

Business Requirement Document (BRD)

Project Name: Simple Calculator App

Version: 1.0

Date: 19th May 2025

Author: Aneesh Gangurde

1. Introduction

1.1 Purpose

Build a lightweight calculator web application that performs basic arithmetic operations, with a clean UI and error handling.

1.2 Scope

- Supports: Addition, Subtraction, Multiplication, Division.
- Input: Numbers (integers/decimals) via UI buttons or manual entry.
- Output: Displays results in real-time.
- Responsive design (works on desktop/mobile).

2. Functional Requirements

2.1 Core Features

ID	Requirement	Priority
FR-01	User can input numbers (0-9) and decimals (.)	High
FR-02	User can perform +, -, *, / operations	High
FR-03	System displays current input/output	High
FR-04	"Clear" button resets the calculator	Medium
FR-05	Handles divide-by-zero errors	High

2.2 UI Mockup (Text Description)

- **Layout:** Numeric keypad (0-9), operation buttons (+, -, *, /), "=", "C" (Clear).

- **Display:** Screen at the top showing input/output.
- **Styling:** Minimalist (e.g., Bootstrap or Tailwind CSS).

3. Non-Functional Requirements

ID	Requirement
----	-------------

NF-01	Response time < 500ms per operation
-------	-------------------------------------

NF-02	Works on Chrome, Firefox, and Safari
-------	--------------------------------------

NF-03	No login required (stateless)
-------	-------------------------------

4. Technical Stack (PoC)

- **Frontend:** HTML/CSS/JavaScript (or React/Svelte for PoC).
- **Backend:** FastAPI/Django (optional for PoC; can be pure frontend).
- **AI Integration:** GPT-4/Claude to parse this BRD → generate initial code.

5. Assumptions

- No backend storage needed (pure computation).
- No advanced features (scientific ops, history) in PoC.

6. Risks

- AI misinterpreting operation precedence (e.g., $2+3*4 = 14$, not 20).
- UI not responsive on mobile.

7. Approval

Role	Name	Approval Date
Project Manager	Mr. Dhyanendra Bhangre	19 th May 2025