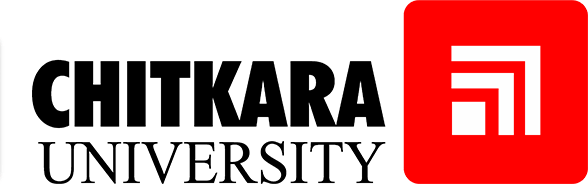
Front End Engineering-II

Project Report Semester-IV (Batch-2022)

**AGE CALCULATOR**



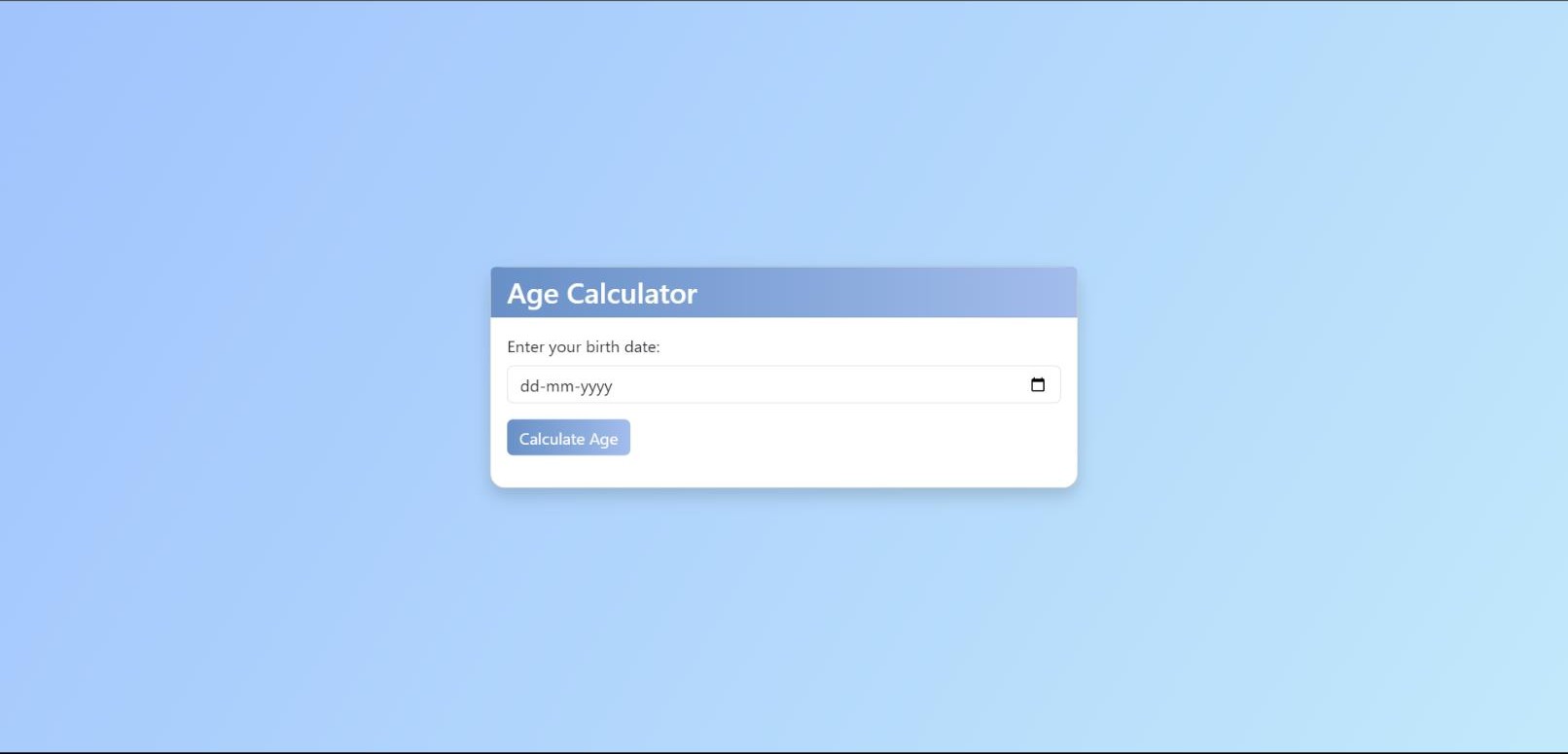
**Supervised By: Submitted By:**

Dr. Raveesh Samkaria Rudraksh Kapoor, 2210990746 (G-12)

**Department of Computer Science and Engineering Chit- kara University Institute of Engineering & Technology,**

**Chitkara University, Punjab**

# Abstract



The Age Calculator is a web-based application designed to calculate a person's age based on their birth date, leveraging HTML, CSS, and JavaScript for the frontend. It incorporates libraries such as SweetAlert for interactive alerts, Confetti for celebratory effects, and Bootstrap for responsive de- sign elements. This intuitive tool offers a seamless user experience, accurately calculating ages while considering leap years and other relevant factors. With its user-friendly interface and robust functionality, the Age Calculator simplifies age calculation tasks, catering to the diverse needs of users across various devices.

