UNIFIED MENTOR

MY PROJECT REPORT

SUBMITTED BY:

J. SHRUTHI,
Internship ID: UMIP218233,
Front - end development Intern
(20/NOV/2024 - 20/DEC/2024).

PROJECT- 1:

PROJECT-NAME: A BASIC DIGITAL CALCULATOR

Technologies Used: HTML, CSS, JavaScript.

1.1 PROJECT DESCRIPTION:

The Basic Digital Calculator project is a simple front-end web-based application which is designed with a objective to perform fundamental arithmetic operations such as addition, subtraction, multiplication, and division. It is Built using HTML, CSS and JavaScript for structure, styling and functionality respectively. This project serves to help users perform everyday math with ease.

The calculator project provides a clean and intuitive user interface that allows users to easily input their required mathematical equations. The layout is responsive, ensuring accessibility and usability across various devices like smartphones, tablets, and desktops. CSS is employed to enhance the aesthetics, using modern design principles to improve user experience.

The core functionality is implemented with JavaScript, where event listeners capture user inputs and handle respective calculations. The logic ensures accurate computation while also managing edge cases, such as division by zero.

This project not only reinforces essential programming concepts such as DOM manipulation and event handling but also emphasizes best practices in code organization and user experience design. As a result, the Basic Calculator serves as both a practical tool for users and a valuable learning resource for aspiring developers.

1.2 MAIN FEATURES AND FUNCTIONALITIES:

- The main functionality of this digital calculator is that All the basic arithmetic operations like Addition, subtraction, multiplication and division can be performed using it.
- It provides the users with a clean and real-time ‘Calculation Display’.
- Also it has ‘Clear’ and ‘Reset’ options too to modify or start over with your calculations desired number of times.
- This Basic calculator also ensures ‘Error handling’ of invalid input values like ‘division by zero’.
- The error messages can be seen directly on the calculator’s display facilitating users to check their inputs and modify them in real-time.
- It also features some engaging animations to captivate the users.
- This project is implemented without usage of potentially risky functions like Eval() which might be vulnerable for malicious code injections.

Technically speaking,

- It is very ‘event-driven project’ and responds really well to all user inputs.
- It is equipped with many input validation-checks to prevent all possible calculation errors.
- It ensures cross-browser compatibility too.

- This project has a clean user interface which is easy to use and understand for any user.
- It is very reliable and accessible to perform everyday mathematical calculations on any device due to its well responsive layout.

1.3 DESIGN AND IMPLEMENTATION:

1.3.1 HTML STRUCTURE:

The HTML code structure of this project contains mainly

1. A Header with the title.
2. An input display where all the numbers and operators the user clicks on will be displayed.
3. And a buttons grid organised well which contains all the necessary inputs like numbers and operators for the calculations.
4. And all the special operators like 'AC', 'C', '=' have 'Event handlers' defined to them which starts operating upon click-events.

KEY COMPONENTS:

- A display bar
- Numbers/digits from 0-9
- Operators like +, -, *, \.
- Functional keys like AC or C.
- And a '. ' for decimal calculations.
- Finally a 'equals to' (=) button to initiate calculations.

1.3.2: CSS STYLING:

The styling of this Basic digital Calculator contains:

- Beautiful Radial-gradient background for the Calculator.
- A dark display screen with clearly visible font and font-size for Output Characters.
- Number and Operator buttons with their specific hover effects.
- Also AC and C buttons having their own respective hover effects too.
- Smooth transitions and animations including a 'welcome-animation'.
- Responsive design with media queries for different screen sizes.

1.3.3: JAVASCRIPT FUNCTIONALITY:

The JavaScript code of this project consists of four functions namely

Clear(), Delete(), Display() and Calculate().

Clear(): It is a simply executed function which Sets the display value to a empty string(‘ ’) so that the display is cleared.

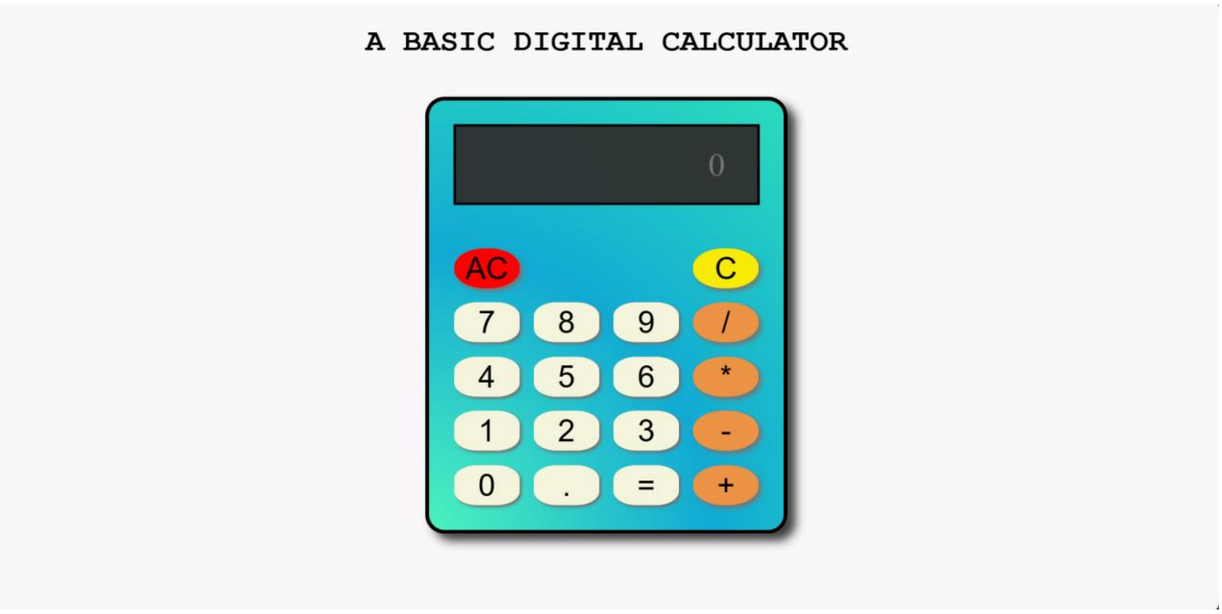
Delete(): This function is executed using slice() method which generically omits/excludes the last character and gives only the remaining characters to the display.

Display(): The display method here takes a input character assigned/asspciated to every key already in the html code. It uses Concatenation of input characters to form a input-string as the equation.

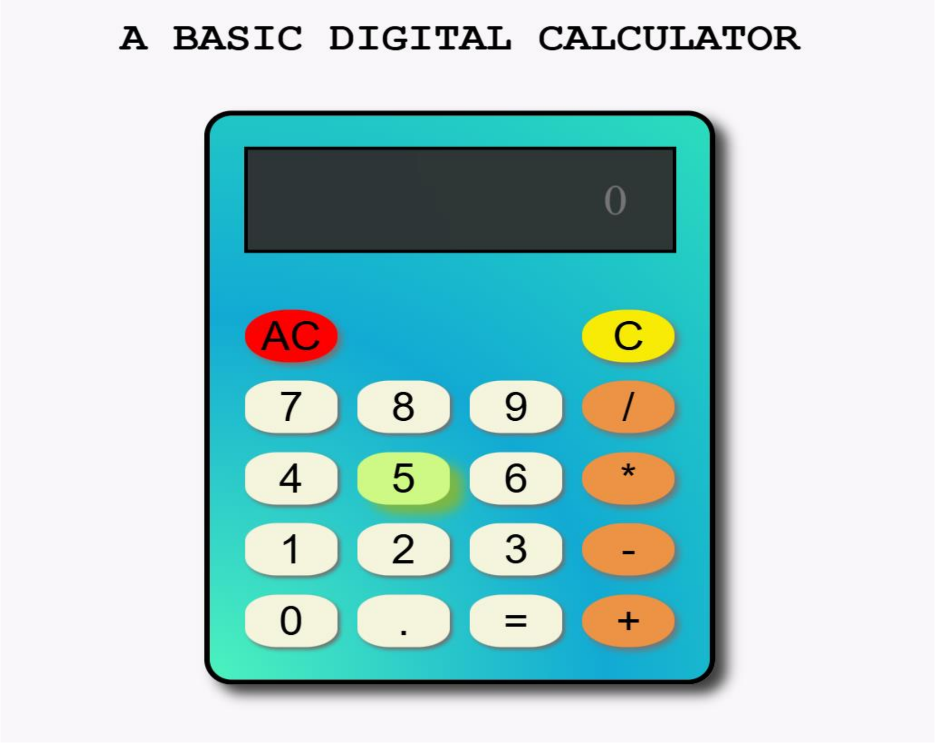
Calculate():This function is the tough part of this project which includes many steps like

