**SOFTWARE REQUIREMENT SPECIFICATION**

**DOCUMENT**

**LIBRARY MANAGEMENT SYSTEM**

**Version:** Version 2.0



**ABSTRACT**

This document is intended to be the SRS for develop **CALCULATOR SYSTEM**



| **Project Title** | **CALCULATOR SYSTEM** | | |
| --- | --- | --- | --- |
| **Lead Institution** | **THE INTERNATIONAL SCHOOL - DUY TAN UNIVERSITY** | | |
| **Project Mentor** | **Mr. Nguyen Dang Quang Huy** | | |
| **Team Name** | **Team 3** | | |
| **Team Members** | **Nguyễn Quang Thông** | | |
| **Mai Hoàng Tùng** | | |
| **Trần Quốc Hoàng** | | |
| **Phạm Ngọc Trung** | | |
| **Phan Quang Trung** | | |
| **Start Date** | Mar 23,2024 | **End Date** | Mar 24,2024 |

**ROPRIETARY INFORMATION**: The information contained in this document is the property of **TEAM 3**. Except as specifically authorized in writing by **TEAM 3**, the holder of this document shall keep all information contained herein confidential and shall protect same in whole or in part from disclosure and dissemination to all third parties

**Table of Contents**

[Revision History 4](#_heading=h.gjdgxs)

[1. Introduction 6](#_heading=h.30j0zll)

[1.1. Purpose 6](#_heading=h.1fob9te)

[1.2. Intended Audience and Reading Suggestions 6](#_heading=h.3znysh7)

[1.3. References 6](#_heading=h.2et92p0)

[2. Project Overview 6](#_heading=h.tyjcwt)

[2.1. Project Description 6](#_heading=h.3dy6vkm)

[2.2. Business Need 7](#_heading=h.1t3h5sf)

[2.3. Project Analyst 7](#_heading=h.4d34og8)

[2.3.1. Business Function Diagram 7](#_heading=h.2s8eyo1)

[2.3.2. System Context Diagram 8](#_heading=h.17dp8vu)

[2.4. Software Requirement Specification 11](#_heading=h.3rdcrjn)

[2.4.1. High level Fucntional Requirement (FR) 11](#_heading=h.26in1rg)

[2.4.2. Stakeholders 13](#_heading=h.lnxbz9)

[2.4.3. Usecase 14](#_heading=h.35nkun2)

[2.4.3.1. <<Library Management>> Use Case Diagram Overview 15](#_heading=h.2xcytpi)

[2.4.4. List of use cases 15](#_heading=h.1ksv4uv)

[2.4.5. Use Case Specification 15](#_heading=h.2jxsxqh)

[2.5.1. Activity Diagrams 42](#_heading=h.ihv636)

[Appendix A: Glossary 51](#_heading=h.41mghml)

# Revision History

| **Date** | **Change Iterm** | **Description** | **by** | **Version** |
| --- | --- | --- | --- | --- |
| **23/03/2024** | Get requests from customers | After preparing the questions about the request and received the request from the customer | Nguyễn Quang Thông, Trần Quốc Hoàng | Version 1.0 |
| **23/03/2024** | Start team meeting | Meet and refer to a number of training points, read through the training points and focus on project implementation, the team can fully understand the system requirements to create | Nguyễn Quang Thông, Trần Quốc Hoàng, Mai Hoàng Tùng, Phạm Ngọc Trung, Phan Quang Trung | Verison 1.0 |
| **23/03/2024** | Job analysis | Through specific requirements, analysis, clearly speaking, the leader needs to prepare in advance for the members. | Nguyễn Quang Thông | Verison 1.0 |
| **23/03/2024** | Share the work | Get BFD, contextual diagram, DFD level 1, DFD level 2,  The mandatory rules of the project | Nguyễn Quang Thông, Trần Quốc Hoàng | Verision 1.0 |
| **23/03/2024** | Editing group | BFD, DFD, USE CASE, Context Diagram, font size, font | Nguyễn Quang Thông, Trần Quốc Hoàng | Verision 1.0 |
| **24/03/2024** | Complete DFD, System Context Diagram | DFD 1 and 2, System Context Diagram | Nguyễn Quang Thông, Trần Quốc Hoàng | Verision 2.0 |

# Introduction

## Purpose

This documentation describes a library management system including all needed information and feature materials in detail for implementation. The purposes of this document are as below:

* To supports the project manager having an overview of the system as well as doing project estimation
* To describes the architectural drivers and use cases in details. Based on this document, architect analyst and designer will be able to implement the system easily.
* To supports tester (QC) writing acceptance test and test plan.

