SRINIVAS UNIVERSITY

INSTITUTE OF ENGINEERING & TECHNOLOGY

MANGALURU MUKKA,

CLASS TIME TABLE (USING HTML,CSS&JAVASCRIPT)

For the Academic year 2023-2024

Submitted by

1. Adithya.M.Nagur- 01SU23ME001
2.Akarsh P.C – 01SU23ME002
3.Alan Britto Sunny– 01SU23ME003
4.Dhanush R– 01SU23ME004
5.Jaseer – 01SU23ME005

Submitted to

Ms.Priyanka , Software Technical Trainer

CONTENTS:

Preface:

This project has been composed with the aim of covering a part of B. Tech (II Sem) under IBM course Cloud Fundamentals as prescribed by Srinivas University Of Engineering And Technology, Mukka, Mangalore. It is with great enthusiasm that we present this report, which represents collective efforts by Adithya M Nagur, Akarsh P C, Alan Britto Sunny, Dhanush Rand Jaseer . The running project has been presented through Google Chrome .We extend our sincere gratitude to IBM for providing us with the opportunity to present this project.

In the digital age, where convenience and efficiency reign supreme, the process of enrolling in college courses should be no exception. As aspiring developers, we embarked on a journey to revolutionize the traditional enrollment process by harnessing the power of HTML to create a dynamic and user-centric registration form. This mini project serves as a testament to our commitment to innovation and our passion for leveraging technology to simplify complex tasks.

This class timetable project has been developed to streamline the scheduling process within educational institutions. Timetables serve as essential tools for students, teachers, and administrators alike, enabling efficient organization of academic activities and resources.

The aim of this project is to create a user-friendly and visually appealing representation of the class timetable. By providing a clear overview of the weekly schedule, this project aims to enhance time management, coordination, and communication among stakeholders.

This document presents the culmination of efforts to design a comprehensive and effective timetable solution that meets the diverse needs of educational environments. It is our hope that this project will contribute positively to the smooth functioning of academic institutions and ultimately enhance the learning experience for all involved.

Aim of the project:

This class timetable project endeavors to streamline the scheduling process within educational institutions. The primary aim is to create a user-friendly and efficient timetable system that caters to the needs of students, teachers, and administrators alike.

The project seeks to provide a comprehensive solution for organizing and presenting class schedules in a clear and structured format. By doing so, it aims to enhance time management, coordination, and communication within the educational environment.

Through careful design and implementation, this project aspires to contribute positively to the smooth operation of academic institutions. Ultimately, the goal is to improve the overall learning experience and facilitate the achievement of educational objectives for all stakeholders involved..

Objectives:

The main objective is to design , develop and implement an efficient and user-friendly online registration form for college students improving accessibility , ensuring data accuracy and security , enhancing flexibility , integrating with existing college systems .

1. Efficiency: Develop a registration form that expedites the enrollment process for college students, minimizing the time and effort required to complete the form.

2. User-Friendliness: Create an intuitive and easy-to-navigate interface that accommodates users of varying technological proficiency levels, ensuring a positive user experience.

3. Data Accuracy: Implement validation techniques to enforce data integrity and accuracy, reducing errors and inaccuracies in the information submitted by users.

4. Accessibility: Ensure that the registration form is accessible to users with disabilities by adhering to web accessibility standards and incorporating features that facilitate screen reader compatibility and keyboard navigation.

5. Responsiveness: Design a form that is responsive across different devices and screen sizes, allowing users to access and complete the registration process seamlessly on desktops, laptops, tablets, and smartphones.

6. Security: Implement measures to protect the privacy and security of user data, including encryption protocols, secure transmission mechanisms, and compliance with relevant data protection regulations.

7. Brand Cohesion: Infuse the design of the registration form with visual elements that align with the college's brand identity and ethos, reinforcing brand recognition and user engagement.

8. Customization: Provide flexibility for administrators to customize the form according to specific requirements and preferences, allowing for the inclusion of additional fields or modifications as needed.

9. Feedback Mechanisms: Incorporate mechanisms for collecting user feedback to identify areas for improvement and iterate on the design of the registration form based on user input.