- Get the input equation from display
- Split it into numbers and operators
- Perform many different input validations. Here I used the help of regular expressions to properly validate all possible inputs like to check if input ends with a digit, if it contains any repeated operators in it etc.
- Error Handling: Shows error message if input is invalid. I assigned different error messages to different error-cases as to Understand Specifically Which error has occurred like ‘Invalid input 1 or 2 or 3’ etc.
- And calculations follows BODMAS rule for operator precedence like Operators with higher precedence like * and \ are processed first and then followed by + and -.
- Lastly the result is outputted to the display of the calculator.

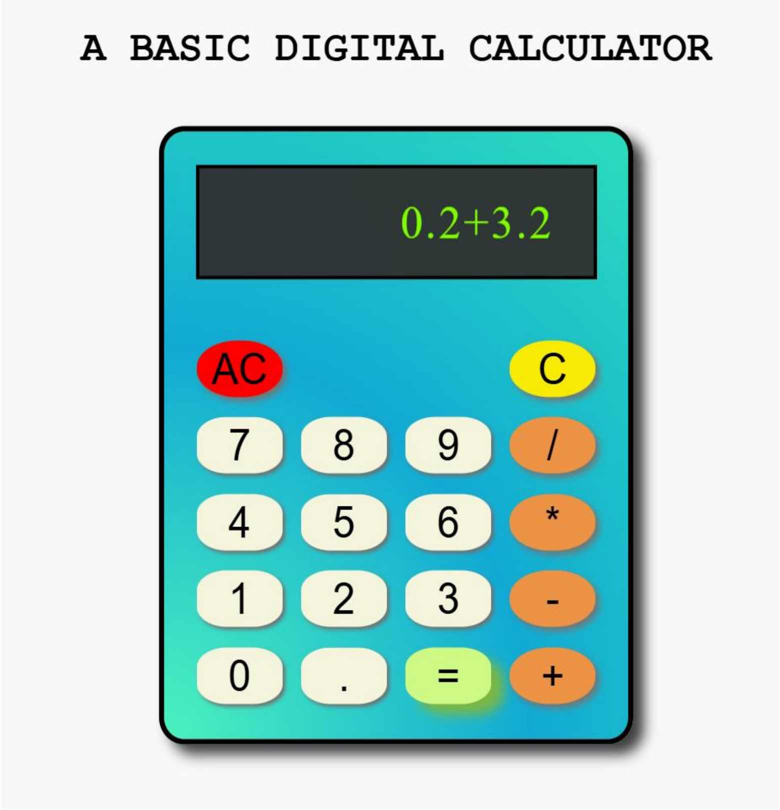
1.4: SCREENSHOTS OF THE PROJECT:



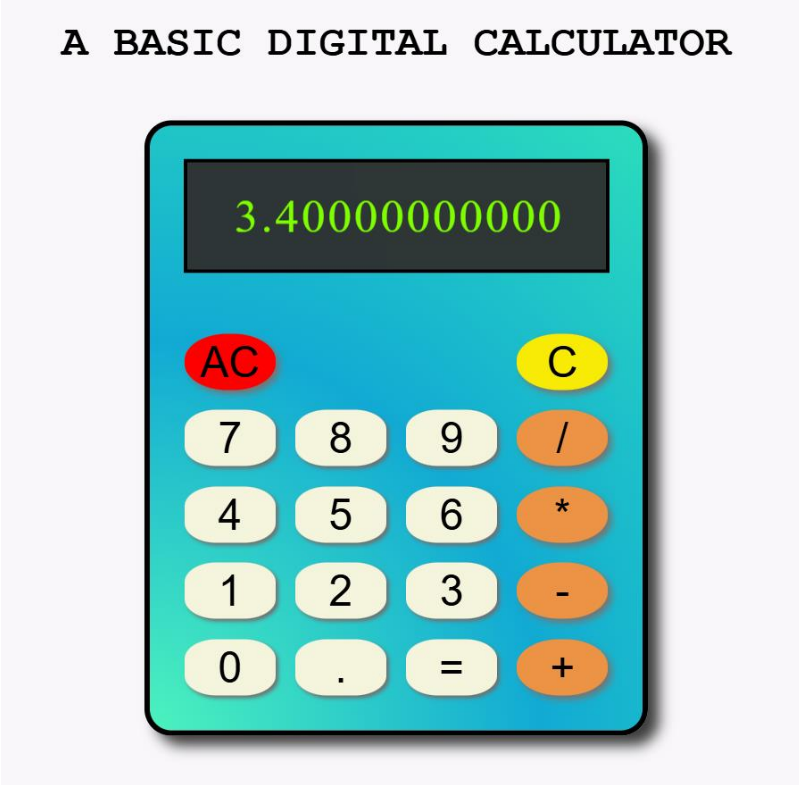
Output 1: The main page of the project.



Respective **Hover effects** upon focusing on keys.

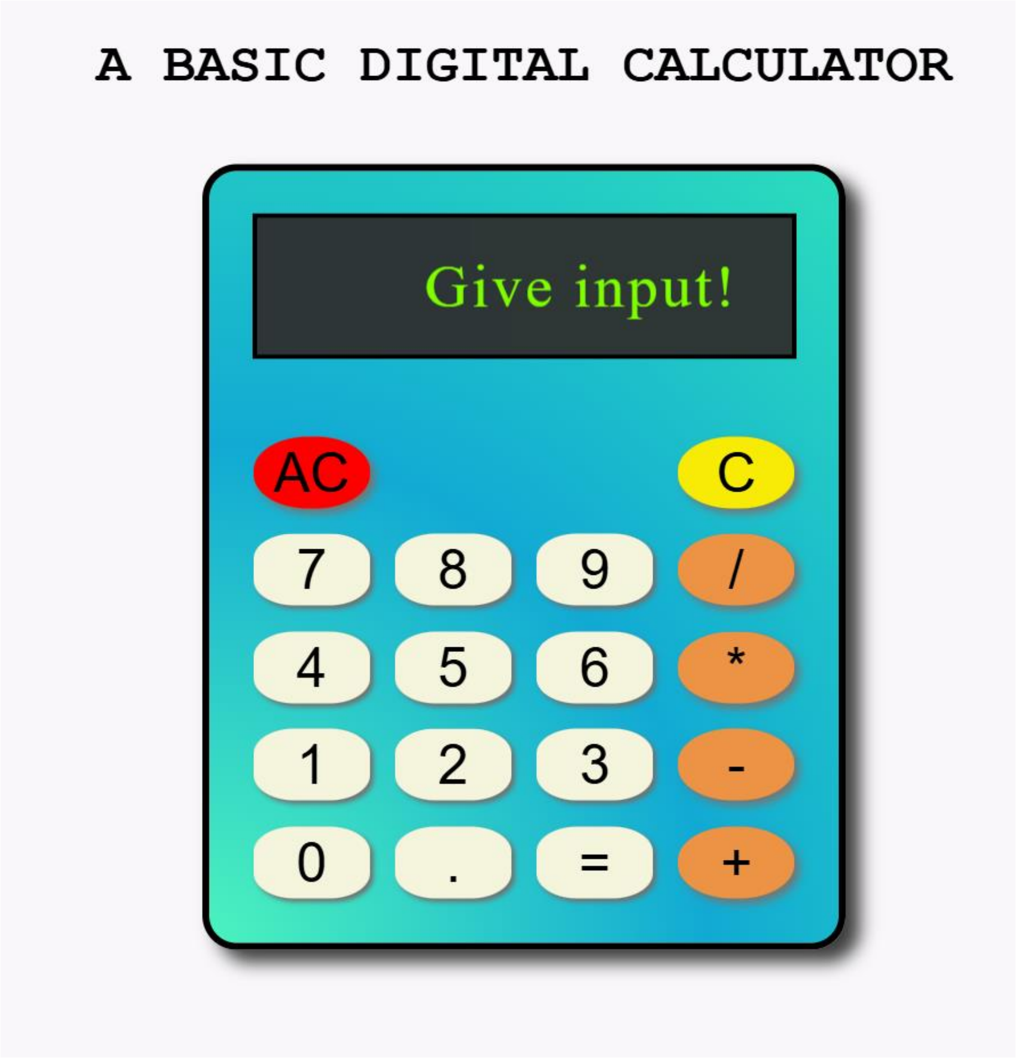


A BASIC DIGITAL CALCULATOR



A BASIC DIGITAL CALCULATOR

Perform all types of **basic arithmetic operations** like addition, subtraction, multiplication and division to see the results.