## Intended Audience and Reading Suggestions

| Intended Audience | Reading Suggestions |
| --- | --- |
| Project manager | High level functional requirement, business constraints for estimation |
| Architect analyst and designer | Overall description and user cases to architect and design the system |
| Quality control | Overall description and user cases to make test plan and write acceptance test |
|  |  |

## References

# Project Overview

## Project Description

## Information Technology has revolutionized the life of human beings and has made lives easier by the various kinds of applications. In the light of the rapid changes with the use of Information Technology, there are many tools,technologies and systems that have been produced and invented.

## This project is concerned with developing a Calculator System for Duy Tan student in order to make calculating more efficient and easy to handle. The Calculator system enables a fully automated calculated service. The goal of this project is to create a program that takes two input numbers, denoted as 'a' and 'b', and performs basic arithmetic operations on them.

## Business Need

This system had a number of advantages:

- Users can easily perform basic arithmetic operations with just a few simple steps, providing a convenient and user-friendly experience.

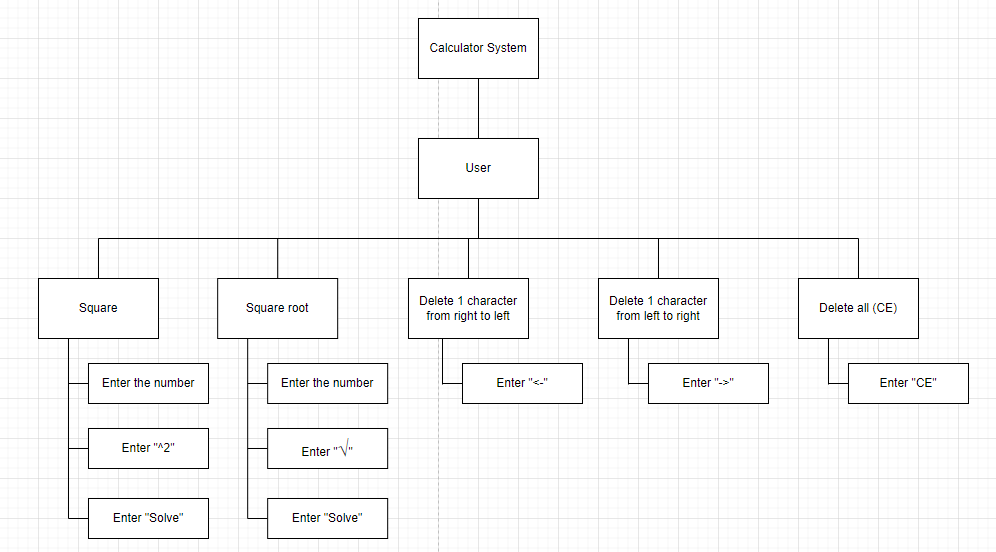
- Supports various basic operations such as addition, subtraction, multiplication, and division, offering flexibility for users in performing different types of calculations.

- The source code is designed for easy maintenance and scalability, making it straightforward for developers to add new features or fix issues when needed.

