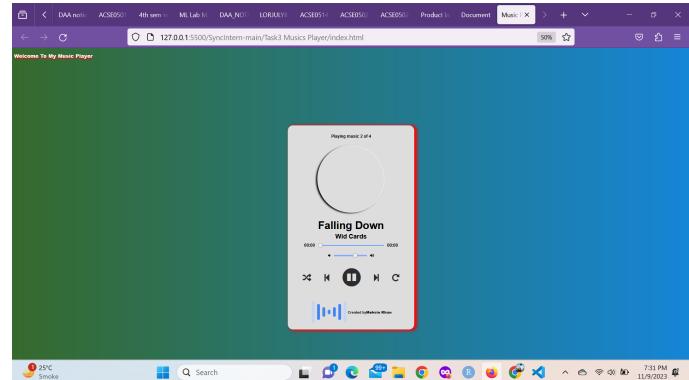
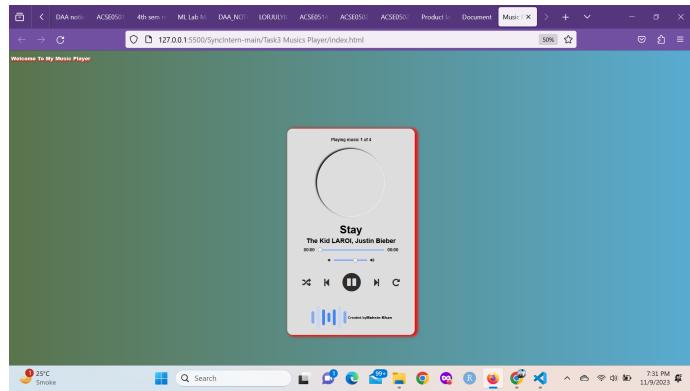
  <div class="about__container bd-grid">
    <div class="about__img">
      
    </div>
  </div>
```



Musics Player Application:

I have also created a simple Music Player Application using my web development knowledge.

```
index.html <--> C:\Users\jairb\OneDrive\Desktop\folder>PROJECT>internship>Calculator>Calculator>index.html ...  
1 <!DOCTYPE html>  
2 <html lang="en">  
3 <head>  
4 <meta charset="utf-8" />  
5 <meta http-equiv="X-UA-Compatible" content="IE=edge" />  
6 <title>AI+BF1 calc</title>  
7 <meta name="description" content="Simple Calculator With Dark Mode." />  
8 <meta name="viewport" content="width=device-width, initial-scale=1" />  
9 <link rel="icon" type="image/x-icon" href="assets/calculator.ico" />  
10 <link rel="stylesheet" href="assets/style.css" />  
11 <link rel="preconnect" href="https://fonts.googleapis.com" />  
12 <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin="crossorigin" />  
13 <link href="https://fonts.googleapis.com/css2?family=Inter:wght@300;400&display=swap" rel="stylesheet">  
14 </head>  
15 <body>  
16 <div class="wrapper">  
17 <div class="container">  
18 <div class="header-container">  
19 <!-- This heading also notifies about the theme change.-->  
20 <h1 id="toast">Calculator</h1>  
21 <div class="top-buttons">  
22 <button type="button" onclick="changeTheme()" class="theme-button">  
23   
24 </button>  
25 </div>  
26 </div>  
27 <div class="first-row">  
28 <input type="text" name="result" id="result" placeholder="Result" readonly />  
29 <input type="button" value="C" onclick="result.value=''" id="clear-button" />  
30 </div>  
31 <div class="second-row">  
32 <input type="button" value="1" onclick="liveScreen(1)" />  
33 <input type="button" value="2" onclick="liveScreen(2)" />  
34 <input type="button" value="3" onclick="liveScreen(3)" />  
35 <input type="button" value="+" onclick="liveScreen('+')"/>  
36 </div>  
37 <div class="third-row">  
38 <input type="button" value="4" onclick="liveScreen(4)" />  
39 <input type="button" value="5" onclick="liveScreen(5)" />  
40 <input type="button" value="6" onclick="liveScreen(6)" />  
41 <input type="button" value="-" onclick="liveScreen('-')"/>  
42 </div>  
43 <div class="fourth-row">  
44 <input type="button" value="7" onclick="liveScreen(7)" />  
45 <input type="button" value="8" onclick="liveScreen(8)" />  
46 <input type="button" value="9" onclick="liveScreen(9)" />  
47 <input type="button" value="*" onclick="liveScreen('*')"/>  
48 </div>  
49 <div class="fifth-row">  
50 <input type="button" value="/" onclick="liveScreen('/')"/>  
51 <input type="button" value="0" onclick="liveScreen(0)" />  
52 <input type="button" value"." onclick="liveScreen('.')"/>  
53 <input type="button" value="=" onclick="calculate(result.value)" />  
54 </div>  
55 </div>  
56 </div>  
57 <script src="scripts/script.js">  
58 </script>  
59 </body>  
60 </html>
```



CONCLUSION

In conclusion, this web development course has provided a comprehensive and invaluable learning experience for all of us. Throughout this journey, we have delved into the intricate world of web development, starting from the fundamentals, and progressing through various advanced concepts. As we wrap up this course, it's important to reflect on the knowledge and skills we have gained. We have learned the building blocks of web development, including HTML, CSS, and JavaScript, and how to combine them to create dynamic and visually appealing websites. We have explored responsive design principles, ensuring that our websites adapt to various screen sizes and devices, providing an optimal user experience. Diving deeper, we have tackled front-end frameworks and libraries like Bootstrap and React, making our web development process more efficient and enjoyable.

On the server-side, we have covered essential topics such as HTTP, APIs, and databases, enabling us to create interactive and data-driven web applications. We've also explored back-end technologies like Node.js, Python, and PHP, and worked with databases like MySQL and MongoDB. By understanding both front-end and back-end development, we are well-equipped to build complete web solutions.

Moreover, this course has emphasized the importance of best practices and code optimization, ensuring our websites are not only functional but also secure and performant. We've also gained insights into version control systems like Git and collaborative development workflows, which are essential skills for real-world web development projects.

The world of web development is constantly evolving, and our course has encouraged us to stay up to date with the latest industry trends and technologies. Continuous learning and adaptability will be crucial as we pursue our careers in web development. As we part ways with this course, we can take pride in the websites and web applications we've created, the bugs we've fixed, and the challenges we've overcome. Each line of code we've written, each design we've crafted, and each problem we've solved has contributed to our growth as web developers.