

Simple Calculator with Logs

Internship Project Documentation

Project Overview

This document provides comprehensive information about the 'Simple Calculator with Logs' internship project. It includes the project's objectives, implementation details, and key achievements.

Project Details

- **Developer:** Praveen Yadav
- **Duration:** 20/07/2025 - 05/08/2025
- **Date:** 02/08/2025

Table of Contents

1. Project Introduction
 - 1.1 Objectives
 - 1.2 Success Criteria
2. API (End-Point)
3. Frontend
4. Layout
5. Database
6. Conclusion

Project Introduction

This project involved the development of a simple calculator application that supports basic arithmetic operations (addition, subtraction, multiplication, division) and incorporates logging functionality to record each operation performed. The application was designed to be user-friendly and provide a clear record of user activity.

Objectives

- Develop a functional calculator application.

- Implement a robust logging system.
- Create a user-friendly interface.
- Adhere to coding best practices.
- Document the development process.

Success Criteria

- The calculator performs all basic arithmetic operations accurately.
- All operations are logged with relevant details (timestamp, operation, operands, result).
- The application is easy to use and understand.
- The codebase is well-structured and maintainable.
- The project documentation is complete and accurate.

API (End-Points)

- POST /api/calculations - Create new calculation.
- GET /api/calculations - Retrieve all calculations.
- GET /api/calculations/:id - Get specific calculation.
- DELETE /api/calculations - Delete all calculations.
- DELETE /api/calculations/:id - Delete specific calculation.

Frontend

- Calculator Component:-
 - State management **for** display and operations.
 - Input validation and error prevention.
 - Integration **with** backend **API**.
- History Panel Component.
 - Fetches history **from** backend **API**.
 - Displays calculations **in** chronological order.
 - Clickable results **for** reuse.
 - Clear all functionality **with** confirmation.
 - Loading states and error handling.
- Custom Hooks.
 - **API** communication functions.
 - Server connection status.
 - Loading state management.

Layout

Calculator

Simple calculator with history logs

● Connected to backend

6 + 3 = 9

9

⌚ Hide History

AC	←	÷	×
7	8	9	-
4	5	6	+
1	2	3	=
0	.		

⌚ Calculation History

⌫ Clear All

6.0000000000 +

3.0000000000 =

Database

The screenshot shows a database interface with a toolbar at the top containing various icons for file operations, search, and navigation. A status bar at the top right indicates "Limit to 1000 rows". Below the toolbar, a SQL query is displayed: "1 • SELECT * FROM calculator_db.calculations;". The main area shows a "Result Grid" with the following data:

	id	operand1	operator	operand2	result	timestamp
▶	8	1.00000000000	+	9.00000000000	10.00000000000	2025-08-04 10:00:14
▶	9	6.00000000000	+	3.00000000000	9.00000000000	2025-08-04 10:00:29
*	NUL	NUL	NUL	NUL	NUL	NUL

A blue sidebar on the right contains a "Result Grid" icon and a "Result Grid" label.

Conclusion

The 'Simple Calculator with Logs' project was a valuable learning experience. It provided practical experience in software development, logging implementation, and documentation. The project successfully met its objectives and provides a solid foundation for future enhancements and learning.