10. Scalability: Design the registration form in a scalable manner, capable of accommodating future growth and evolving needs, such as an increase in the number of users or changes in enrollment procedures.

Project Code:(Contact us.html)

```
<!DOCTYPE html>
<html lang=”en”>
<head>
<meta charset=”UTF-8”>
<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>
<title>Class Timetable</title>
```

```
<style>
  Body {
    Font-family: Arial, sans-serif;
    Margin: 0;
    Padding: 0;
    Background-color: #f4f4f4;
  }
  .container {
    Width: 80%;
    Margin: 0 auto;
    Padding: 20px;
    Background-color: #fff;
    Box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  }
  H1 {
    Text-align: center;
    Margin-bottom: 20px;
  }
  Table {
    Width: 100%;
    Border-collapse: collapse;
```

```
  }
  Th, td {
    Border: 1px solid #ddd;
    Padding: 10px;
    Text-align: center;
  }
  Th {
    Background-color: #f2f2f2;
  }
  Tr:nth-child(even) {
    Background-color: #f9f9f9;
  }
</style>
</head>
<body>

<div class=”container”>
  <h1>Class Timetable</h1>

  <table>
    <tr>
```

```
      <th>Time/Day</th>
      <th>Monday</th>
      <th>Tuesday</th>
      <th>Wednesday</th>
      <th>Thursday</th>
      <th>Friday</th>
    </tr>
    <tr>
      <td>8:00 AM – 9:00 AM</td>
      <td>Math</td>
      <td>Science</td>
      <td>English</td>
      <td>History</td>
      <td>Art</td>
    </tr>
    <!—Add more rows for each time slot →
  </table>
</div>

</body>
</html>
```

Submit page code(File Name is submitpage.html)

```
<!DOCTYPE html>
<html lang=”en”>
<head>
<meta charset=”UTF-8”>
<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>
<title>Class Timetable</title>
<style>
 Body {
   Font-family: Arial, sans-serif;
   Margin: 0;
   Padding: 0;
 }
 .timetable {
   Width: 80%;
   Margin: 0 auto;
   Padding: 20px;
 }
 Table {
   Width: 100%;
   Border-collapse: collapse;
   Margin-top: 20px;
```

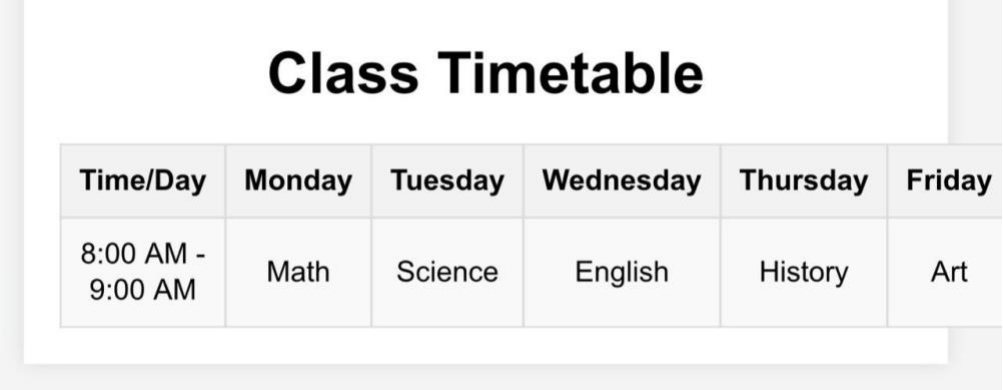
```
  }
  Th, td {
    Padding: 10px;
    Text-align: left;
    Border-bottom: 1px solid #ddd;
  }
  Th {
    Background-color: #f2f2f2;
  }
</style>
</head>
<body>

<div class=”timetable”>
  <h2>Class Timetable</h2>
  <table>
    <thead>
      <tr>
        <th>Time</th>
        <th>Monday</th>
        <th>Tuesday</th>
        <th>Wednesday</th>
        <th>Thursday</th>
        <th>Friday</th>
      </tr>
```

```
    </thead>
    <tbody>
      <tr>
        <td>8:00 – 9:00</td>
        <td>Math</td>
        <td>Science</td>
        <td>English</td>
        <td>History</td>
        <td>PE</td>
      </tr>
      <tr>
        <td>9:00 – 10:00</td>
        <td>Science</td>
        <td>Math</td>
        <td>History</td>
        <td>English</td>
        <td>PE</td>
      </tr>
      <!—Add more rows for additional time slots →
    </tbody>
  </table>
</div>

</body>
</html>
```