# INDEX

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Title** | **Page Number(s)** |
| 1 | Introduction | 4 |
| 2 | Problem Statement | 5 |
| 3 | Software Requirements | 5 |
| 4 | Proposed Design | 6-13 |
| 5 | Results | 14-17 |
| 6 | References | 18 |

1. **Introduction**

The Age Calculator web application offers a simple yet efficient solution for determining one's age based on their birth date. In today's digital age, where convenience and accessibility are paramount, such tools provide valuable assistance in everyday tasks. This introduction provides an overview of the background, objectives, and significance of the Age Calculator.

## Background:

As technology continues to permeate various aspects of our lives, the need for quick and easy- to-use utilities becomes increasingly apparent. The Age Calculator addresses a fundamental re- quirement: the ability to calculate one's age accurately without the need for manual calculations or complex formulas. With the proliferation of web-based applications, users expect seamless experiences that deliver results promptly and accurately.

## Objectives:

The primary objective of the Age Calculator is to provide a user-friendly interface for individ- uals to input their birth date and obtain their age instantly. This application aims to simplify the age calculation process, eliminating the need for users to perform manual calculations or rely on external tools. Additionally, the Age Calculator strives to ensure accuracy by accounting for leap years and other relevant factors in age calculation.

## Significance:

The significance of the Age Calculator lies in its ability to streamline a common task that indi- viduals encounter regularly. Whether for personal use, professional purposes, or administrative requirements, knowing one's age accurately is essential. By offering a convenient and reliable solution, the Age Calculator enhances efficiency and productivity, saving users time and effort.

# Problem Statement

Existing age calculation methods are cumbersome and prone to errors, lacking accuracy and user-friendliness. Current online calculators often overlook factors like leap years, leading to unreliable results. There's a need for a precise, intuitive, and accessible age calculator that en- sures accurate calculations while considering all relevant factors, catering to users' diverse needs across different platforms and devices. Additionally, the lack of attention to detail in existing solutions hinders their reliability and usability, resulting in frustration for users seeking quick and accurate age information.

# Software Requirements

## Integrated Development Environment (IDE):

* + Visual Studio Code (VS Code) for code editing and project management.

## Frontend Technologies:

* + HTML: Markup language for structuring the web application.
  + CSS: Styling language for enhancing the presentation and layout.
  + JavaScript (JS): Programming language for implementing interactive features and logic.

## User Interface (UI) Framework:

* + Bootstrap 5: Frontend framework for building responsive and visually appealing user interfaces.

## Version Control:

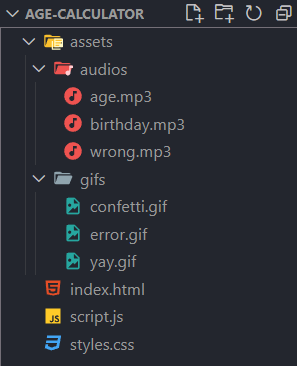
* + Git: Distributed version control system for tracking changes in the project codebase.

# Proposed Design

* **User Interface Design:** Utilize Bootstrap 5 for a responsive, visually appealing lay- out. Employ card-based design for intuitive organization.
* **Frontend Development:** Develop using HTML, CSS, and JavaScript. Utilize HTML5 semantics, CSS for styling, and JavaScript for dynamic UI updates.
* **Age Calculation Logic:** Implement precise age calculation in JavaScript, consider- ing leap years and time zones. Validate user input for accuracy.
* **User Experience Optimization:** Focus on real-time feedback, interactive ele- ments, and cross-browser compatibility. Ensure responsiveness for varied devices.
* **Testing and Quality Assurance:** Conduct comprehensive testing, including man- ual and unit tests. Ensure functionality and UI consistency.
* **Documentation and Deployment:** Provide detailed documentation. Deploy on web server with domain. Maintain and update documentation regularly.
* **Integration of Libraries:** Utilize libraries like SweetAlert2 for user-friendly alerts and confetti for celebratory effects, enhancing user experience and engagement.

