**Software Requirements Specification for Calculator App**

1. **Introduction**
   1. Purpose

The purpose of this document is to provide a detailed description of the Calculator App, which will have basic arithmetic operations, scientific calculations, and graph plotting capabilities.

* 1. Scope

This document will cover the functional and non-functional requirements, the interfaces, and the interactions that the Calculator App will have with the users.

* 1. Definitions, Acronyms, and Abbreviations
* GUI: Graphical User Interface
* Tkinter: Standard GUI toolkit for Python
* Matplotlib: Library for creating static, animated, and interactive visualizations in Python.
  1. Overview

The rest of this document will detail the system features, interfaces, constraints, and other necessary specifications required to develop the Calculator App.

1. **Overall Description**
   1. Product Perspective

The Calculator App is a standalone application that provides users with mathematical computation and graph plotting functionalities.

* 1. Product Functions
* Basic Arithmetic Operations
* Scientific Calculations
* Graph Plotting
  1. User Classes and Characteristics
* General Users: Individuals requiring basic arithmetic operations.
* Students/Professionals: Individuals requiring advanced scientific calculations and graph plotting functionalities.
  1. Operating Environment

The Calculator App will run on any standard computer with a Python environment.

* 1. Design and Implementation Constraints
* The app will use Python's Tkinter for GUI and Matplotlib for graph plotting.
* The app should run on any standard operating system, including Windows, MacOS, and Linux.

1. **System Features**
   1. Arithmetic Operations

* Addition, Subtraction, Multiplication, and Division.
  1. Scientific Operations
* Trigonometric, Logarithmic, Exponential, etc.
  1. Graph Plotting
* Users can input functions and view plotted graphs.
  1. History
* Display history of calculations.

1. **External Interface Requirements**
   1. User Interfaces

* Graphical User Interface (GUI) designed using Tkinter.
  1. Software Interfaces
* Python 3.11.
* Tkinter for GUI.
* Matplotlib for graph plotting.
  1. Hardware Interfaces
* Standard computer hardware.

1. **Other Non-functional Requirements**
   1. Performance Requirements

* Fast computation and responsive GUI.
  1. Security Requirements
* Exception handling to prevent crashes and erroneous outputs.
  1. Software Quality Attributes
* Usability, Reliability, and Maintainability

1. **Other Requirements**
   1. Error handling and messages

* Clear error messages for incorrect input.