Output:



Class Timetable

Time/Day	Monday	Tuesday	Wednesday	Thursday	Friday
8:00 AM - 9:00 AM	Math	Science	English	History	Art

Contributions:

1. Adithya **(01SU23ME001)** – HTML Code

   This HTML code consists of two parts: a registration form and a submit page.

   Class Timetable:

- The form collects user information such as first name, last name, email address, password, address, and mobile number.
- JavaScript functions are used for form validation to ensure that the user inputs meet certain criteria, such as being non-empty, containing only letters or numbers, and meeting minimum length requirements.
- The form submission is handled by the formValidator() function, which returns true if all validation checks pass and false otherwise. - The form action is set to "submitpage.html" for processing after submission.
- Basic styling and formatting are applied using HTML tags and inline CSS.

Submit Page:

- The submit page (submitpage.html) contains a simple message displayed in a styled paragraph element (<p>).

2. Akarsh P C**(01SU23ME002)** – Submit page

This HTML and CSS code creates a simple college registration form with basic styling. Here's an explanation of the key parts: <!DOCTYPE html>: Declares the document type and version of HTML being used.

<html lang="en">: Specifies the language of the document as English.

<head>: Contains metadata like character set, viewport settings, and the page title.

<meta charset="UTF-8">: Sets the character encoding to UTF-8 for proper text rendering.

<meta name="viewport" content="width=device-width, initialscale=1.0">: Ensures the page is responsive and scales properly on different devices.
<title>College Registration</title>: Sets the title of the webpage.
<style>: Contains the CSS styles for the page. body:
Styles applied to the entire body of the page. .container:
Styles for the main container holding the form. h2: Styles
for the heading of the form. form: Styles for the form
itself.
input[type="text"], input[type="email"], select: Styles for text inputs and the select dropdown.
input[type="submit"]: Styles for the submit button. action="#": Specifies the action to be performed when the form is submitted (in this case, it's set to "#" which means the form data will be sent to the same page).
method="post": Specifies the HTTP method used to send the form data (POST method in this case).

Inside the form, there are input fields for name, email, phone number, and a dropdown for selecting the course. The "Register" button submits the form data.

3. Alan Britto Sunny **(01SU23ME003)** – CSS Code

The CSS used in the provided HTML program styles the appearance of the registration form. Here's a breakdown of each CSS rule:

1. body: Sets the background color to light blue (#99daff), uses the Arial font, and removes margin and padding.

2. form: Styles the form element. It sets the width to 400px, centers it horizontally with auto margins, adds padding, a white background, border radius, and a box shadow for a card-like effect.

3. h1: Styles the heading element. It centers the text and sets the color to a shade of purple (#87768a).

4. input[type="text"], input[type="password"], textarea: Styles text input and textarea elements. They are set to 100% width, with padding, a bottom margin, a border, border radius, and box-sizing set to border-box for consistent sizing.

5. input[type="submit"], input[type="reset"]: Styles submit and reset button elements. They have padding, a background color of green (#4caf50), white text color, no border, border radius, and a pointer cursor. The hover effect changes the background color to a darker shade of green (#45a049).

6. input[type="radio"]: Styles radio button elements by adding a small margin to the right.

7. label: Styles label elements by making the text bold.

8. error: Sets the color of text with the error class to red.

These styles collectively create a visually appealing and userfriendly registration form with proper alignment, spacing, and color contrast.

