Charotar University of Science and Technology [CHARUSAT]

Faculty of Technology and Engineering

Chandubhai S. Patel Institute of Technology(CSPIT) &

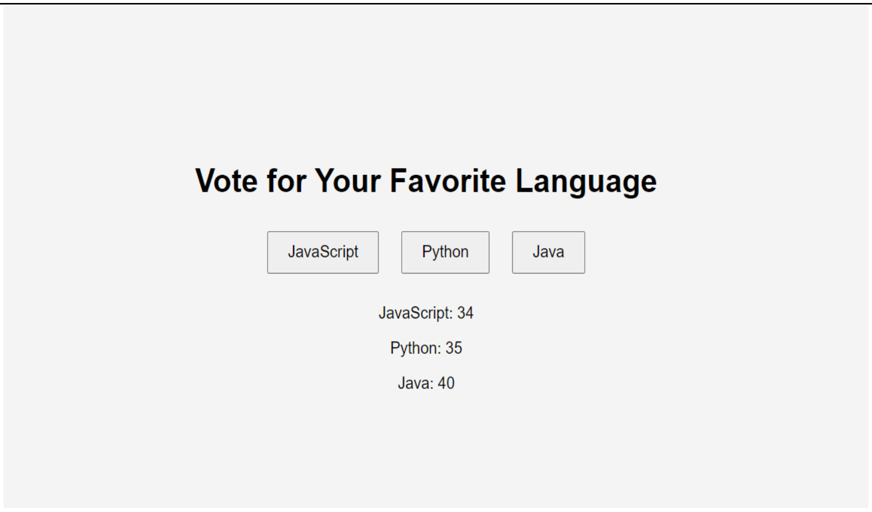
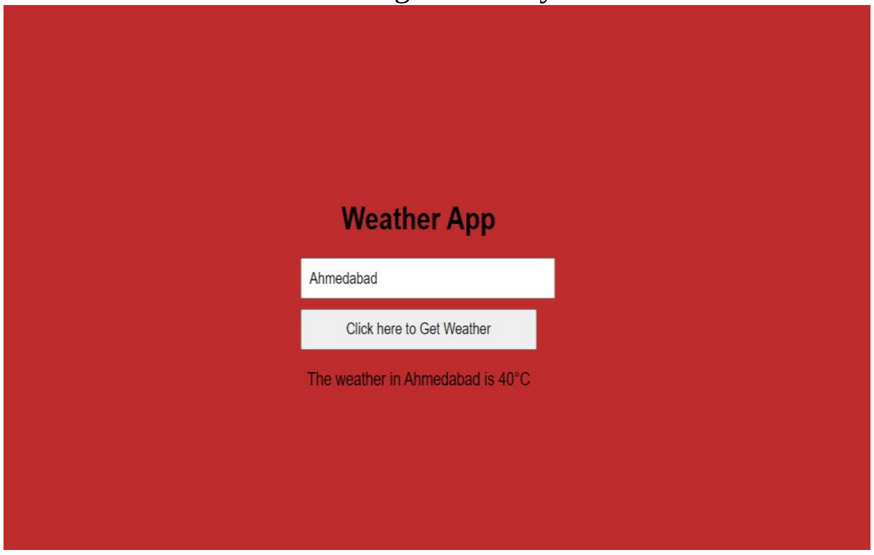
Devang Patel Institute of Advance Technology and Research(DEPSTAR)

