./

Learning Report – Applied System Development Life Cycle and Software Testing



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **To be approved By** | **Remarks/Revision Details** |
| 1 |  | Name/PS No | Name/PS No | Module Owner Name | Comments |
| 2 | 15/02/21 | Shivanshu/99003731 |  |  |  |

**Document History**

Table of Contents

[Table of Figures 4](#_Toc64285864)

[Table of Tables 4](#_Toc64285865)

[ACTIVITY 1: SYSTEM/SOFTWARE DEVELOPMENT 5](#_Toc64285866)

[Requirements 5](#_Toc64285867)

[Introduction 5](#_Toc64285868)

[Research 5](#_Toc64285869)

[Cost and Features (Our system) 5](#_Toc64285870)

[SWOT ANALYSIS 6](#_Toc64285871)

[4W's and 1'H 6](#_Toc64285872)

[Who: 6](#_Toc64285873)

[What: 6](#_Toc64285874)

[When: 6](#_Toc64285875)

[Where: 7](#_Toc64285876)

[How: 7](#_Toc64285877)

[HIGH LEVEL REQUIREMENTS 7](#_Toc64285878)

[LOW LEVEL REQUIREMENTS 8](#_Toc64285879)

[UML DESIGN 9](#_Toc64285880)

[1. CLASS DIAGRAM(HLR) 9](#_Toc64285881)

[2. DEPLOYMENT DIAGRAM(LLR) 10](#_Toc64285882)

[3. USE-CASE DIAGRAM(LLR) 11](#_Toc64285883)

[TEST PLAN 12](#_Toc64285884)

[GIT 13](#_Toc64285885)

[GIT ISSUES 13](#_Toc64285886)

[GIT COMMITS 14](#_Toc64285887)

[GIT MAKE FILE 14](#_Toc64285888)

[GIT CODE QUALITY(CPP CHECK) 15](#_Toc64285889)

[REFERENCES 16](#_Toc64285890)

[ACTIVITY 2: AGILE METHODOLOGY 17](#_Toc64285891)

[THEME 17](#_Toc64285892)

[EPIC 17](#_Toc64285893)

[USER STORY 17](#_Toc64285894)

[BASIC ARITHMETIC OPERATIONS: 17](#_Toc64285895)

[PERMUTATION AND COMBINATION: 17](#_Toc64285896)

[MATRIX OPERATIONS: 17](#_Toc64285897)

## Table of Figures

[figure1 1 .CLASS DIAGRAM FOR HLR 9](#_Toc64285898)

[figure1 2. DEPLOYMENT DIAGRAM FOR LOW LLR 10](#_Toc64285899)

[figure1 3. USE-CASE DIAGRAM FOR LLR 11](#_Toc64285900)

[figure1 4.GIT COMMITS 14](#_Toc64285901)

[figure1 5.GIT MAKE FILE 14](#_Toc64285902)

[figure1 6. GIT CPP CHECK 15](#_Toc64285903)

## Table of Tables

[TABLE1 1.HIGH LEVEL REQUIREMENT 7](#_Toc64285904)

[TABLE1 2. LOW LEVEL REQUIREMENTS 8](#_Toc64285905)

[TABLE1 3. TEST PLAN 13](#_Toc64285906)

# 

# ACTIVITY 1: SYSTEM/SOFTWARE DEVELOPMENT

[https://github.com/99003733/SDLC\_N2\_Calculator.git (Link](https://github.com/99003733/SDLC_N2_Calculator.git%20(Link) to GITHUB REPOSITORY)

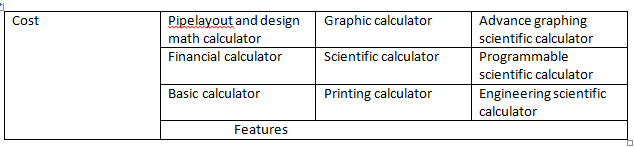
# Requirements

## Introduction

This is a multi-purpose low cost scientific calculator which has near about 4-5 new modes compare to the scientific calculator which already in the market. We have added physics operation, matrix operations, profit loss, finance, permutation and combination, mensuration etc.