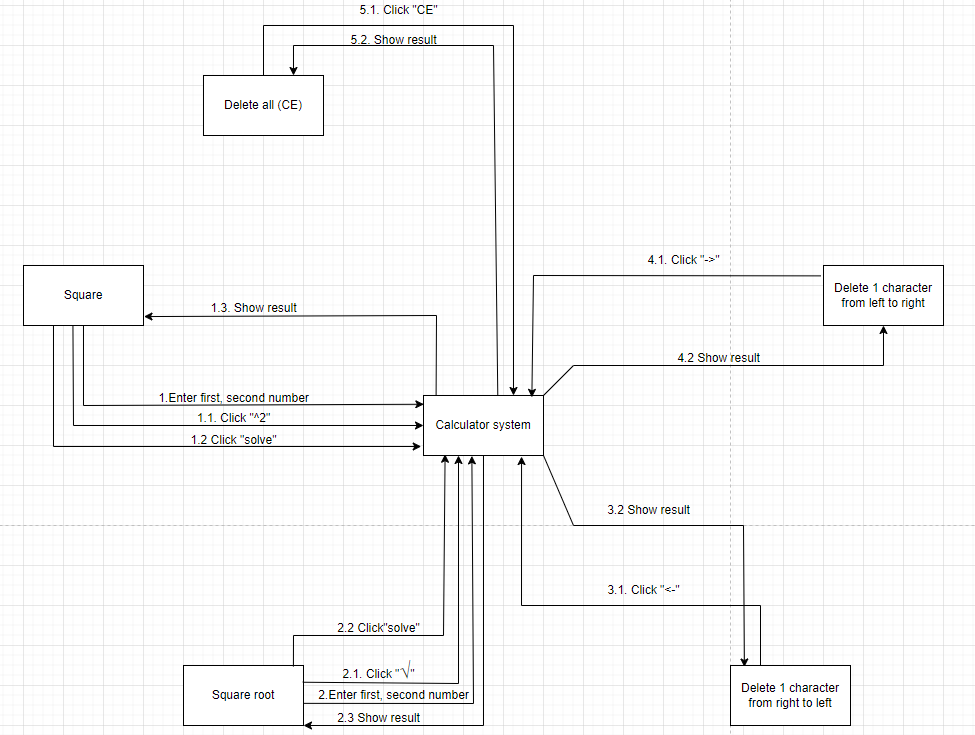
- Easy-to-use interface.

## Project Analyst

### Business Function Diagram



### System Context Diagram



## Software Requirement Specification

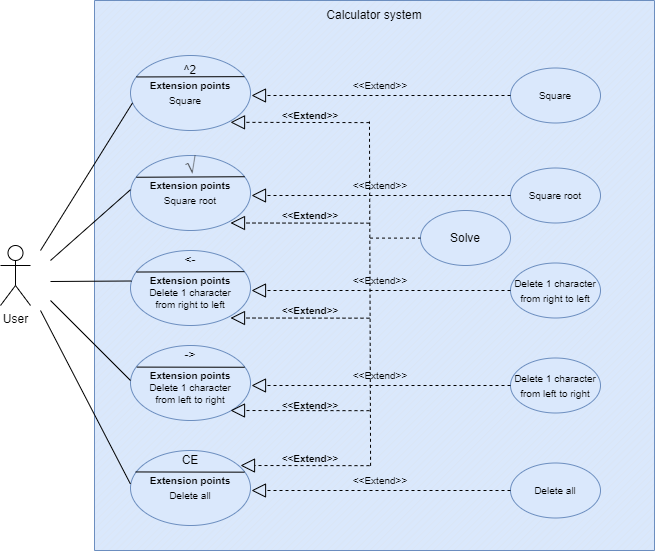
### High level Functional Requirement (FR)

| FR1 | **Title** | **Square** |
| --- | --- | --- |
| **User** | Actor uses this Use Case to perform square calculation in the system. |
| Description | The input to this function usually includes a number, often called "a", that represents the value to be added. The function's operation is simple: the value of "a" is multiplied by itself, creating a product, and the product is then returned for use or display on the screen.This fundamental arithmetic process is a core element in both mathematics and programming. |
| FR2 | **Title** | **Square root** |
| User | The actor uses this Use Case to calculate the square root of a value entered in the system |
| Description | Square root is a calculation used in mathematics to find the square root of a number. When applying this calculation, we find a number that, when multiplied by itself, gives the same result as the number whose square root we need to find. For example, the square root of 9 is 3, because 3 multiplied by itself gives the result. The result is 9. For negative numbers, the square root is usually not defined in real numbers, but it can be defined in complex numbers. |
| FR3 | **Title** | **Delete from right to left** |
| User | The actor uses this Use Case to delete the values ​​of entered numbers from right to left in the system. |
| Description | The act of removing or deleting right-to-left data, information or certain items from a system or application is often a major decision and should be made with caution and confirmation. from the user before implementation. |
| FR4 | Title | **Delete from left to right** |
| User | The user agent Cao uses this to delete the values ​​of entered numbers from left to right. |
| Description | The act of removing or deleting left-to-right data, information or certain items from a system or application is often a major decision and should be made with caution and confirmation. from the user before implementation. |
| FR5 | Title | **Delete all** |
|  | User | The actor uses this Use Case to erase all data entered on the screen. |
|  | Description | The act of removing or deleting all data, information or certain items from a system or application is often an important decision and should be done carefully and with user confirmation before proceeding. into implementation. |