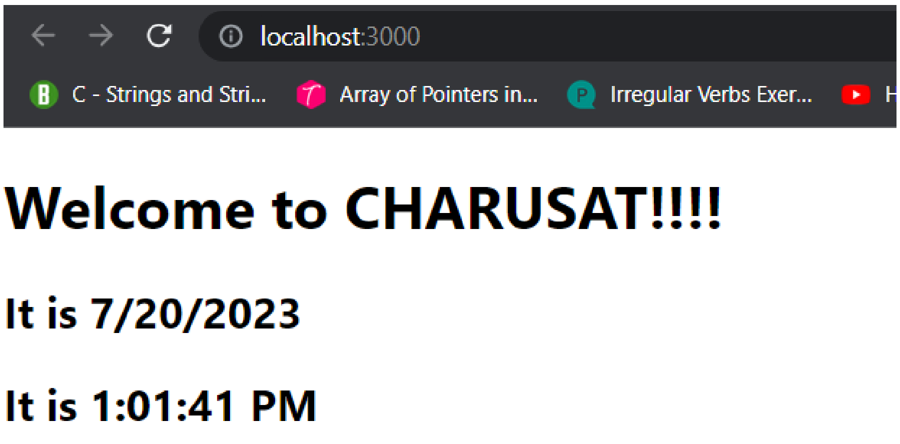
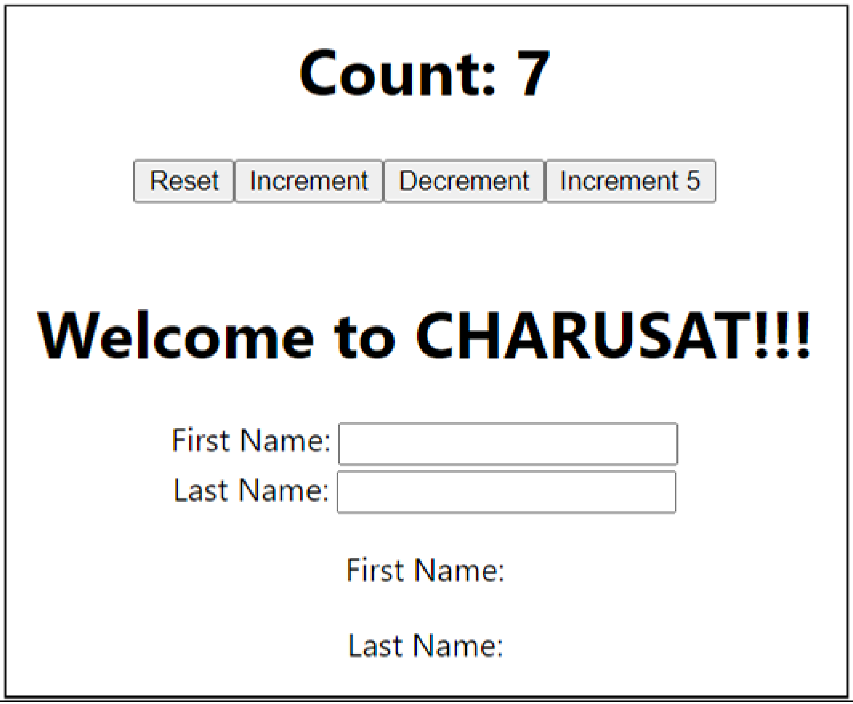
Department of Computer Science & Engineering