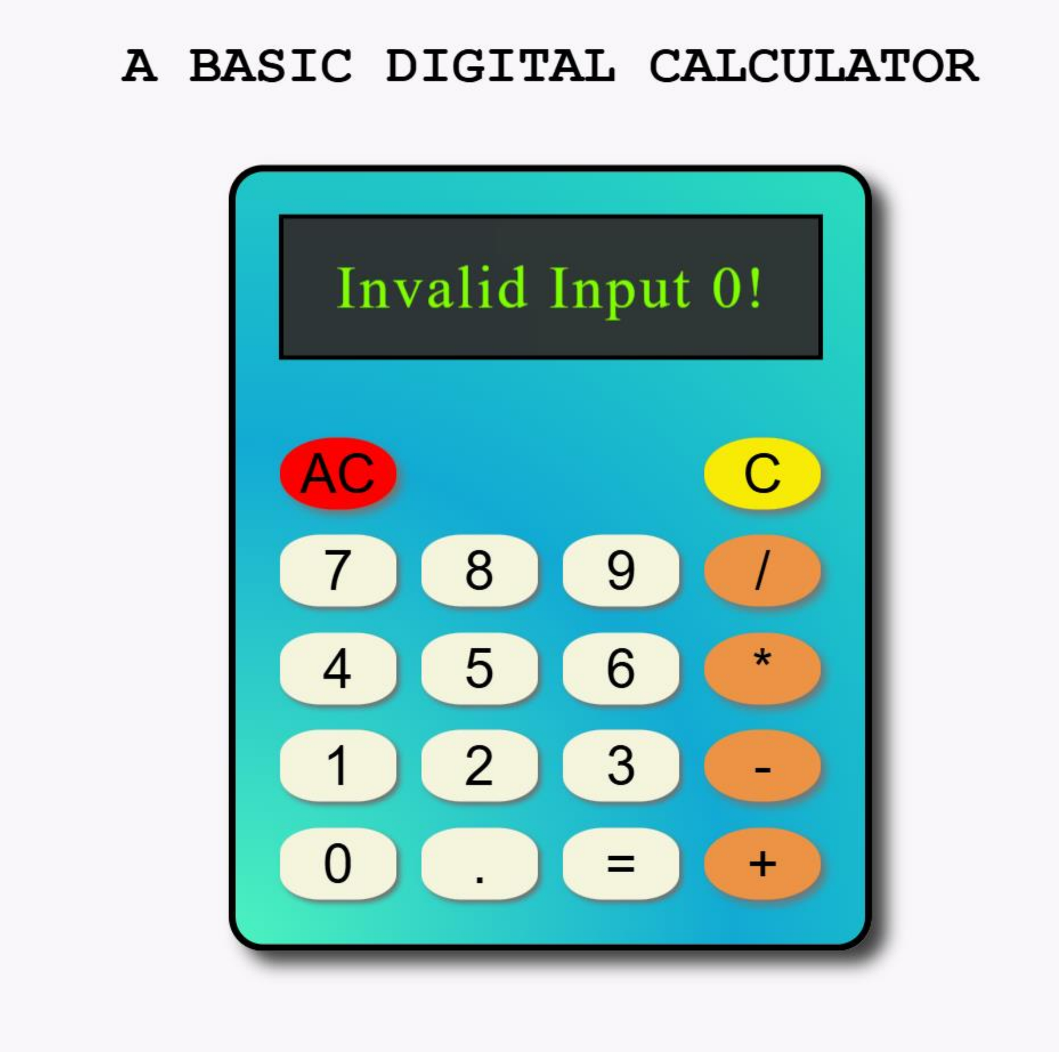


A BASIC DIGITAL CALCULATOR

Give input!

Input validations like it displays ‘Give input!’ message to the display bar

IF the user clicks on equals to(=) key without giving any input.

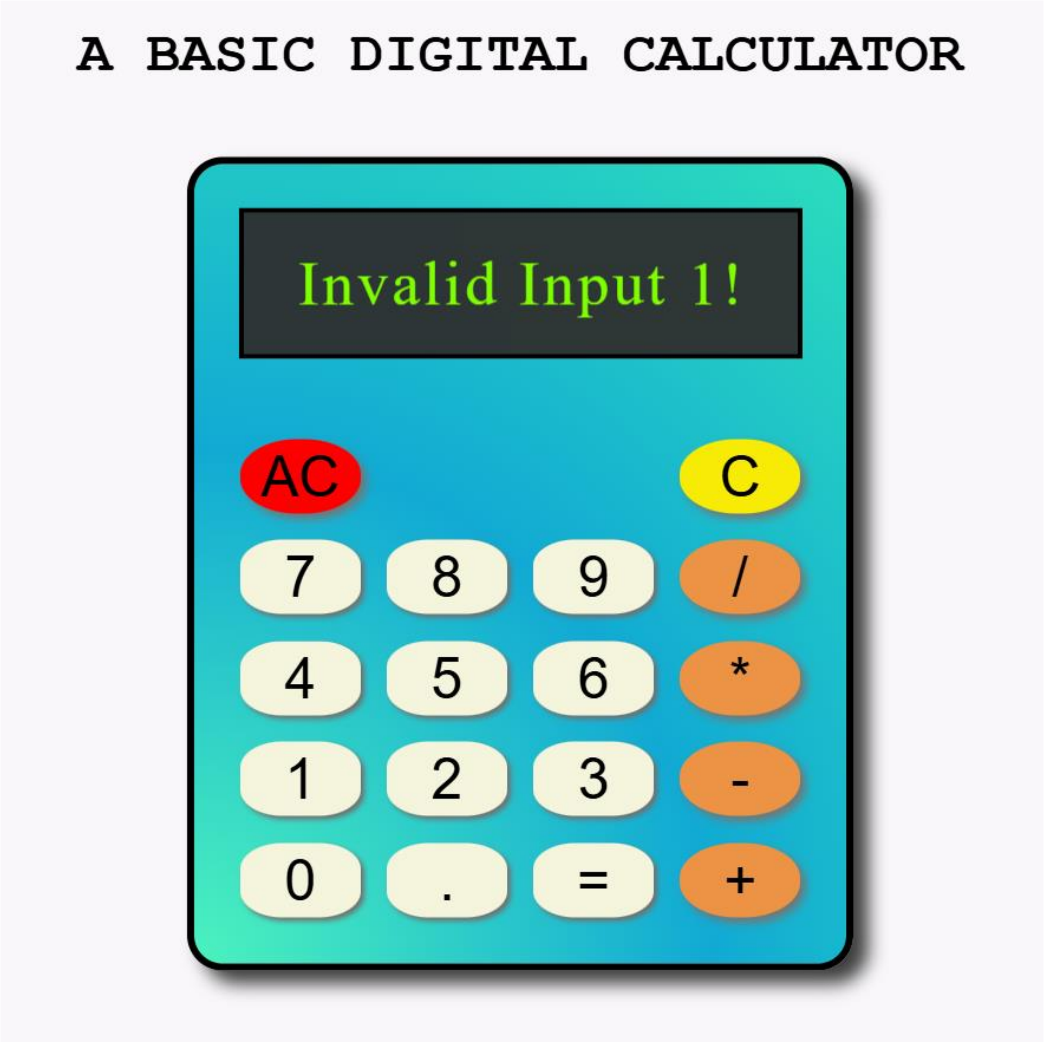


A BASIC DIGITAL CALCULATOR

Invalid Input 0!