### Stakeholders

| **Stakeholder** | **Description** |
| --- | --- |
| Users | System users |

### Use case



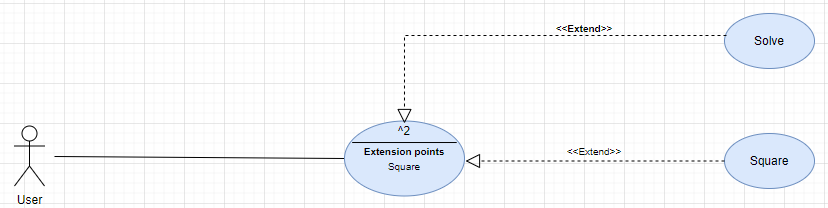
### List of use case

| **Use case ID** | **Use case name** | **Functional Req.** |
| --- | --- | --- |
| UC.01 | Square | FR.1 |
| UC.02 | Square root | FR.2 |
| UC.03 | Delete 1 character from right to left | FR.3 |
| UC.04 | Delete 1 character from left to right | FR.4 |
| UC.05 | Delete all (CE) | FR.5 |

**2.4.5.Use Case Specification**

##### UC 01:Square

1. Use Case Diagram



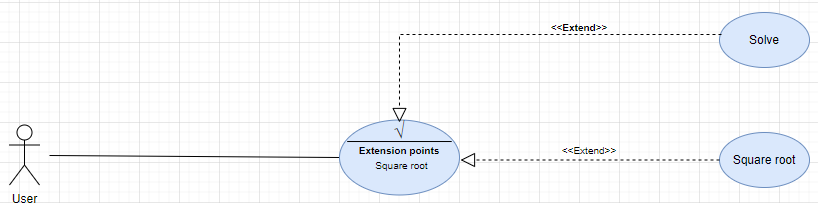
1. Use Case Specification

| Use case ID | UC.01 | | | | |
| --- | --- | --- | --- | --- | --- |
| Use case name | **Square any 2 numbers** | | | | |
| Create by | Nguyễn Quang Thông | | **Last updated by** | | Nguyễn Quang Thông |
| Date created | March 23, 2024 | | **Date last updated** | | March 24, 2024 |
| Actor | Users of the system | | | | |
| Description | This Use Case describes the process of square one arbitrary numbers within the system. | | | | |
| Trigger | Access to the system interface successfully | | | | |
| Pre-condition | **Both values entered are numeric** | | | | |
| Post-condition | If the use case is successful, the system displays the result of square one arbitrary numbers on the user interface. | | | | |
| Main Success | **Step** | **Actor Action** | | **System Response** | |
| Scenario | 1 | Press the "square" button | | 2. The system displays a box for the user to enter 1 numbers | |
| 3 | The user enters 1 numbers into the box | | 4. The system confirms the number the user entered | |
| 5 | Press the "solve" button | | 6. Check the correct data type, the system displays the results to the user | |
| 7 |  | | 8. Display the total results on the screen | |
| Exception | **Step** | **Actor Action** | | **System Response** | |
|  |  | | 6.1 If the data type is checked incorrectly, the system will notify the error and let the user re-enter the number. | |
| Priority | High | | | | |
| Business rule | N/A | | | | |
| Description: | The square function for a number is a fundamental mathematical and programming operation used to multiply a number by itself. The main purpose of squaring is to generate the square of the given number, effectively aggregating the value of the number with itself in a multiplicative manner. | | | | |