## Research

**Cost Vs Feature Curve**



## Cost and Features (Our system)

**1.Cost is near about 1000-1500 Rs**

***2.Features included***

1. Basic arithmetic operation
2. Matrix operations
3. Combinatorics operations

## SWOT ANALYSIS



# 4W's and 1'H

## Who:

**Generally businessman, engineers , students and scientist uses calculator but they use different calculator for each purpose.**

## What:

**There are already so many types of calculator that exist in the market already for making our life easier. Few of the calculators we have researched are basic calculator, scientific calculator, financial, graphing ,printing calculator etc.**

## 

## When:

**This requirement of this type of calculator started because firstly manual calculations are difficult.**

## Where:

**It can be used for design analysis of various mechanical equipment’s such as spaceships, vehicle’s speed and diagnostic analysis etc.**

## How:

**We are going to make functions the necessary physics phenomena’s which are difficult to calculate manually.**

# HIGH LEVEL REQUIREMENTS

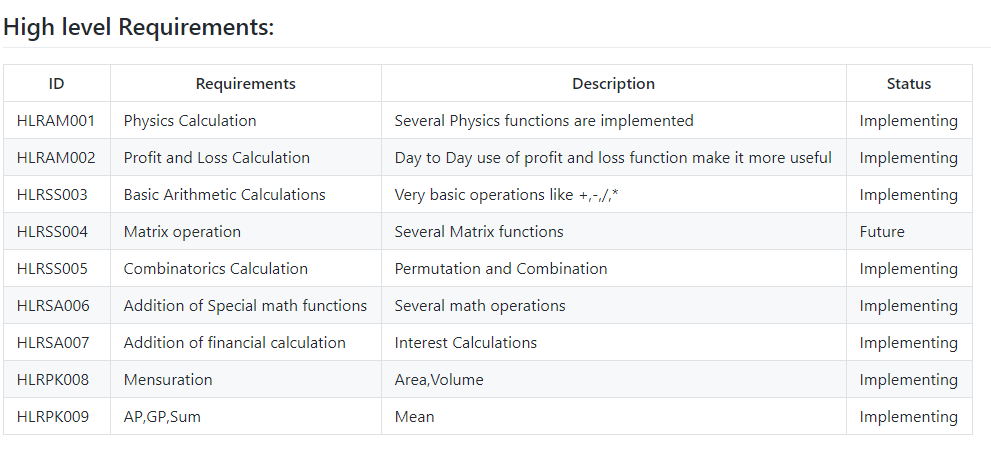
[](#_Table_of_Tables)

TABLE1 .HIGH LEVEL REQUIREMENT

# LOW LEVEL REQUIREMENTS

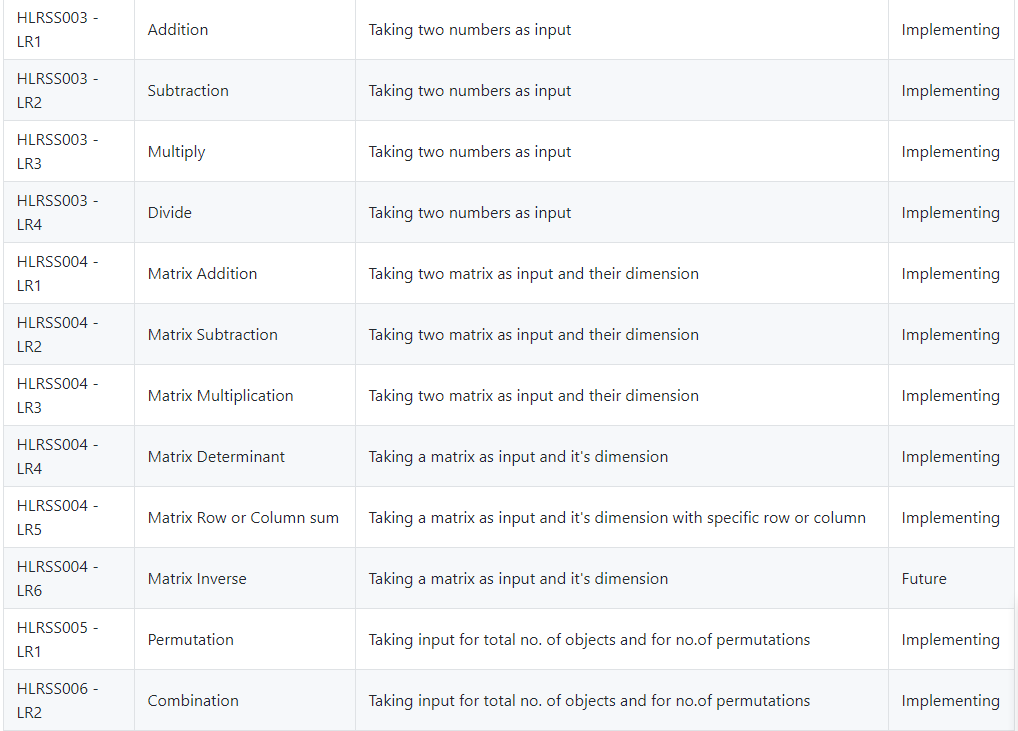
[](#_Table_of_Tables)