ERROR HANDLING: It displays ERROR MESSAGES like ‘invalid input 0!’

WHEN the user tries to ‘divide any number by zero.’



A BASIC DIGITAL CALCULATOR

Invalid Input 1!

It Displays ‘Invalid input 1!’

WHEN the user doesn’t give OR forgets to give numbers or operators in the input.

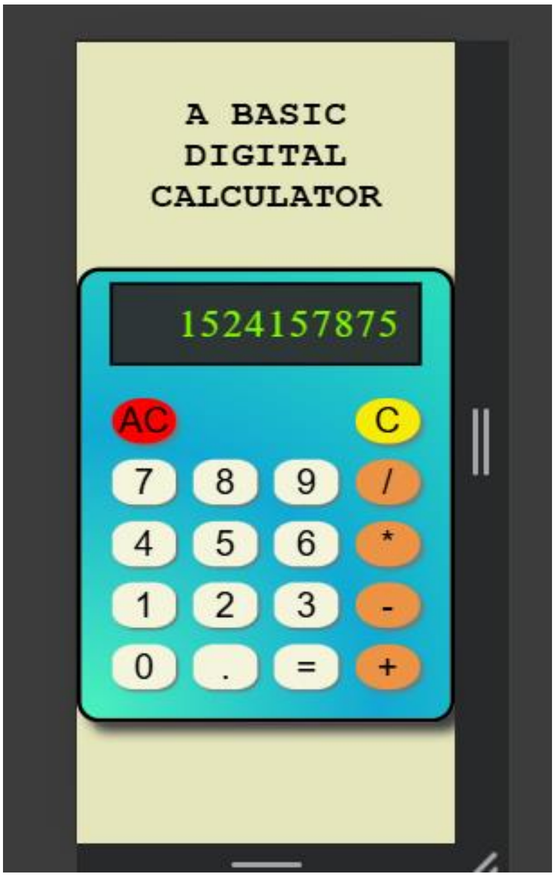
Similarly the messages ‘invalid input 3!‘

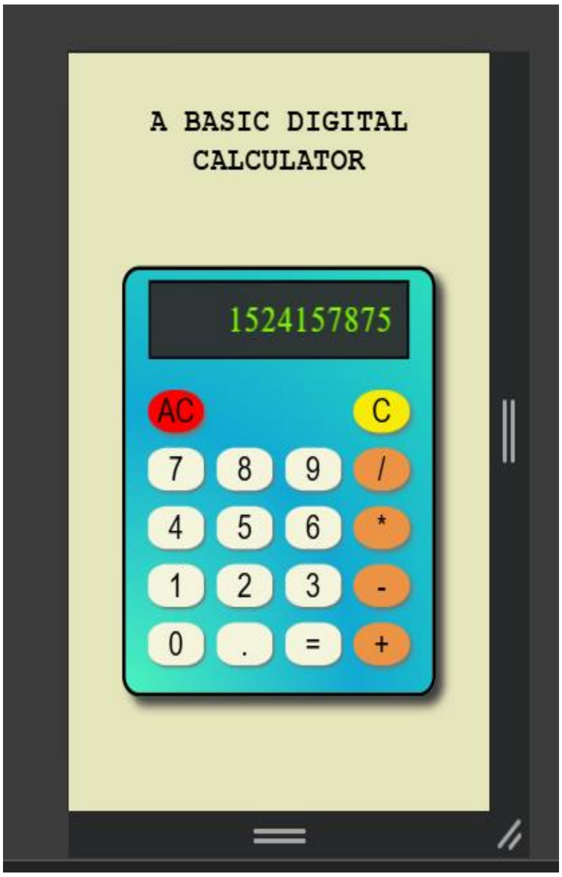
WHEN the inputted equation doesn’t end with a digit

and

The message ‘invalid input 4!‘

WHEN the input has repeated operators like 2++3, 5***7 or 2////6 etc. in it.





Given different background colors for understanding the responsiveness of the calculator code.

The calculator seamlessly adjusts to different screen-sizes including all of its components like headers, display, key-pad and the calculator itself resizes accordingly without any problems.

The best part about this code is It functions the same efficiently in all given conditions.

1.5: CONCLUSION:

The Basic Calculator project successfully implements all the fundamental arithmetic operations with a user-friendly interface and robust error handling.

1.6: REPOSITORY LINKS:

https://github.com/Yogadventure/Unified-Mentor-folder.

PROJECT- 2:

PROJECT-NAME: A DYNAMIC COUNTDOWN TIMER

Technologies Used: HTML, CSS, JavaScript.

2.1 PROJECT DESCRIPTION:

The project of a Dynamic Multiple Countdown Timer is a web-based application developed using web technologies like HTML, CSS, and JavaScript that enables users to track the remaining time until a ‘specified future date and time’. This application is capable of managing up to five countdown timers simultaneously. And this project also provides a feature of an audible alarm upon completion of each of the countdown timers.

This project provides a user-friendly interface where users can input their desired target date and time through an intuitive datetime picker.

The application dynamically calculates and displays the remaining-time broken down into days, hours, minutes and seconds, updating it every second in real-time.

The core functionality utilizes JavaScript's Date object and setInterval() method to perform precise time calculations and maintain accurate countdown updates.

The user interface is enhanced through modern CSS styling techniques, featuring responsive design principles and visual feedback elements.

The application's structure follows a modular approach with clear separation between markup (HTML), presentation (CSS), and functionality (JavaScript) respectively.

That way this project serves as a practical implementation of web development fundamentals while providing a useful tool for users to keep the track of upcoming events or deadlines.

2.2 MAIN FEATURES AND FUNCTIONALITIES:

The project’s Key features include:

- User-friendly input interface.
- Dynamic content sections.
- Animations like rotating timer icon, welcome-glow-effect, hover effects etc to capture and engage with the users.
- Built-in datetime picker for selecting target date and time.
- Input validation to ensure proper date selection.
- Start button to initiate countdown.
- Automatic updates at one-second intervals.
- Real-time countdown timer display in multiple time units.
- Accurate time tracking using JavaScript's Date object.
- Leading zero padding for consistent display format
- Clear error handling and user feedback.
- Completion notification when the countdown reaches zero.
- Also rings a alarm upon completion of the countdown timer.

Technically speaking, the project is a Time Management Web Application designed with front-end technologies like HTML, CSS AND JavaScript which facilitates the user to set countdown timers to their desired times and get notified upon completion helping users to keep track of forthcoming events without any issues.

2.3 THE DESIGN AND IMPLEMENTATION:

2.3.1 THE HTML STRUCTURE:

The HTML code structure of this project mainly contains:

- A header featuring title ‘Dynamic Countdown Timers’.
- A hour-glass icon from font-awesome page for visual representation.
- A paragraph description explaining the purpose of countdown timers
- A timer section containing input element for collecting user input, major functions like createTimers(), startTimers() which gets triggered upon their respective button-clicks.
- And other div sections to display input date-times and the remaining countdown times.

KEY COMPONENTS:

- Multiple timer support (up to 5 simultaneous timers).
- Dynamic timer creation based on user input.
- Audio notification system i.e alarm upon completion along with completion message.
- User-friendly interface with clear instructions.
- Centralized control for managing multiple timers.

2.3.2 THE CSS STYLING:

- Rotating hourglass icon animation.
- Box shadows for div container elements for depth.
- Glow animations upon hover for input and button elements.
- Custom Date-time-picker-indicator styling.
- Smooth transitions for interactive elements.
- Media queries for screens below 600px.