##### 

##### UC.02: Square root

a) Use Case Diagram

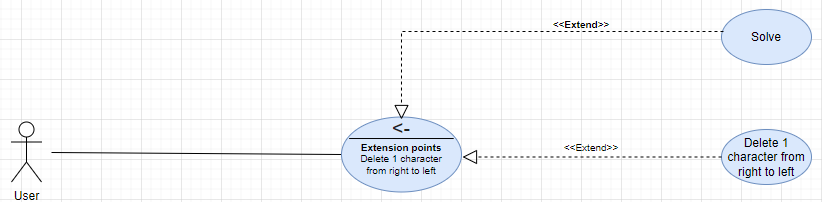


b) Use Case Specification

| Use case ID | UC.02 | | | | |
| --- | --- | --- | --- | --- | --- |
| Use case name | **Square root any 2 numbers** | | | | |
| Create by | Nguyễn Quang Thông | | **Last updated by** | | Nguyễn Quang Thông |
| Date created | March 23, 2024 | | **Date last updated** | | March 24, 2024 |
| Actor | Users of the system | | | | |
| Description | This Use Case describes the process of square root one arbitrary numbers within the system. | | | | |
| Trigger | Access to the system interface successfully | | | | |
| Pre-condition | **Both values entered are numeric** | | | | |
| Post-condition | If the use case is successful, the system displays the result of square root two arbitrary numbers on the user interface. | | | | |
| Main Success | **Step** | **Actor Action** | | **System Response** | |
| Scenario | 1 | Press the "Square root" button | | 2. The system displays a box for the user to enter 1 numbers | |
| 3 | The user enters 1 numbers into the box | | 4. The system confirms the number the user entered | |
| 5 | Press the "solve" button | | 6. Check the correct data type, the system displays the results to the user | |
| 7 |  | | 8. Display the total results on the screen | |
| Exception | **Step** | **Actor Action** | | **System Response** | |
|  |  | | 6.1 If the data type is checked incorrectly, the system will notify the error and let the user re-enter the number. | |
| Priority | High | | | | |
| Business rule | N/A | | | | |
| Description: | The square root function for a number is a fundamental mathematical and programming operation used to find a value that, when multiplied by itself, gives the original number. The main purpose of the square root is to deconstruct the value of the given number into its base multiplier, effectively disaggregating the squared value back to its original form. | | | | |

##### UC.03: Delete 1 character from right to left

1. Use Case Diagram

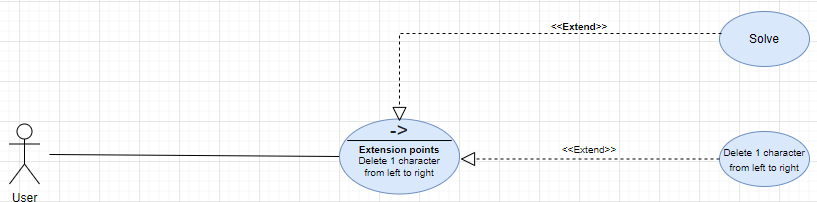


1. Use Case Specification

| Use case ID | UC.03 | | | | |
| --- | --- | --- | --- | --- | --- |
| Use case name | **Delete 1 character from right to left** | | | | |
| Create by | Nguyễn Quang Thông | | **Last updated by** | | Nguyễn Quang Thông |
| Date created | March 23, 2024 | | **Date last updated** | | March 24, 2024 |
| Actor | Users of the system | | | | |
| Description | This Use Case describes the process of delete 1 character from right to left within the system. | | | | |
| Trigger | Access to the system interface successfully | | | | |
| Pre-condition |  | | | | |
| Post-condition | If the use case is successful, the system displays the result of delete 1 character from left to right on the user interface. | | | | |
| Main Success | **Step** | **Actor Action** | | **System Response** | |
| Scenario | 1 | Press the "Delete 1 character from right to left" button | | 2. The system receives and delete 1 character from right to left | |
| Exception | **Step** | **Actor Action** | | **System Response** | |
|  |  | |  | |
| Priority | High | | | | |
| Business rule | N/A | | | | |
| Description: | The deletion function for a character in a string is a fundamental operation in both mathematics and programming, used to decrease the length of a string by removing one character at a time from right to left. The main purpose of deletion is to decrement the sequence of characters in the given string. | | | | |