TABLE1 . LOW LEVEL REQUIREMENTS

# UML DESIGN

## CLASS DIAGRAM(HLR)

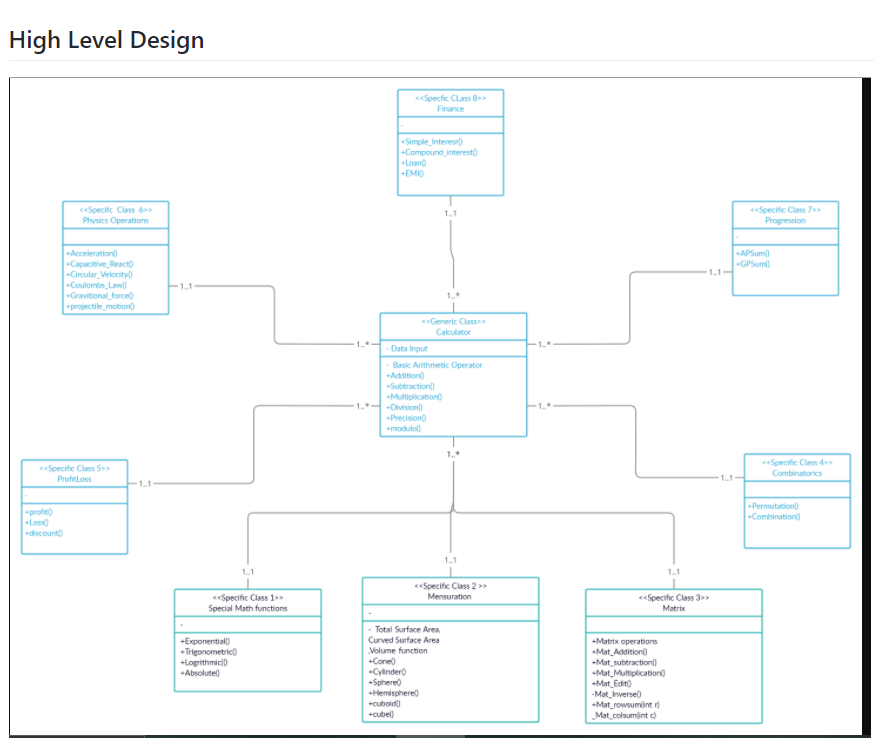
[](#_Table_of_Figures)