2.3.3 THE JAVASCRIPT FUNCTIONALITY:

The JavaScript code of this project contains mainly three functions like CreateTimers(), StartTimers() and StartTimer1() along with a nested updateTimer() in it.

createTimers() -   This function starts by clearing the timeInputs div container. It dynamically creates the Date-time input fields the DOM manipulation way, according to the user input of desired number of timers. It then generates ‘ label based input fields’ as per the specified count and ids , to the output page.

StartTimers() - This function starts by clearing any previous active intervals. It then creates a displayContainer which holds the different timers, which will be displayed on to the screen. Then it initiates all the individual ‘created timers’ simultaneously using ‘for loop’ for iterations considering the number of desired timers entered by a user. Lastly It performs input Validations for each timer displaying respective alert messages for users so as to facilitate them to check their inputs clearly upon any mistake. Then it calls a StartTimer1() to actually start the timer’s countdown and so on.

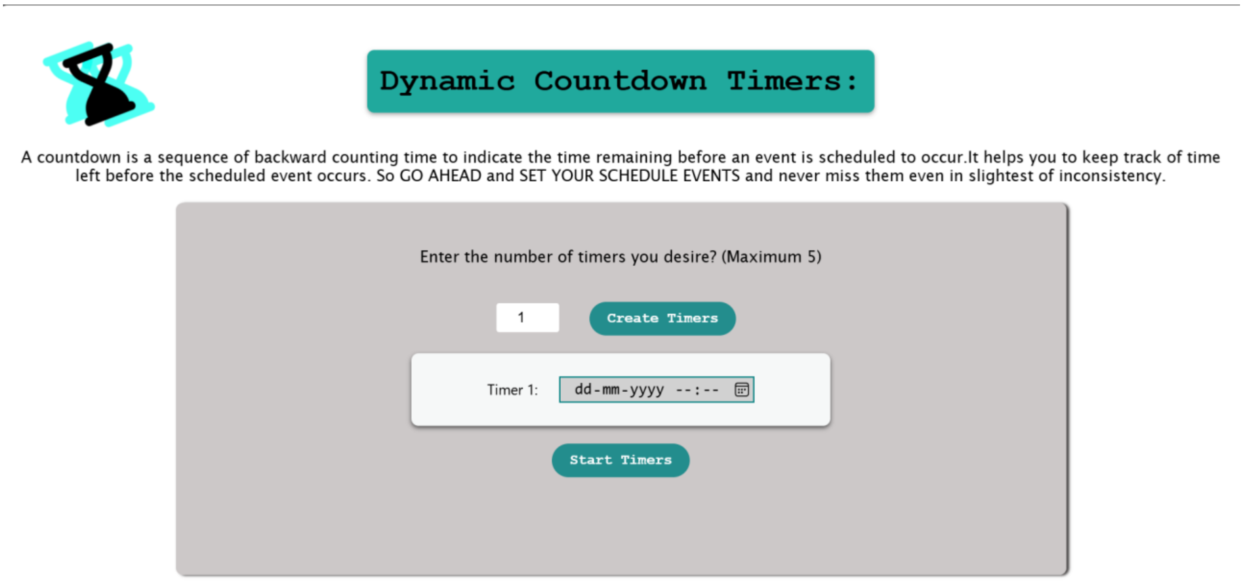
StartTimer1() – This function takes two arguments like TimerID and targetTime at the initialisation. This function is generally defined to handle the individual timer logic literally starting the individual timers and updating it continuously to display the proper countdown logic in action which is done inturn by calling the nested updateTimer() function in it.

updateTimer() – This function calculates the remaining time

(like timeLeft = targetTime – Date().now) breaking it down to Days, Hours, Minutes and seconds accordingly, using the Date() object and Math functions respectively. It even gives the padded input to two digits to all the components for a clear and concise time in the output display. All while updating the display every second using the setInterval().

Eventually upon completion of the respective countdown timers it displays a completion message too along with an alarm ringing to get users attention if busy.

2.4 SCREENSHOTS OF THE PROJECT:



Dynamic Countdown Timers:

A countdown is a sequence of backward counting time to indicate the time remaining before an event is scheduled to occur.It helps you to keep track of time left before the scheduled event occurs. So GO AHEAD and SET YOUR SCHEDULE EVENTS and never miss them even in slightest of inconsistency.

Enter the number of timers you desire? (Maximum 5)

1

Timer 1: dd-mm-yyyy --:--

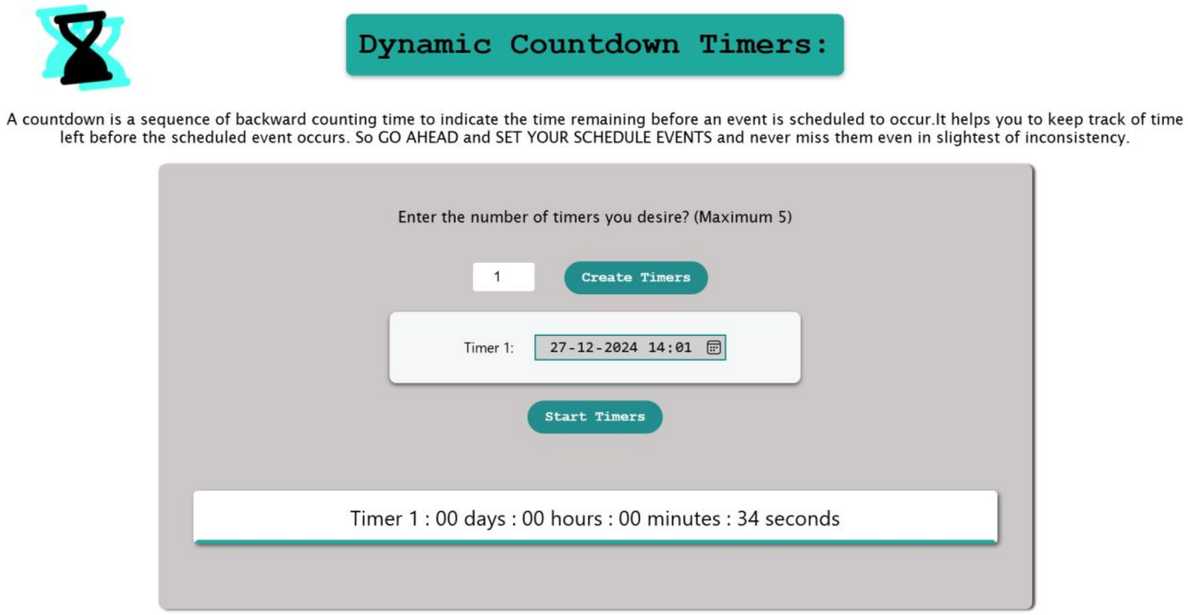
The main output page of the project.

It contains a rotating hourglass icon animation on the top-left corner.

The button elements have a glow effect applied to them which initiates right upon the page loading.

It also contains a input element for the user to specify the number of countdown timers they need.

Upon choosing, the user have to click on the create timers button which will give them the option to choose their required dates and times using a date-time-picker by clicking on its indicator with a hover effect applied on it.



Dynamic Countdown Timers:

A countdown is a sequence of backward counting time to indicate the time remaining before an event is scheduled to occur.It helps you to keep track of time left before the scheduled event occurs. So GO AHEAD and SET YOUR SCHEDULE EVENTS and never miss them even in slightest of inconsistency.