##### UC.04: Delete 1 character from left to right

1. Use Case Diagram

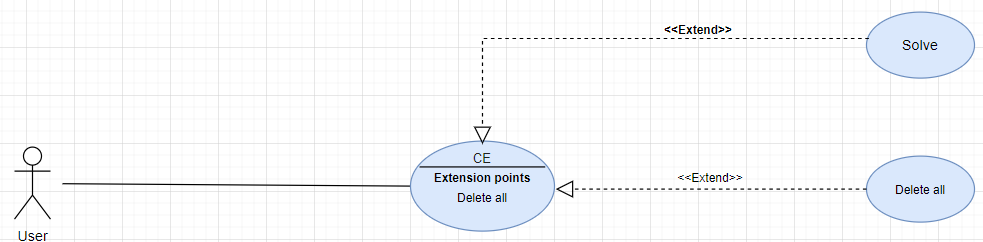


1. Use Case Specification

| Use case ID | UC.04 | | | | |
| --- | --- | --- | --- | --- | --- |
| Use case name | **Delete 1 character from left to right** | | | | |
| Create by | Nguyễn Quang Thông | | **Last updated by** | | Nguyễn Quang Thông |
| Date created | March 23, 2024 | | **Date last updated** | | March 24 2024 |
| Actor | Users of the system | | | | |
| Description | This Use Case describes the process of delete 1 character from left to right within the system. | | | | |
| Trigger | Access to the system interface successfully | | | | |
| Pre-condition |  | | | | |
| Post-condition | If the use case is successful, the system displays the result of delete 1 character from left to right on the user interface. | | | | |
| Main Success | **Step** | **Actor Action** | | **System Response** | |
| Scenario | 1 | Press the "Delete 1 character from left to right" button | | 2. The system receives and delete 1 character from left to right | |
| Exception | **Step** | **Actor Action** | | **System Response** | |
|  |  | |  | |
| Priority | High | | | | |
| Business rule | N/A | | | | |
| Description: | The deletion function for a character in a string is a fundamental operation in both mathematics and programming, used to decrease the length of a string by removing one character at a time from left to right. The main purpose of deletion is to decrement the sequence of characters in the given string. | | | | |