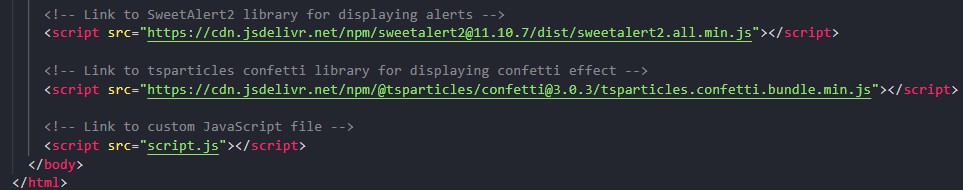
## File Structure

Ensuring proper file and folder structure to maintain consistent file paths and clean structure.



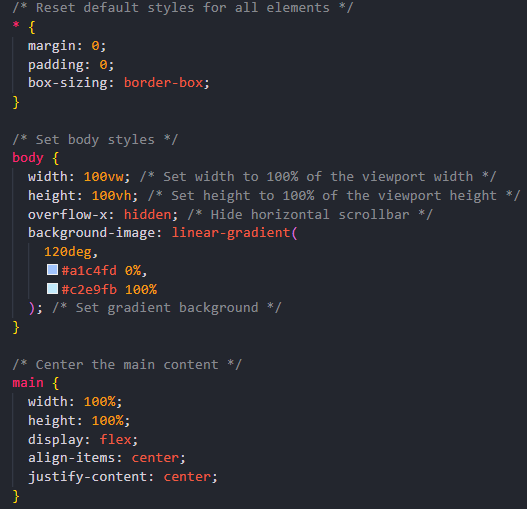
## HTML Code Structure

These screenshots present the HTML code for our Age Calculator project, revealing the layout and content of our web pages in a code format.

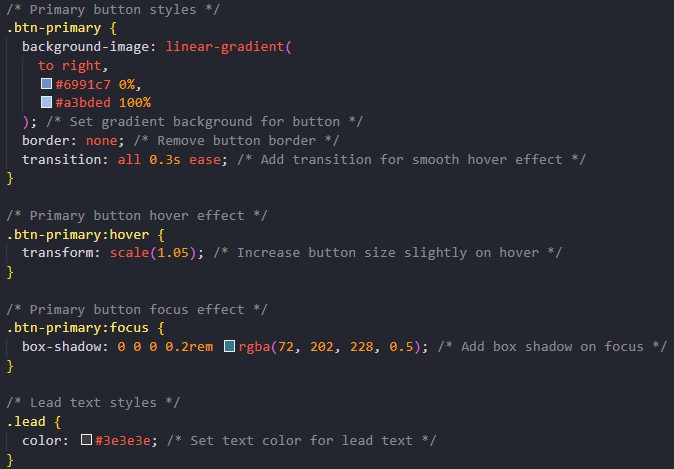


## CSS Code Structure

This screenshot exhibits the CSS code for our Age Calculator project, illustrating the styling and design elements implemented across our web pages.