Enter the number of timers you desire? (Maximum 5)

1

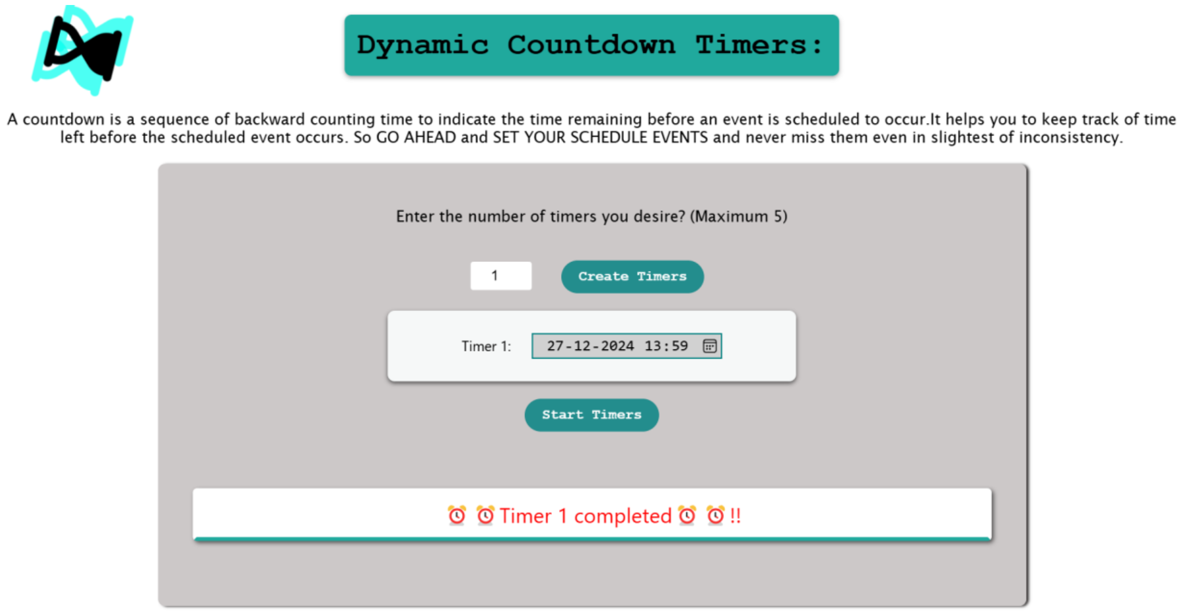
Timer 1: 27-12-2024 14:01

Timer 1 : 00 days : 00 hours : 00 minutes : 34 seconds

After choosing the dates and times,

the user have to press the Start Timers button which will display the countdown timer/s at the bottom of the container displaying clearly the remaining time in Days, Hours, Minutes and seconds

It will automatically update the values every second and gives it to the display so as to facilitate the user to know the time left before the timer ends.



Dynamic Countdown Timers:

A countdown is a sequence of backward counting time to indicate the time remaining before an event is scheduled to occur.It helps you to keep track of time left before the scheduled event occurs. So GO AHEAD and SET YOUR SCHEDULE EVENTS and never miss them even in slightest of inconsistency.

Enter the number of timers you desire? (Maximum 5)

1 Create Timers

Timer 1: 27-12-2024 13:59

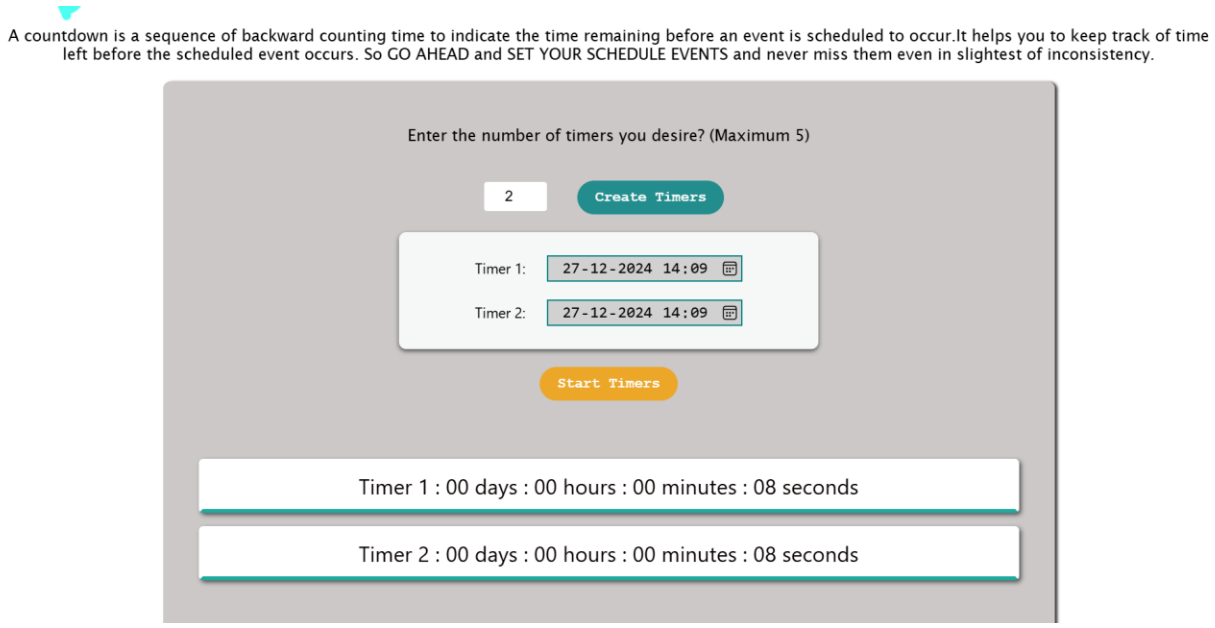
Start Timers

⏰ ⏰ Timer 1 completed ⏰ ⏰ !!

Upon completion of the countdown,

It displays a message in red color as shown in above output.

And even triggers a alarm tone which was programmed to play when the timer ends.



A countdown is a sequence of backward counting time to indicate the time remaining before an event is scheduled to occur.It helps you to keep track of time left before the scheduled event occurs. So GO AHEAD and SET YOUR SCHEDULE EVENTS and never miss them even in slightest of inconsistency.

Enter the number of timers you desire? (Maximum 5)

2

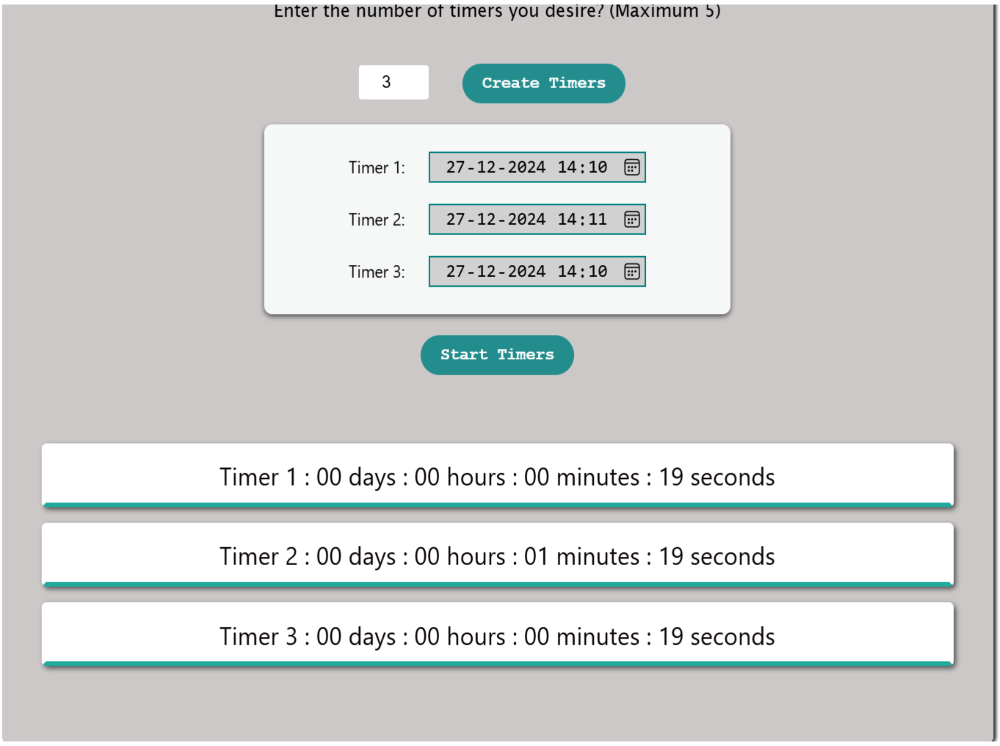
Timer 1: 27-12-2024 14:09

Timer 2: 27-12-2024 14:09

Timer 1 : 00 days : 00 hours : 00 minutes : 08 seconds

Timer 2 : 00 days : 00 hours : 00 minutes : 08 seconds

Similarly it works for even two timers or three too.