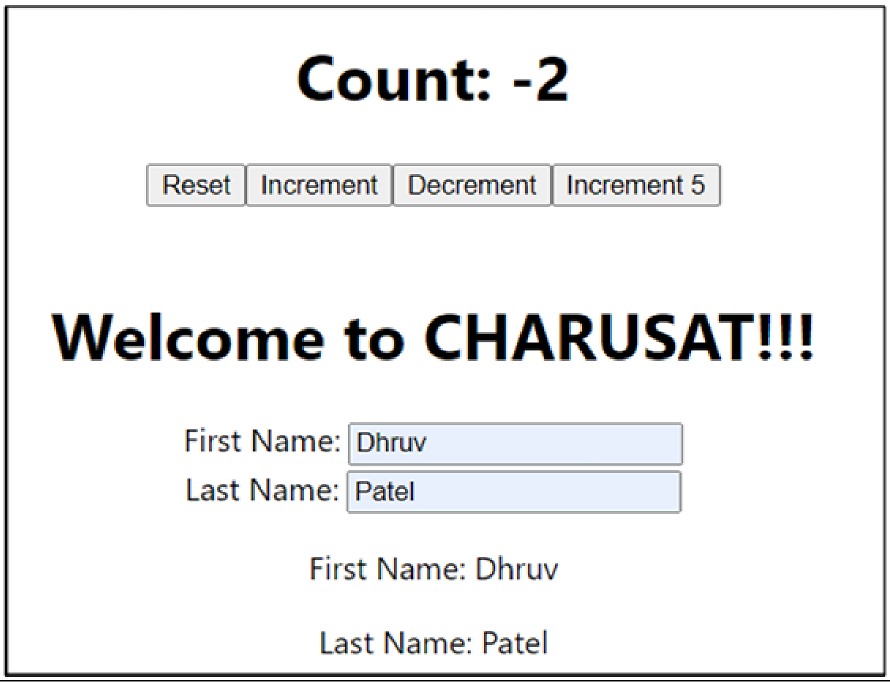
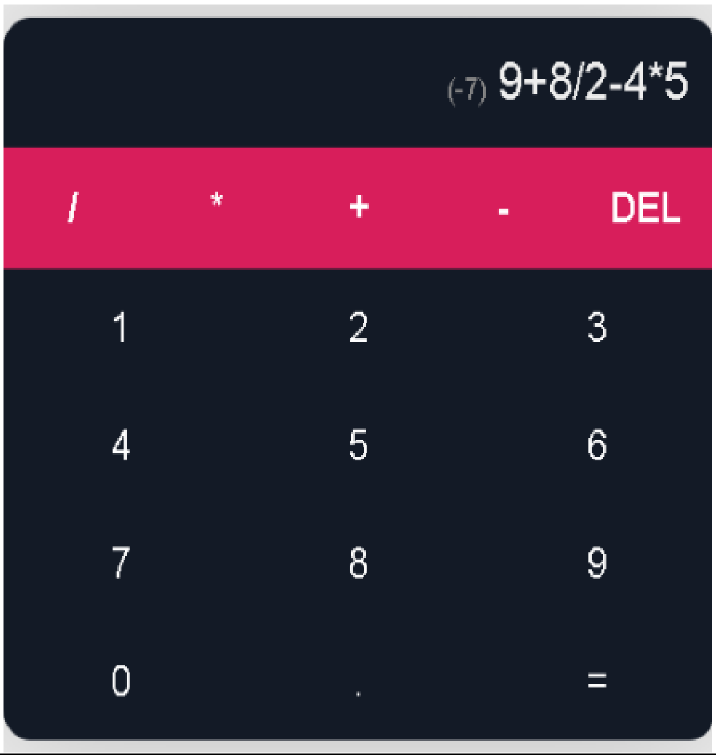
Practical List