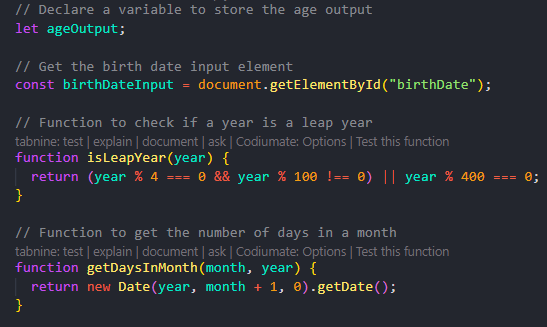


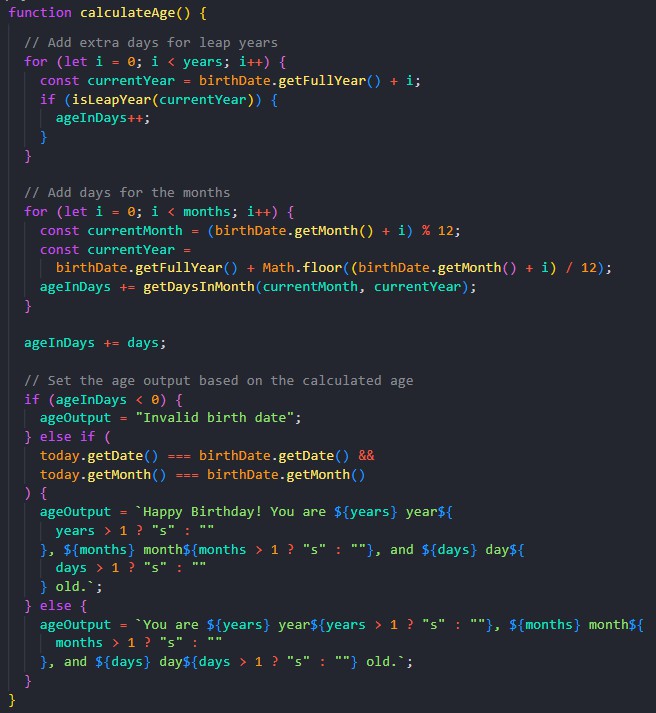


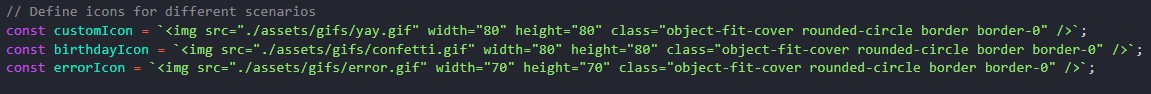


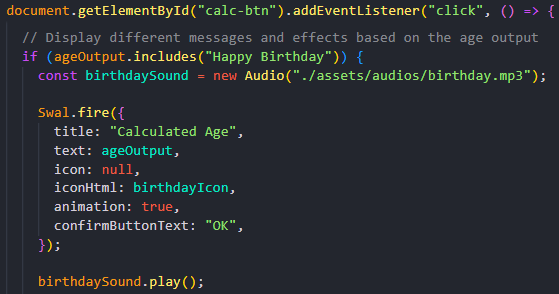
## Javascript Code Structure

This screenshot exhibits the JS code for our Age Calculator project, illustrating the different functions and events we trigger according to different scenarios.