Enter the number of timers you desire? (Maximum 5)

3

Timer 1: 27-12-2024 14:10

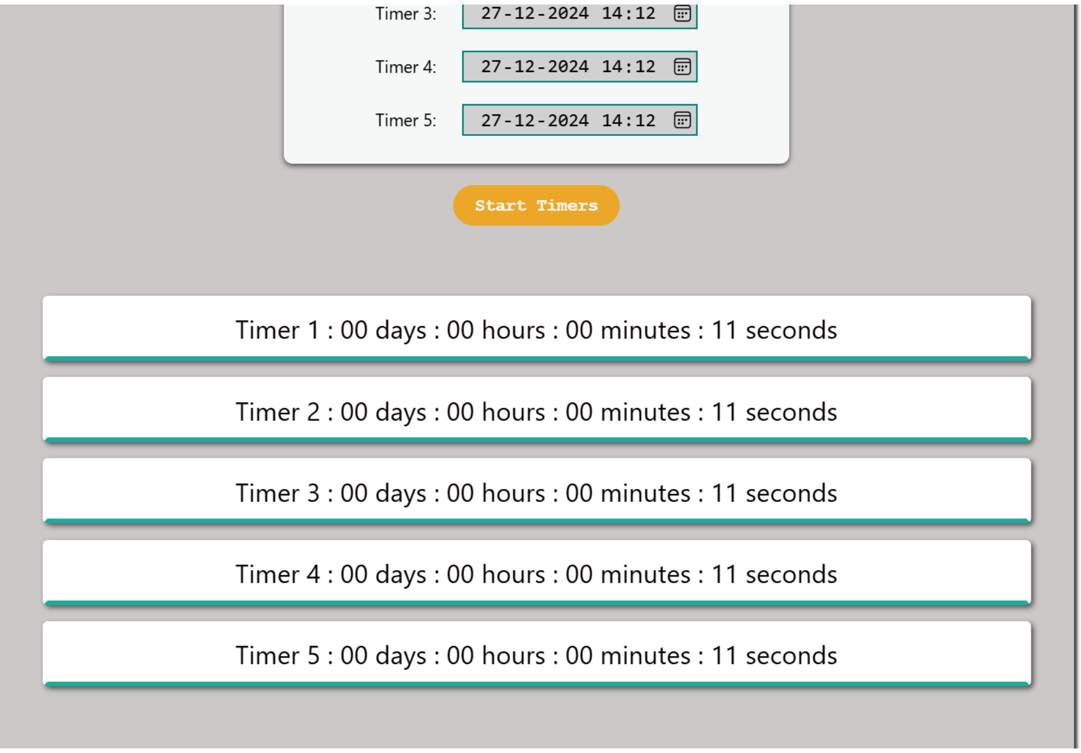
Timer 2: 27-12-2024 14:11

Timer 3: 27-12-2024 14:10

Timer 1 : 00 days : 00 hours : 00 minutes : 19 seconds

Timer 2 : 00 days : 00 hours : 01 minutes : 19 seconds

Timer 3 : 00 days : 00 hours : 00 minutes : 19 seconds



Timer 3: 27-12-2024 14:12

Timer 4: 27-12-2024 14:12

Timer 5: 27-12-2024 14:12

Timer 1 : 00 days : 00 hours : 00 minutes : 11 seconds

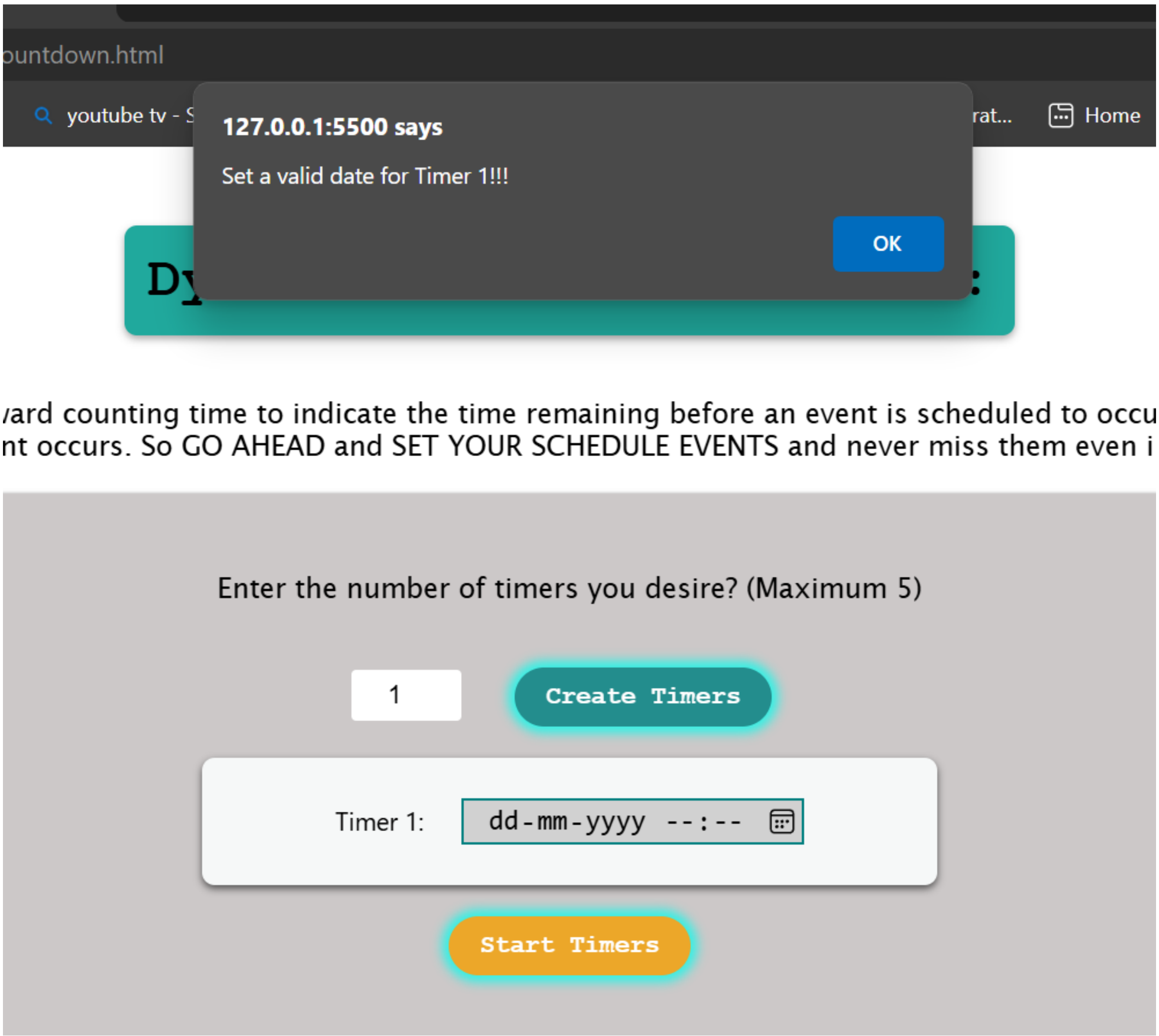
Timer 2 : 00 days : 00 hours : 00 minutes : 11 seconds

Timer 3 : 00 days : 00 hours : 00 minutes : 11 seconds

Timer 4 : 00 days : 00 hours : 00 minutes : 11 seconds

Timer 5 : 00 days : 00 hours : 00 minutes : 11 seconds

This is the output for 5 consecutive timers working simultaneously, example purpose.



127.0.0.1:5500 says

Set a valid date for Timer 1!!!