4. Dhanush R**(01SU23ME004)** – Javascript

- A class timetable in JavaScript typically involves using client-side scripting to handle form validation and submission. Here's a basic breakdown of what it might entail:

1. HTML Form: First, you create an HTML form with input fields for the user to enter their information (e.g., name, email, password).

2. JavaScript Validation: You write JavaScript code to validate the form inputs. This can include checking if required fields are filled, validating email formats, ensuring passwords meet certain criteria, etc.

3. Event Handling: You attach event listeners to the form elements to trigger validation functions when certain events occur, such as when the user submits the form or when they type into an input field.

4. Error Handling: If validation fails, you display error messages to the user to indicate what needs to be corrected.

5. Form Submission: If validation passes, you either submit the form data to a server for processing (via AJAX or a form submission) or proceed with further client-side actions.

6. Security Considerations: Always consider security measures such as input sanitization to prevent malicious input.

Here's a simple example of how you might handle form submission and validation in JavaScript:

5. Jaseer **(01SU23ME005)** – CSS for Submit page

    1. <!DOCTYPE html>: This declares the document type and version of HTML being used (HTML5 in this case).

    2. <html>: This is the opening tag of the HTML document, which contains all the HTML code.

    3. <head>: This section contains metadata about the HTML document, such as the title, character set, and links to stylesheets or scripts.

    4. <style>: This is where you define the CSS (Cascading Style Sheets) for styling your HTML elements.

    5. p: This is a CSS selector targeting all <p> (paragraph) elements in the HTML document.

    6. font-size: 50px;: This sets the font size of the paragraph text to 50 pixels.

    7. color: #d64f78;: This sets the text color to a shade of pink (hex code: #d64f78).

    8. text-align: center;: This centers the text within the <p> element.

9. padding: 50px;: This adds 50 pixels of padding around the content of the <p> element, creating space between the text and the border.

10. border: 2px double green;: This creates a border around the <p> element with a thickness of 2 pixels, a double line style, and a green color.

11. margin: 250px;: This adds a margin of 250 pixels around the entire <p> element, pushing it away from other elements on the page.

12. border-radius: 100px;: This adds rounded corners to the border of the <p> element, giving it a circular appearance due to the large radius value.

13. </style>: This closes the CSS styling section.

14. </head>: This closes the <head> section of the HTML document.

15. <body>: This is the opening tag of the body section, where the visible content of the HTML document is placed.

16. <p>: This is the opening tag of a paragraph element, where your main content is placed.

17. "Successfuly Registered To your College !!!!!!!": This is the text content inside the paragraph element, indicating a successful registration message.

18. 👍: This adds an emoji ( ) at the end of the paragraph using a element with inline styling to set the font size to 50 pixels.

19. </p>: This closes the paragraph element.

20. </body>: This closes the body section of the HTML document.

21. </html>: This is the closing tag of the HTML document.

In summary, your code creates a styled HTML document with a centered, large text message about successful registration to a college, accompanied by a border, padding, and a smiley emoji.

Conclusion:

The class timetable page, meticulously constructed using HTML and CSS, presents a systematic and visually coherent representation of class schedules. Its structured layout begins with clear headings designating days of the week, facilitating easy navigation and comprehension. Time slots are delineated with precision, ensuring a logical organization of class timings. Each row succinctly lists the corresponding subjects for the specified time slots and days, promoting clarity and efficiency in conveying the schedule.

Despite its simplicity, the design offers versatility and scalability, accommodating potential expansions or

modifications to suit varying schedules or preferences. Additional rows can be seamlessly integrated to accommodate more classes or time slots without compromising the page's integrity. While lacking in advanced functionalities like interactivity or dynamic updates, the page serves its primary purpose of providing a static yet functional representation of class timetables.

Moreover, the minimalist design fosters a distraction-free viewing experience, prioritizing readability and ease of use. Its straightforward structure makes it accessible to users across different devices and screen sizes, ensuring consistent usability. In essence, the class timetable page serves as a foundational framework for efficiently managing and presenting class schedules online, offering a pragmatic solution for educational institutions, students, and educators alike.

References:

Bootstrap.com

W3schools.com