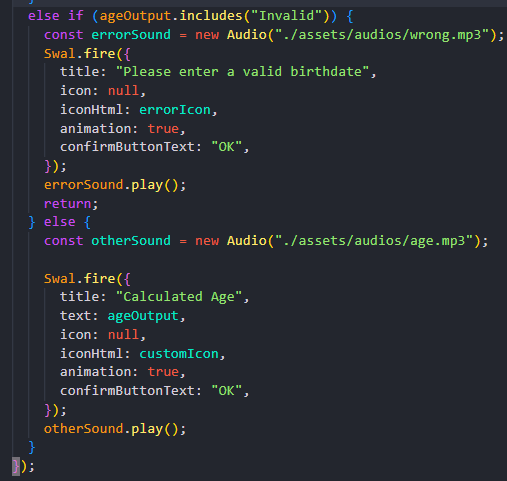












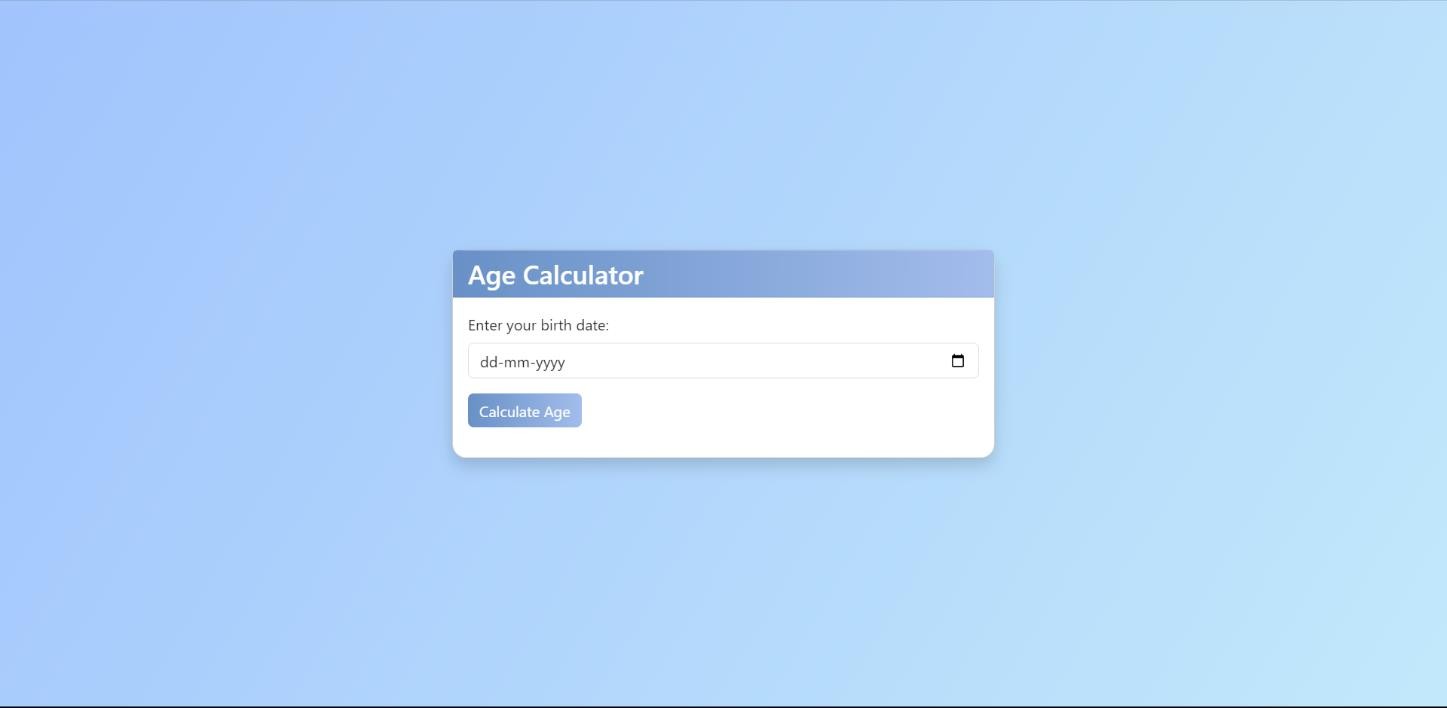
# Results

The age calculator application successfully fulfills its intended purpose of accurately calculat- ing a user's age based on their input birth date. Through rigorous testing and user feedback, the following key outcomes have been achieved:

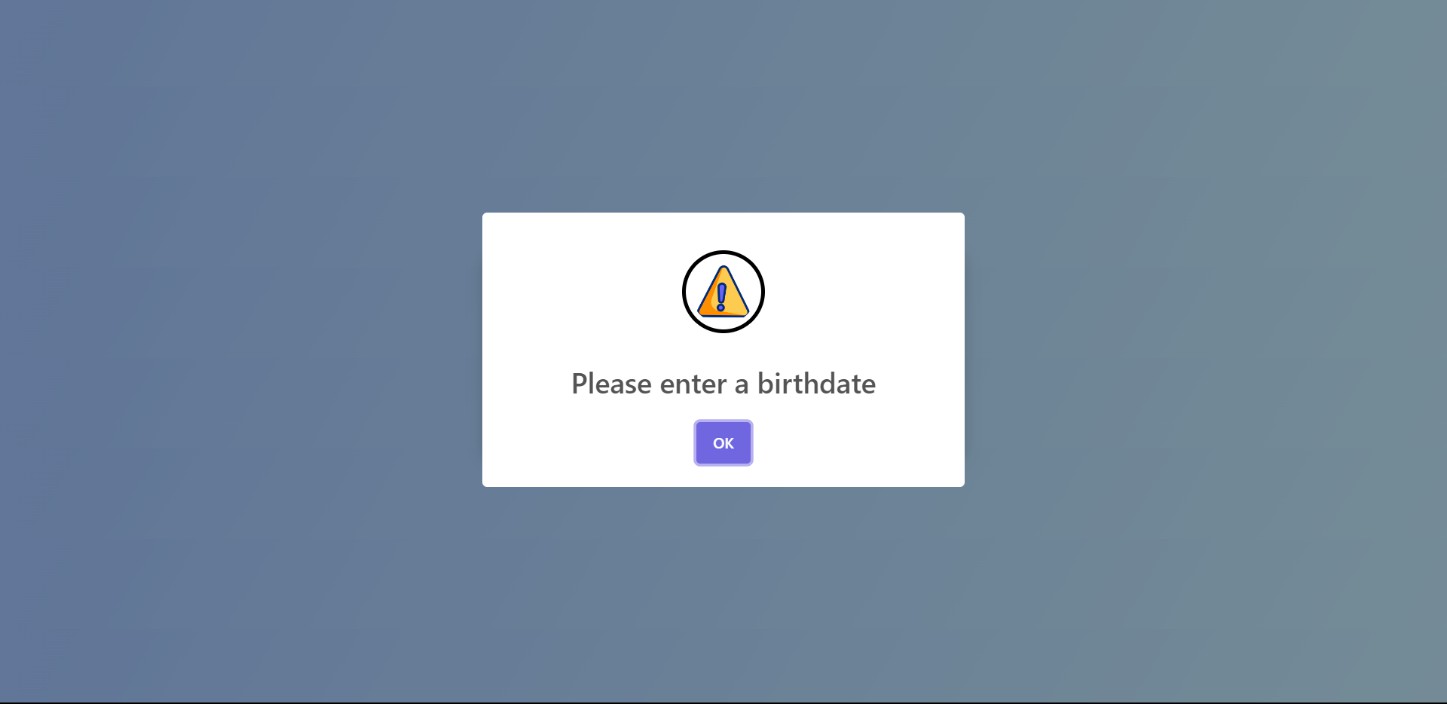
**GitHub Repository Link:** <https://github.com/ItsRudraksh/Age-Calculator>