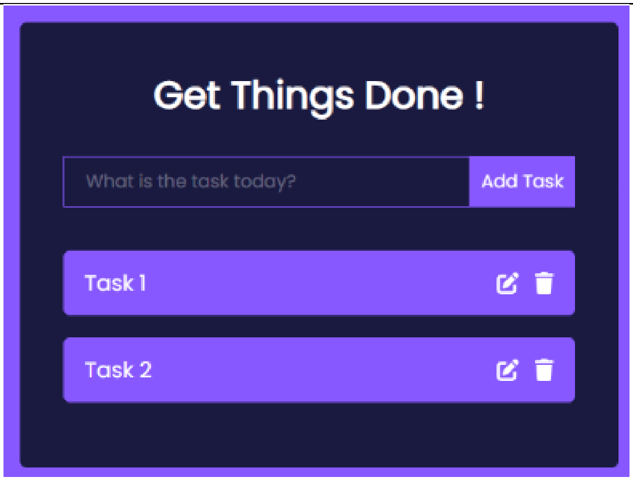
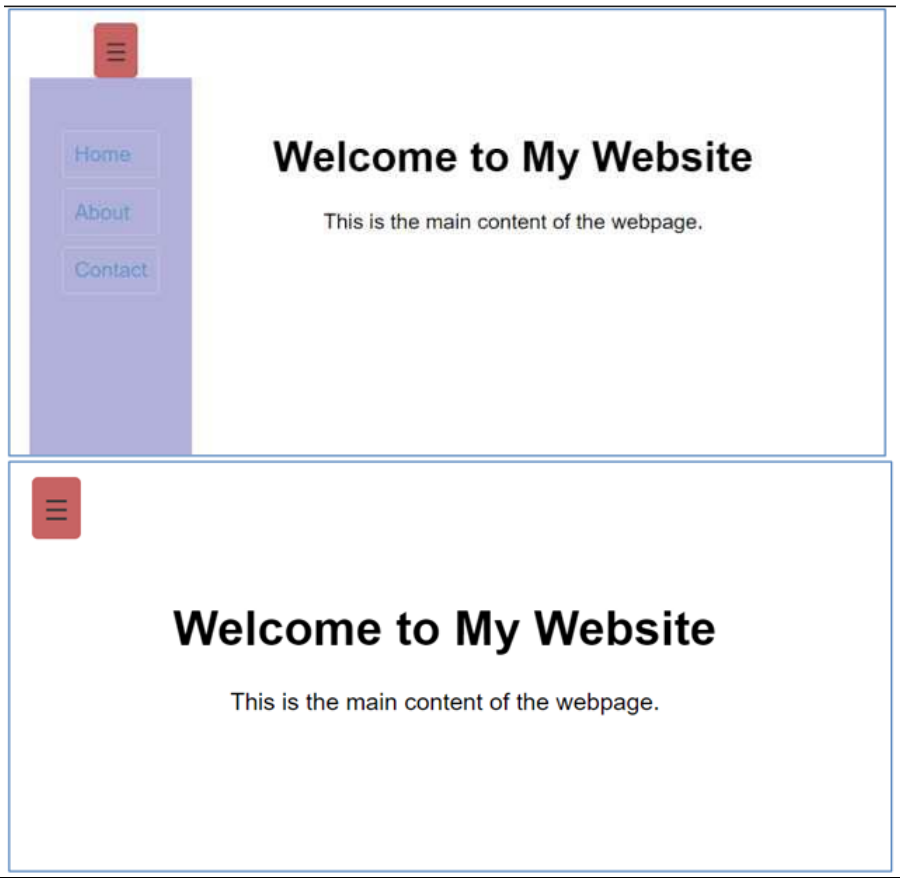
Subject code	:	CSE304	Semester	:	5	Academic Year	:	2025-26
Subject name	:	Full Stack Development						

Sr. No.	Aim	Hrs.	CO
	ReactJs Tasks		
1.	Create a real-time voting system where users can vote on a poll and see the results updated in real-time using only JavaScript, HTML, and CSS. **HTML:** A simple poll interface with buttons to vote and display the results. **CSS:** Styles the poll and results. **JavaScript:** Defines a vote function to update the local votes. Updates the vote counts in the UI. Simulates real-time voting by randomly incrementing votes. **Notes:** • **Votes object:** Keeps track of the current vote counts for each language. It initializes each language with 0 votes. • **vote function:** This function is called when a button is clicked. It increments the vote count for the selected language and calls updateVotes to refresh the displayed vote counts. • **updateVotes function:** Updates the displayed vote counts by setting the text content of the spans in the results section to the current vote counts. • **setInterval:** Simulates real-time updates by randomly incrementing the vote counts every 2 seconds. This mimics votes coming in from other users in real time.	3	1

	 Vote for Your Favorite Language JavaScript Python Java JavaScript: 34 Python: 35 Java: 40		
2.	Create a simple weather application that displays a hardcoded temperature for a given city. This simple weather application demonstrates basic HTML structure for user input, CSS styling for layout and appearance, and JavaScript functionality to handle user interactions and display dynamic content. It provides a foundational example of building an interactive web application using essential front-end technologies. Technologies Used: HTML, CSS, JavaScript (ES6) **Notes:** • **weatherData:** This object holds hardcoded weather information for specific cities. In a real-world scenario, this data would typically come from an API. • **addEventListener:** Listens for a click event on the Get Weather button. • **Fetching Weather:** When the button is clicked, it retrieves the city entered by the user, checks if weather data exists in weatherData, and displays the corresponding weather information or a message if the city is not found.  Weather App Ahmedabad Click here to Get Weather The weather in Ahmedabad is 40°C	3	1

3.	Create a React page that displays a Welcome message with the current local date and time which changes every second. Use the React JSX concept. Note: Use the setInterval function to update the time.  localhost:3000 C - Strings and Stri... Array of Pointers in... Irregular Verbs Exer... **Welcome to CHARUSAT!!!!** **It is 7/20/2023** **It is 1:01:41 PM**	3	2
4.	Create a Counter App that includes the functionality of Increment, Decrement, Reset, and Increment Five. The app also has two text boxes for first and surname names, the contents of which are shown as text on the same page. Utilize the React Hooks idea  **Count: 7** Reset Increment Decrement Increment 5 **Welcome to CHARUSAT!!!** First Name: Last Name: First Name: Last Name:	4	2

