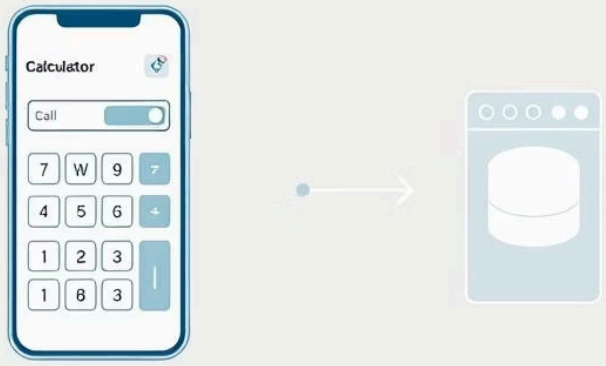
The Calculator Project: A Deep Dive

Welcome to the Calculator Project. We'll explore its architecture, development, and capabilities. A journey into precision computing.

by Gurbaksh Singh



Project Overview & Core Functionality

Essential Operations

Basic arithmetic, roots, powers

Flask Backend

Robust and scalable server logic

Intuitive Interface

Clean HTML/CSS design

Mathematical Precision

Advanced Functions

Trigonometry (sin, cos, tan)

Logarithms (log, ln)

Exponentials (e^x)

Constant Support

Includes constants like π and e

Accurate to high decimal places

Python Flask Backend Architecture

Request Handling

Receives user input

Expression Parsing

Breaks down mathematical strings

Computation Engine

Performs calculations

Response Generation

Sends results to frontend

Frontend: HTML & CSS for User Experience

Structured HTML

Semantic tags for calculator layout

Clear button and display elements

Styling with CSS

Responsive design for various devices

Modern, clean aesthetic

Interactive Elements

Buttons for numbers and operations

Real-time display updates

Deployment and Accessibility

Seamless Deployment

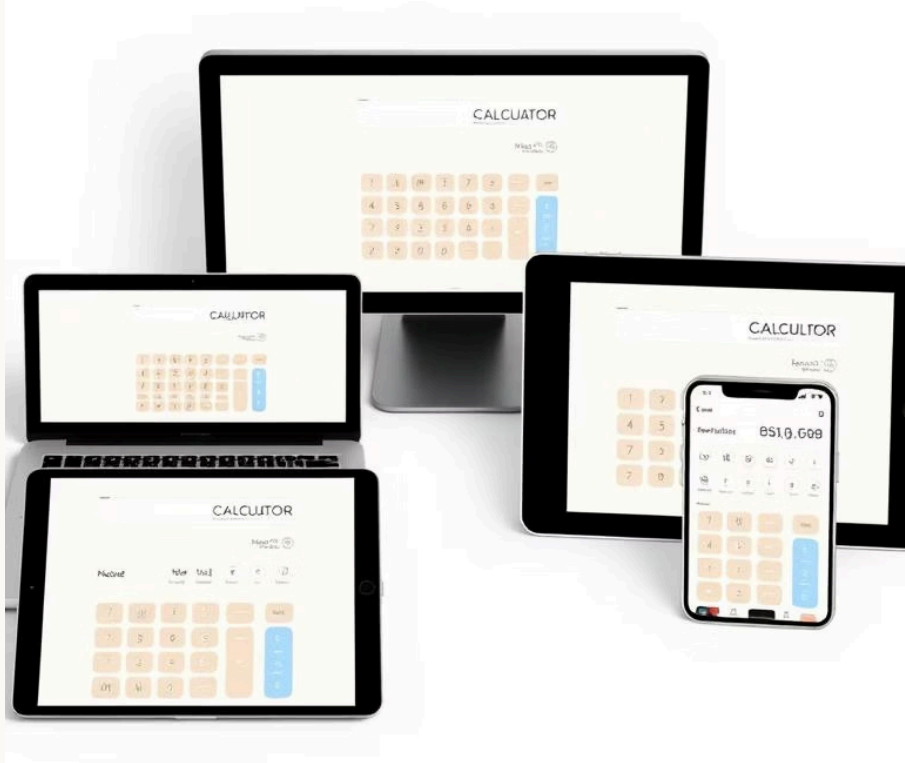
Leverages Flask's built-in server

Web-Based Access

Accessible from any browser

Cross-Device Compatibility

Works on desktops, tablets, mobiles



Future Enhancements & Scalability

Memory Functions

Store and recall previous calculations

Graphing Capabilities

Visualize mathematical functions

Unit Conversions

Integrate various unit types

User Customization

Personalize themes and layouts

Key Takeaways & Next Steps

1

Robust Backend

Python Flask for logic

2

Clean Frontend

HTML/CSS for UI

3

Mathematical Accuracy

Precision in calculations