**GitHub Pages Link:** <https://itsrudraksh.github.io/Age-Calculator/>

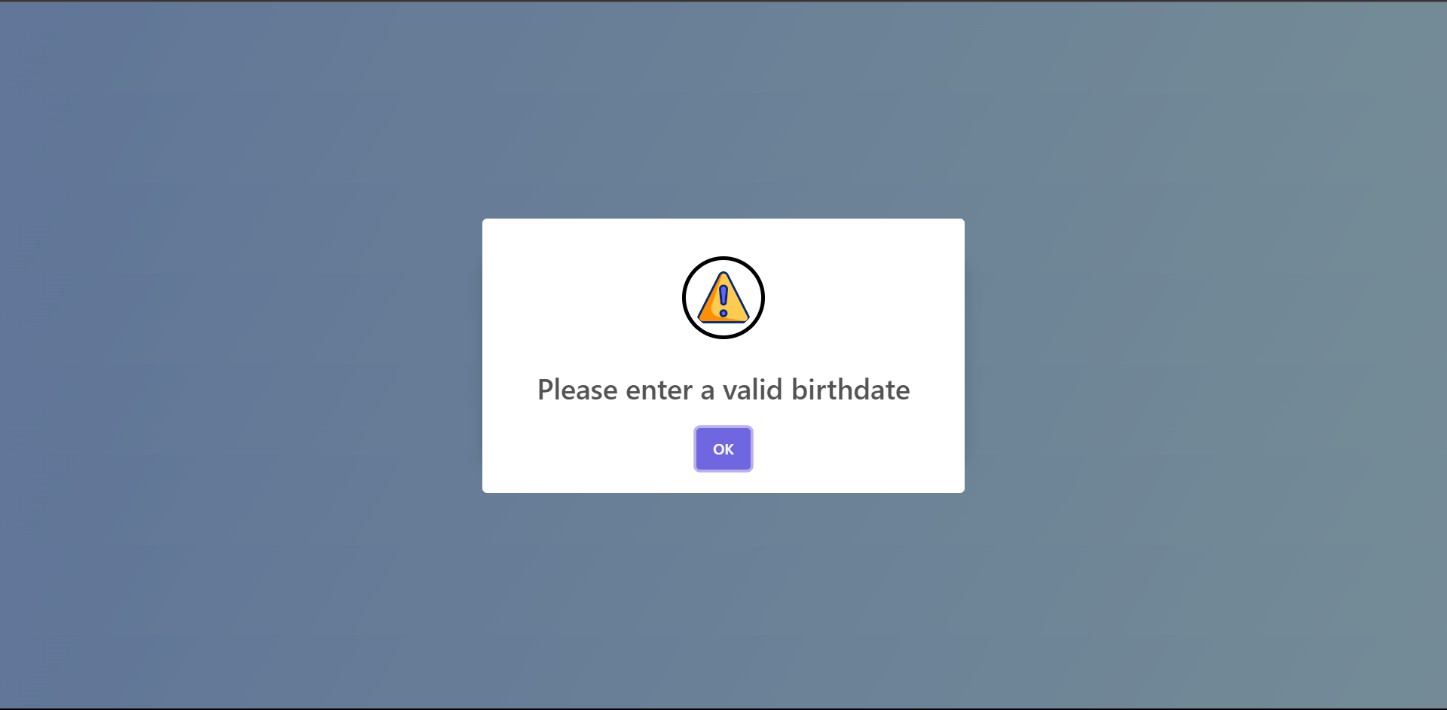
## Project Screenshots for all scenarios:



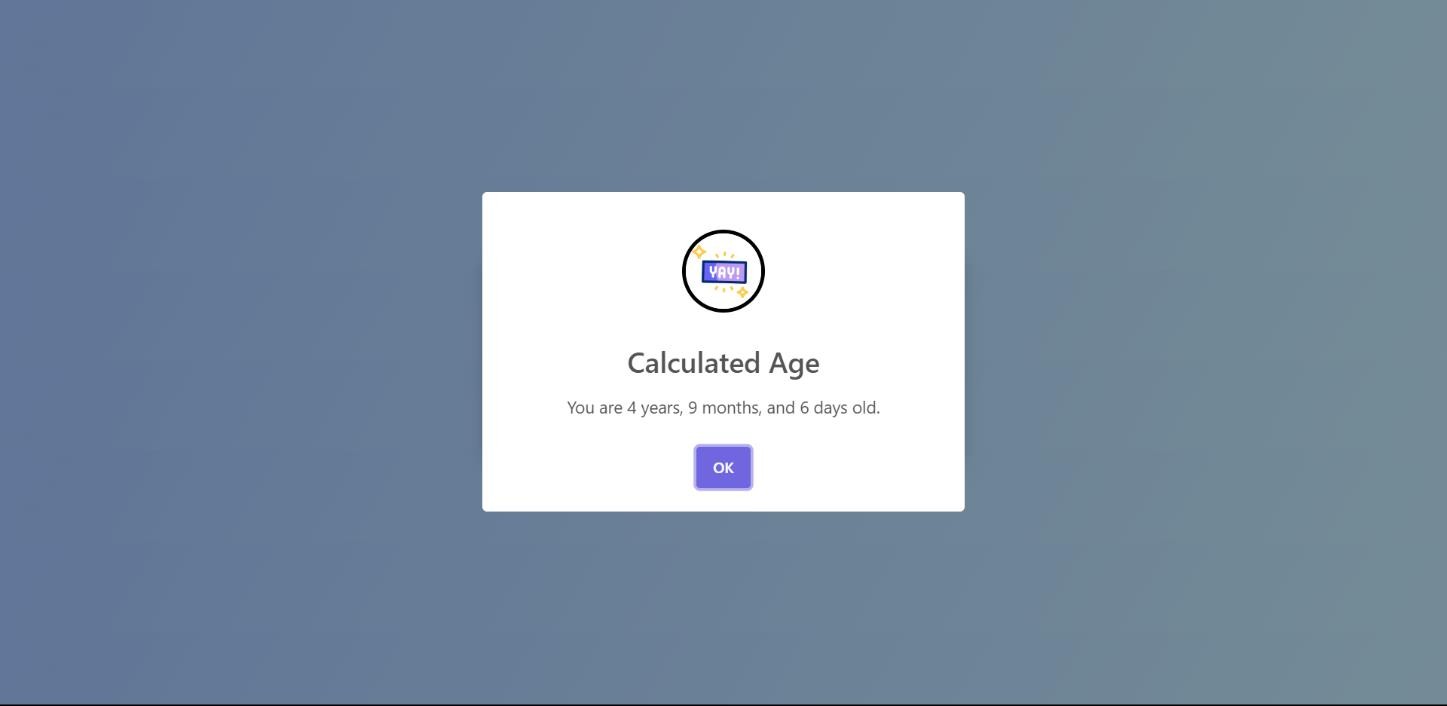
Static Scenario



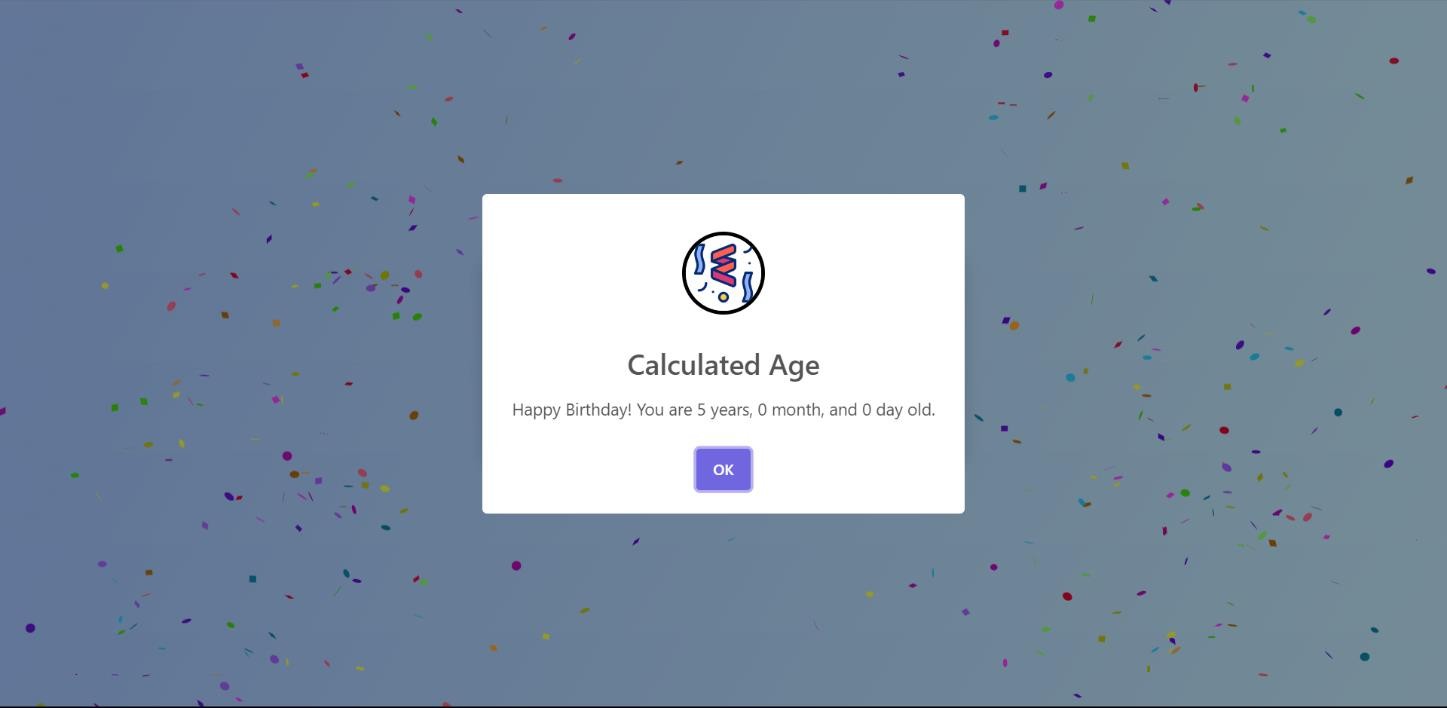
If user enters no input



If user enter a birthdate greater than today’s date



When age is successfully calculated



If user’s birthdate is equal to today’s i.e. birthday scenario with confetti

* **Accuracy:** The age calculation logic, implemented in JavaScript, accurately determines the user's age, accounting for factors such as leap years and varying month lengths. Validation en- sures precise results even for edge cases.
* **User Experience:** The user interface design, leveraging Bootstrap 5, provides a seamless and visually appealing experience across different devices and screen sizes. Interactive ele- ments and real-time feedback enhance usability and engagement.
* **Functionality:** Users can easily input their birth date through the intuitive interface, trigger- ing the age calculation process with the click of a button. The calculated age is displayed dy- namically, with informative alerts provided for invalid input.
* **Integration of Libraries:** The integration of libraries such as SweetAlert2 for user- friendly alerts and confetti for celebratory effects adds an element of delight to the user experi- ence, enhancing engagement and satisfaction.
* **Documentation and Deployment:** Comprehensive documentation guides users through setup, usage, and troubleshooting. Deployment on a web server with domain access ensures public availability, while regular updates maintain relevance and accuracy.