	 Count: -2 Reset Increment Decrement Increment 5 Welcome to CHARUSAT!!! First Name: Dhruv Last Name: Patel First Name: Dhruv Last Name: Patel		
5.	Create a calculator React app involves several steps, from setting up the React environment to implementing the logic for basic arithmetic operations. Here's a detailed explanation of creating a simple calculator app with addition, subtraction, multiplication, and division capabilities.  (-7) 9+8/2-4*5	4	2, 3
6.	Build a To-do List using React Hooks involves creating a user interface that allows users to add, remove, and delete tasks. This can be achieved using React's state management capabilities through hooks like useState and event handling for managing tasks.	4	2, 3

	 Get Things Done ! What is the task today? Add Task Task 1 Task 2		
7.	Creating a React Sidebar Navigation Menu involves using React Hooks to manage the state and handle the functionality of the sidebar, such as opening and closing the menu, and navigating between different sections of the webpage.  Home About Contact Welcome to My Website This is the main content of the webpage. Welcome to My Website This is the main content of the webpage.	4	2, 3
NodeJs and ExpressJs Tasks			
8.	You're hired to create a tool for gym users to count their exercise reps. They want a simple interface to increase or decrease the count with each move. Build a web-based counter that updates instantly when users click buttons. Make it easy to use on both mobile and desktop. Ensure the counter doesn't reset on page reload unless reset manually.	2	4

9.	Your manager asks you to set up a backend for a small product site. As a proof of concept, create a basic Express JS server. When someone visits the home route, it should display “Welcome to our site”. This helps your team understand the Express framework structure. Keep the code clean and scalable for future features.	2	4, 5
10.	A company stores error logs in .txt files on the server. You need to create a tool for developers to view these logs via a browser. Write an Express JS app that reads a text file and shows its content on a web page. Include error handling if the file is missing or inaccessible. This helps developers debug without accessing the server files directly.	1	5
11.	You're onboarding a new team and need a project template. Create a basic Express JS app with a /home route. When users visit this route, they should see a greeting message. This route will later serve as a dashboard page. Use this as the base for future web applications.	1	4, 5
12.	You’re building a simple web-based calculator for kids. Create a form where users enter two numbers. Use Express JS to perform basic operations: add, subtract, multiply, divide. Display the result on the screen. Ensure invalid inputs (like letters) are handled properly.	2	5
13.	A tax form website asks users to enter income from two sources. You need to take these numbers and display the total income. Use Express JS, accept inputs via POST, and show results using EJS templates. Validate user input and avoid client-side calculations. Keep the interface user-friendly and clear.	2	5
14.	You’re building a job portal where users upload resumes. Allow only PDF files up to 2MB in size for upload. Create an Express JS server with file upload functionality. If a file is too big or the wrong type, reject it with a message. This keeps the server safe and efficient.	3	5
15.	You're working on a library portal that tracks user sessions. When users log in, create a session to store their name and login time. Let them see this session info on a profile page. Also, provide a logout option to destroy the session. This simulates real login/logout functionality for web apps.	3	4, 5
16.	You’re designing a freelance portfolio website. Add a contact form where users can send you their message. When they click "Submit", their message should be sent to your email ID. Use NodeMailer in Express to handle email sending. Validate entries and show a success/failure message on submission.	3	5
17.	You’re developing an admin panel for a tuition class. The admin should be able to add, view, edit, and delete student data. Use MongoDB for data storage and Express JS for routes. Each operation should be reflected in the database. Develop a simple UI or API to efficiently manage these records.	4	4, 5
18.	A mobile developer needs a backend for a notes-taking app. Create a RESTful API using Express and MongoDB to handle notes. Implement endpoints to create, read, update, and delete notes. Use Postman or similar tool to test API responses. Ensure that each note has a title, content, and timestamp.	5	4, 5