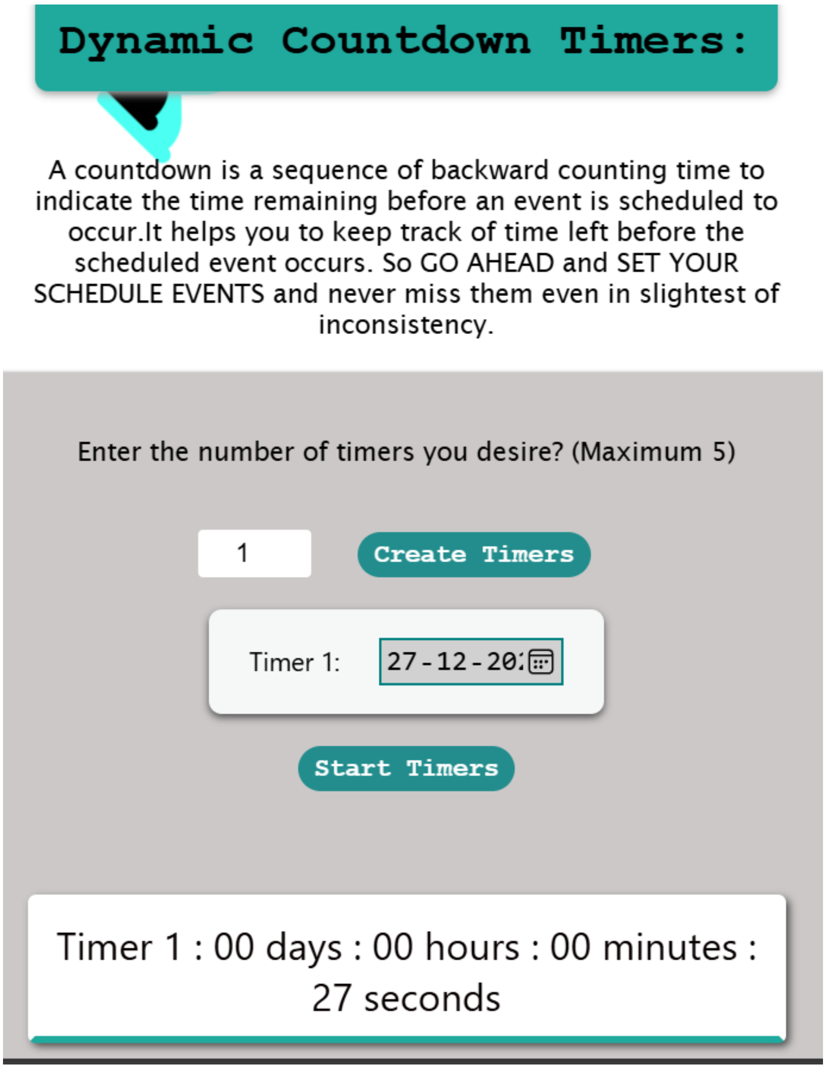
ıard counting time to indicate the time remaining before an event is scheduled to occu
nt occurs. So GO AHEAD and SET YOUR SCHEDULE EVENTS and never miss them even i

Enter the number of timers you desire? (Maximum 5)

1

Timer 1: dd-mm-yyyy --:--

ERROR HANDLING by providing alerts to the users when they type invalid values or give no value either too.



Dynamic Countdown Timers:

A countdown is a sequence of backward counting time to indicate the time remaining before an event is scheduled to occur.It helps you to keep track of time left before the scheduled event occurs. So GO AHEAD and SET YOUR SCHEDULE EVENTS and never miss them even in slightest of inconsistency.

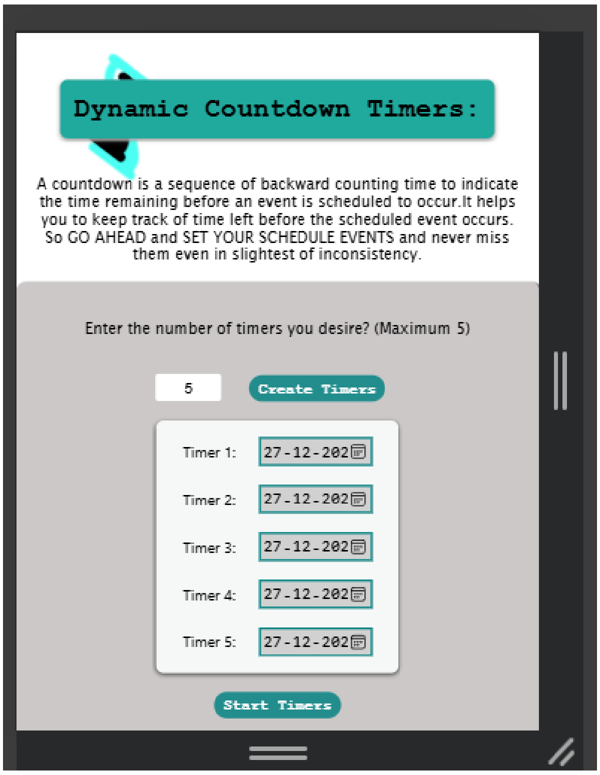
Enter the number of timers you desire? (Maximum 5)

1

Timer 1: 27-12-20

Timer 1 : 00 days : 00 hours : 00 minutes : 27 seconds

Responsive layout too to ensure it can run on different screen sizes.



Dynamic Countdown Timers:

A countdown is a sequence of backward counting time to indicate the time remaining before an event is scheduled to occur.It helps you to keep track of time left before the scheduled event occurs. So GO AHEAD and SET YOUR SCHEDULE EVENTS and never miss them even in slightest of inconsistency.

Enter the number of timers you desire? (Maximum 5)

5

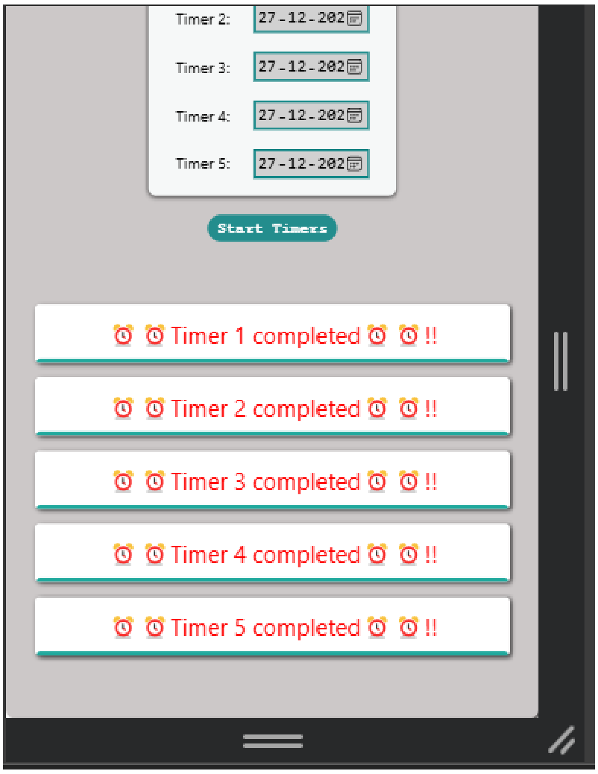
Timer 1: 27-12-202

Timer 2: 27-12-202

Timer 3: 27-12-202

Timer 4: 27-12-202

Timer 5: 27-12-202



Timer 2: 27-12-202

Timer 3: 27-12-202

Timer 4: 27-12-202

Timer 5: 27-12-202

⏰ ⏰ Timer 1 completed ⏰ ⏰ !!

⏰ ⏰ Timer 2 completed ⏰ ⏰ !!

⏰ ⏰ Timer 3 completed ⏰ ⏰ !!

⏰ ⏰ Timer 4 completed ⏰ ⏰ !!

⏰ ⏰ Timer 5 completed ⏰ ⏰ !!

This is the output on smaller screens !!!

Which shrinks seamlessly without omitting out any details of the functionality.

2.5 THE CONCLUSION:

The Dynamic Countdown Timer project successfully runs with multiple countdown timers feature and it effortlessly displays the countdown time in real time with a very user-friendly interface and robust error handling.

2.6: REPOSITORY LINKS:

https://github.com/Yogadventure/Unified-Mentor-folder.