figure1 .CLASS DIAGRAM FOR HLR

## DEPLOYMENT DIAGRAM(LLR)

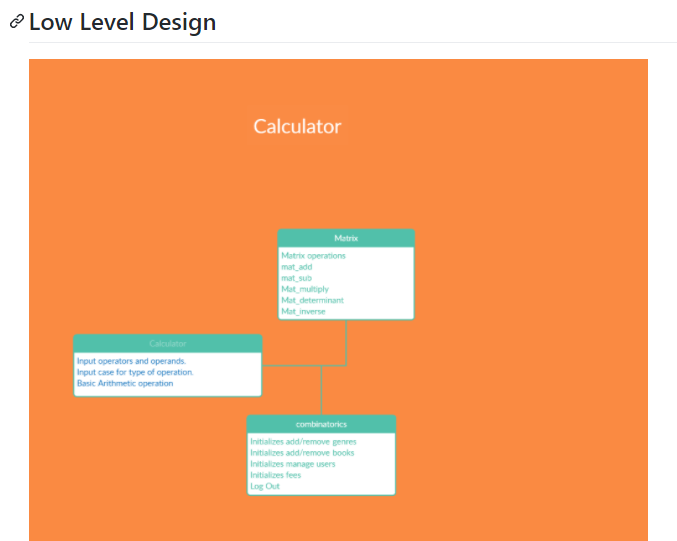


figure1 . [DEPLOYMENT](#_Table_of_Figures) DIAGRAM FOR LOW LLR

## USE-CASE DIAGRAM(LLR)

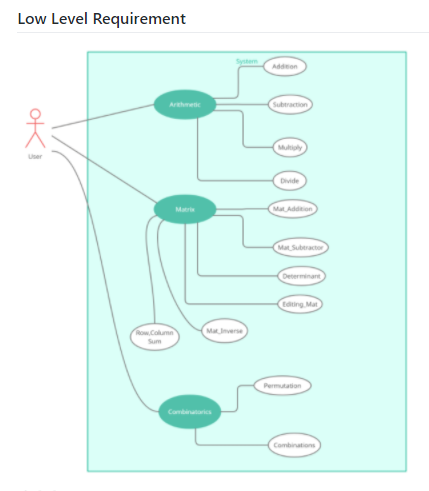
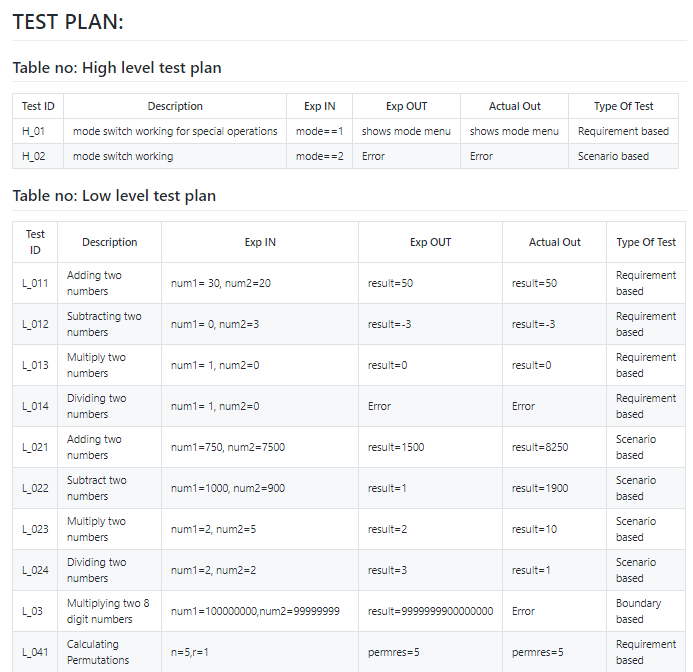
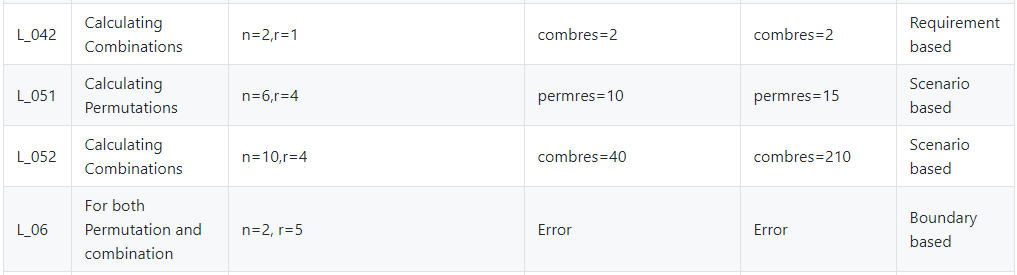


