

# Calculator Design Document

## 1. Introduction

This document outlines the technical design for a browser-based calculator with advanced functionalities, implemented using modern web technologies. The calculator features a user-friendly interface, supports basic and advanced mathematical operations, and includes user authentication. Importantly, the sign-up/log-in process does not interact with the calculator's behavior.

## 2. User Interface

The calculator's frontend leverages the "next" framework, incorporating TypeScript and Tailwind. The interface has a comprehensive set of features, including a numeric keypad with decimal point, arithmetic operation buttons, a dynamic display for inputs and results, as well as specialized functions such as memory operations, percentages, square root, exponentiation, and history tracking. External libraries have been utilized to enhance user experience, such as integrating a toaster for intuitive notifications.

## 3. Order of Operations (PEMDAS)

The calculator operations follows the PEMDAS order of operations by using the built-in javascript function `.eval()`.

## 4. Authentication

### 4.1 Sign Up and Log In

To guarantee a seamless experience, the sign-up and log-in process has been strategically implemented to function independently of the calculator's core behavior. Utilizing the Express framework, the backend efficiently manages user authentication. Secure password storage techniques are employed to safeguard user data within the MongoDB database and bcrypt. All the "logged in"/ "signed in" users receive a JWT.

## 5. Features

### 5.1 Memory Functions

The calculator integrates memory functions (M+, M-, MR, MC) for enhanced versatility. These functions operate within the calculator's workflow, allowing users to store and retrieve values without disrupting their mathematical processes.

## **5.2 Percentage Function**

The percentage function (%) augments the calculator's capabilities, enabling users to effortlessly calculate percentages.

## **5.3 Square Root Function**

The square root function ( $\sqrt{\phantom{x}}$ ) has been integrated into the calculator's user interface, providing users with the ability to calculate square roots effortlessly.

## **5.4 Exponential Function**

Empowering users to perform advanced calculations, the calculator features an exponential function ( $\wedge$ ). This function seamlessly integrates into the existing framework, allowing users to raise numbers to specified exponents.

## **5.5 History Function**

The calculator's history function diligently tracks and showcases a log of calculations. This feature allows users to revisit and reuse previous calculations.

# **6. Conclusion**

This calculator possesses numerous powerful features. In addition, it effectively displays the user's authentication status. This application leverages cutting-edge technologies such as Next, enabling server-side rendering for improved frontend performance, along with robust tools like Node, Express, and MongoDB for efficient backend authentication handling.