Overall, the age calculator application delivers on its objectives, providing users with a reliable, intuitive, and enjoyable tool for determining their age with ease and accuracy.

# References

## HTML, CSS, and JavaScript Documentation:

* + **Mozilla Developer Network (MDN) - HTML:** https://devel- oper.mozilla.org/en-US/docs/Web/HTML
  + **Mozilla Developer Network (MDN) - CSS:** https://developer.mozilla.org/en-

US/docs/Web/CSS

* + **Mozilla Developer Network (MDN) - JavaScript:** https://devel- oper.mozilla.org/en-US/docs/Web/JavaScript

## Bootstrap Documentation:

* + **Bootstrap Official Documentation:** https://getbootstrap.com/docs/5.3/getting- started/introduction/
  + **W3Schools Bootstrap Tutorial:** https:/[/www.w3schools.com/bootstrap/boot](http://www.w3schools.com/bootstrap/boot-)- strap\_get\_started.asp

## Confetti.js Documentation and Examples:

* + **Confetti.js GitHub Repository:** https://github.com/mathusummut/confetti.js
  + **Confetti.js Examples and Demos:** https://mathusummut.github.io/confetti.js/

## SweetAlert2 Documentation and Examples:

* + **SweetAlert2 Official Documentation:** https://sweetalert2.github.io/
  + **SweetAlert2 GitHub Repository:** https://github.com/sweetalert2/sweetalert2

## Frontend Development Tutorials and Articles:

* + **CSS-Tricks:** https://css-tricks.com/
  + **Smashing Magazine - HTML/CSS:** https://[www.smashingmagazine.com/cat-](http://www.smashingmagazine.com/cat-) egory/css/
  + **JavaScript.info:** https://javascript.info/