figure1 . USE-CASE DIAGRAM FOR LLR

# TEST PLAN





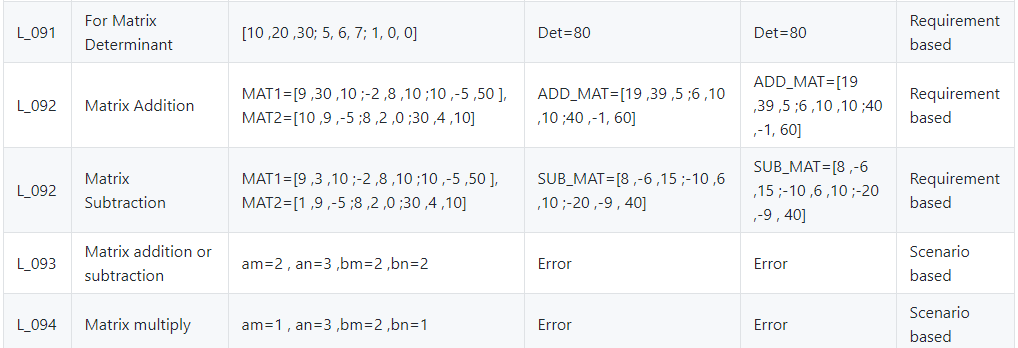
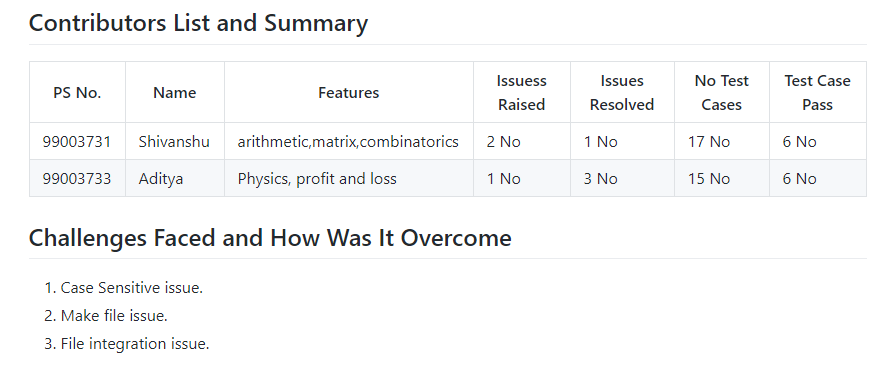
[](#_Table_of_Tables)

TABLE1 . TEST PLAN

# GIT

## GIT ISSUES



## GIT COMMITS

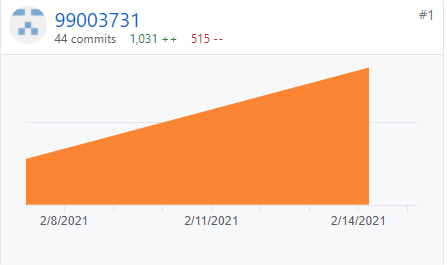
[](#_Table_of_Figures)

figure1 .GIT COMMITS

## GIT MAKE FILE

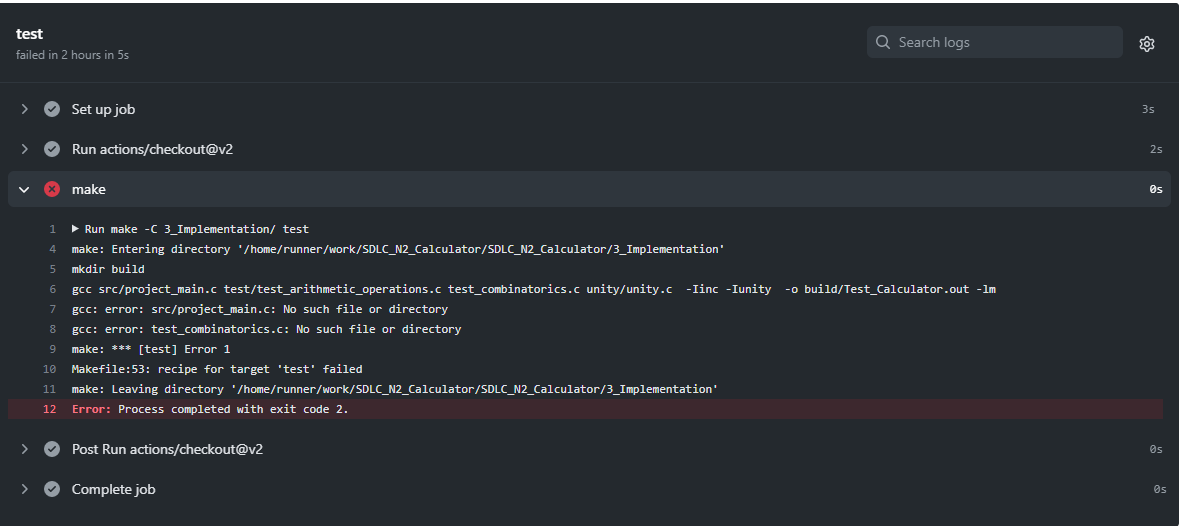
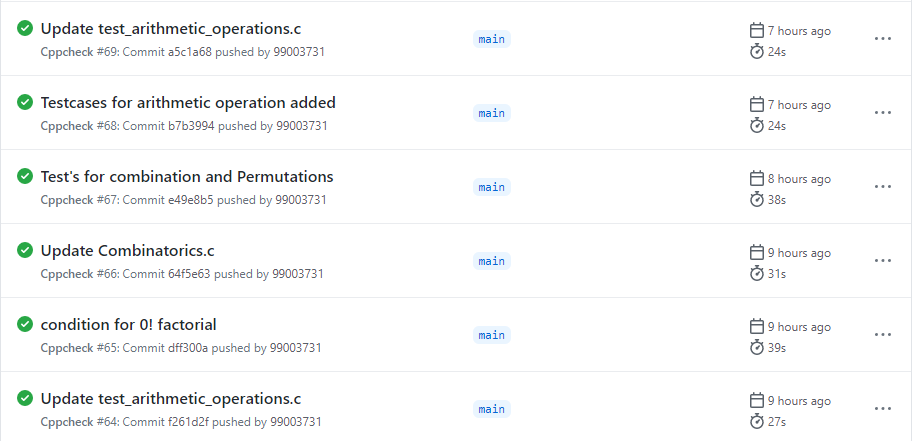
[](#_Table_of_Figures)