##### UC.05: Delete all

1. Use Case Diagram

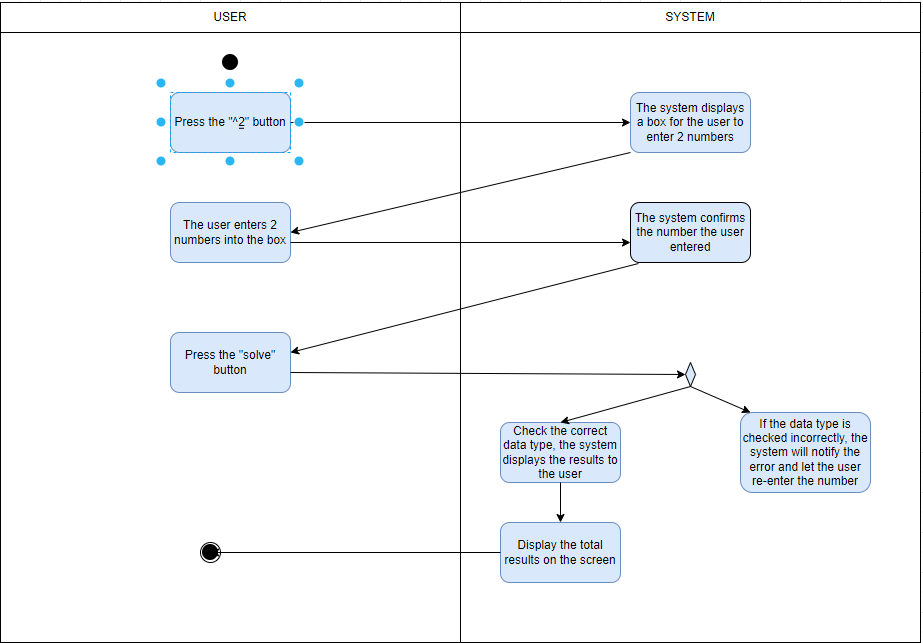


1. Use Case Specification

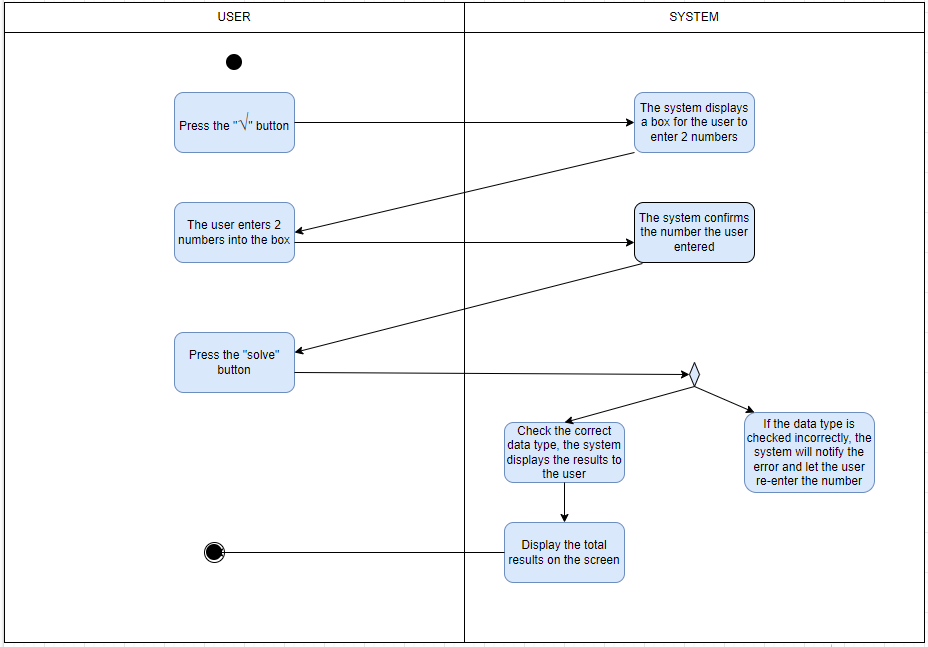
| Use case ID | UC.05 | | | | |
| --- | --- | --- | --- | --- | --- |
| Use case name | **Delete all characters** | | | | |
| Create by | Nguyễn Quang Thông | | **Last updated by** | | Nguyễn Quang Thông |
| Date created | March 23, 2024 | | **Date last updated** | | March 24, 2024 |
| Actor | Users of the system | | | | |
| Description | This Use Case describes the process of delete all characters within the system. | | | | |
| Trigger | Access to the system interface successfully | | | | |
| Pre-condition |  | | | | |
| Post-condition | If the use case is successful, the system displays the result of delete all characters on the user interface. | | | | |
| Main Success | **Step** | **Actor Action** | | **System Response** | |
| Scenario | 1 | Press the "Delete all" button | | 2. The system receives and delete all characters. | |
| Exception | **Step** | **Actor Action** | | **System Response** | |
|  |  | |  | |
| Priority | High | | | | |
| Business rule | N/A | | | | |
| Description: | The deletion function for all characters in a string is a crucial operation in both mathematics and programming, utilized to clear the content of a string by removing all characters simultaneously. The primary purpose of this comprehensive deletion is to reset or empty the sequence of characters in the given string entirely. | | | | |

## 2.5.1 . Activity Diagrams

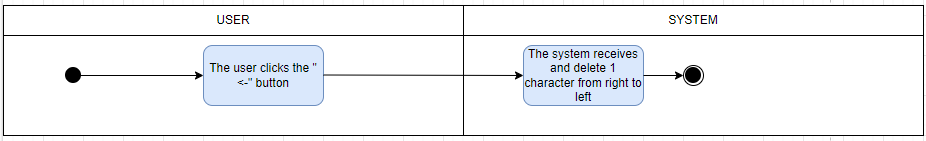
## Square



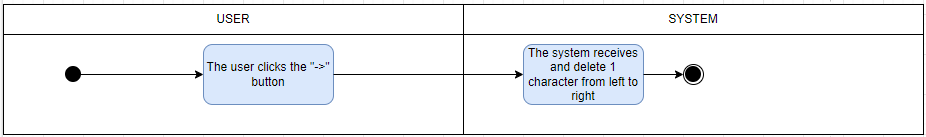
## Square root



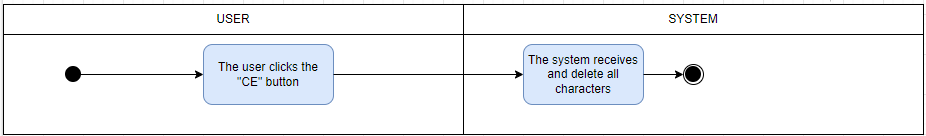
## Delete 1 character from right to left



## Delete 1 character from left to right



## Delete all (CE)



# Appendix A: Glossary

| FR | Functional Requirement |
| --- | --- |
| QA | Quality Attribute |
| UC | Use case |
| BR | Business rule |