figure1 .GIT MAKE FILE

## 

## GIT CODE QUALITY(CPP CHECK)



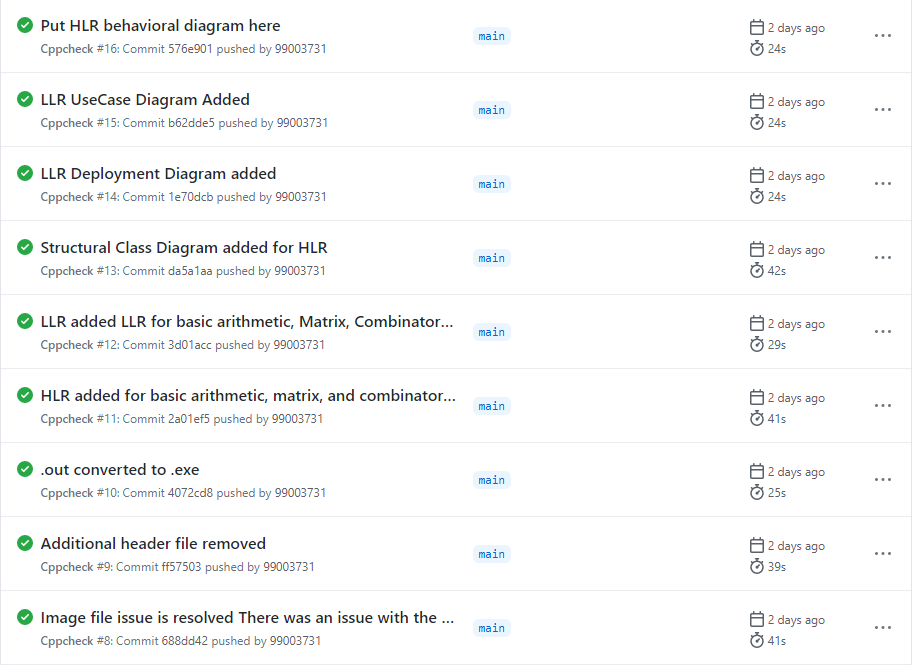
[](#_Table_of_Figures)

figure1 . GIT CPP CHECK

# REFERENCES

1. https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet)

2. https://guides.github.com/features/mastering-markdown/)

3. https://github.com/ejwa/gitinspector.git

4. https://docs.github.com/en/actions/learn-github-action

5. https://stackedit.io/app#

6. https://creately.com/

# ACTIVITY 2: AGILE METHODOLOGY

# THEME

The theme is designing a calculator with certain features according to the specific requirements. The target customers for the calculator are students, engineers ,scientist’s etc .

# EPIC

Epic according to the requirements provided are:

1. Basic Arithmetic Operations.
2. Permutation and Combination.
3. Matrix Operations.

# USER STORY

## BASIC ARITHMETIC OPERATIONS:

1. Addition, Subtraction, Multiplication, Divide output should be of 14 digits.
2. Divide by zero shows error.

## PERMUTATION AND COMBINATION:

1. Ensuring the total no. of objects(n)>=no. of permutations or combinations(r).
2. Output Should be less than 14 digits.

## MATRIX OPERATIONS:

1. Checking matrix is square matrix for determining it’s determinant.
2. Matrix can have maximum dimension as 3x3.
3. Matrix of same dimension’s can be added or subtracted.