

Learning Report – Applied System Development Life Cycle and Software Testing



L&T Technology Services



GLOBAL
ENGINEERING
ACADEMY

Genesis



Document History

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	Himanshu Kumar/99003721			
3					

Table of Contents

ACTIVITY 1: SYSTEM/ SOFTWARE DEVELOPMENT.....	3
INTRODUCTION.....	4
MY PRODUCT: “NAME”.....	5
SWOT ANALYSIS.....	6
REQUIREMENTS.....	6
DESIGN.....	7
HIGH LEVEL DESIGN.....	<i>Error! Bookmark not defined.</i>
LOW LEVEL DESIGN.....	9
TEST PLANS.....	ERROR! BOOKMARK NOT DEFINED.
REFERENCES.....	ERROR! BOOKMARK NOT DEFINED.
ACTIVITY 2: AGILE METHODOLOGY.....	ERROR! BOOKMARK NOT DEFINED.
THEME.....	ERROR! BOOKMARK NOT DEFINED.
EPIC.....	ERROR! BOOKMARK NOT DEFINED.
USER STORY.....	ERROR! BOOKMARK NOT DEFINED.
REFERENCES.....	13
APPENDIX:	ERROR! BOOKMARK NOT DEFINED.

GitHub Repo Link - https://github.com/99003721/AppliedSDLC_Calculator_N1

INTRODUCTION

This is an attempt to build the software development cycle of a semi-engineering calculator. It includes the numerous high-level and low-level requirements of the various features involved in semi-engineering such as log, antilog, exponents roots and polynomial functions. It then explores about the various test cases that should be taken in consideration such that all requirements are taken care of. In the end, we build the codes that follow all the requirements and design efficiently as per them.

MY PRODUCT: “Name ”
Semi-engineering Calculator












Git Resource

Feb 7, 2021 – Feb 15, 2021

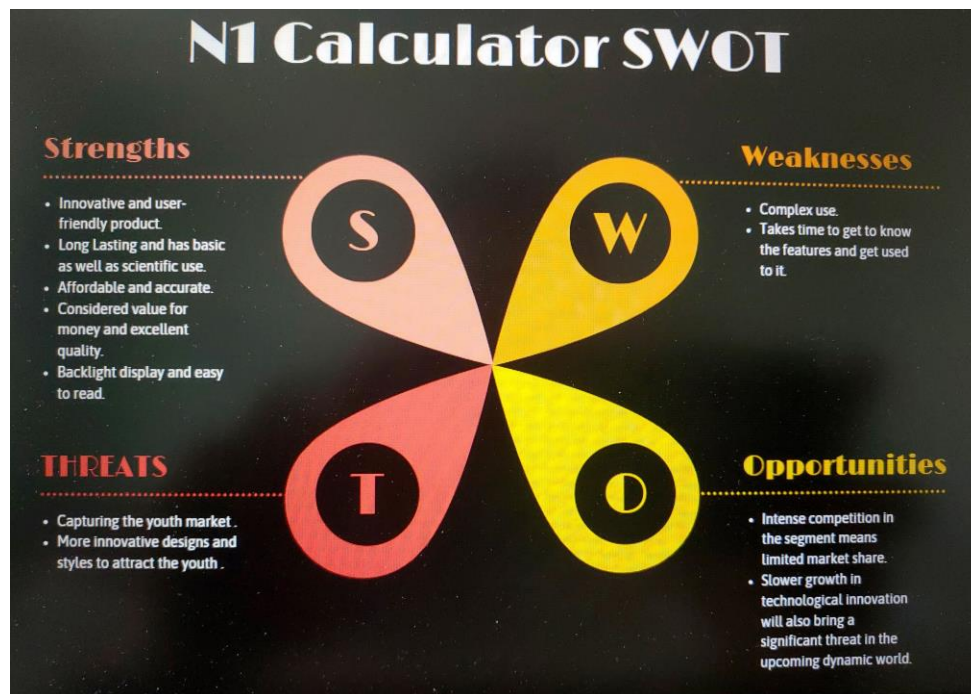
Contributions: Commits ▾

Contributions to master, excluding merge commits



 99003721 Update Readme.md	61e20dd 1 hour ago	 112 commits
 .github/workflows	New local git repo	2 days ago
 .vscode	some part of code done by me on 13/02/21	2 days ago
 1_Requirements	Update Readme.md	4 hours ago
 2_Design	Update Readme.md	1 hour ago
 3_Implementation	Conv test added as comment	5 hours ago
 4_TestPlan	test plan added	4 hours ago
 .gitattributes	New local git repo	2 days ago
 .gitignore	New local git repo	2 days ago
 Readme.md	Update Readme.md	6 hours ago

SWOT ANALYSIS



REQUIREMENTS

HLR –**High Level Requirements:**

id	Requirments	Description	Status
HL001	CALCULATING NUMBERS	BASIC ARITHMETIC CALCULATION- Addition- Subtraction- Division- multiplication	Implemented
HL002	LOG ANTILOG & POWER OPERATIONS	SOLVING POWER FUNCTIONS AND LOG/ANTILOG OPERATIONS	IMPLEMENTED
HL003	TRIGONOMETRIC FUNCTIONS	- SINE- COSINE- TAN	IMPLEMENTED
HL004	PERMUTATION & COMBINATION	PnC FORMULAE OUTPUTS	IMPLEMENTED
HL005	CONVERSIONS OF UNITS	UNIT COVERSIONS FOR TEMPERATURE,LENGTH UNITS,etc	IMPLEMENTED
HL005	FACTORIAL	FACTORIAL OF THE POSITIVE NUMBERS	IMPLEMENTED

LLR –**Low Level Requirements**

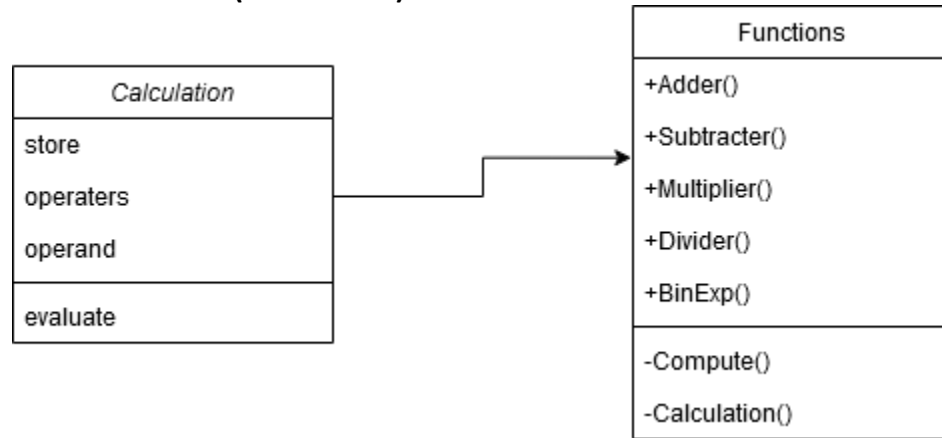
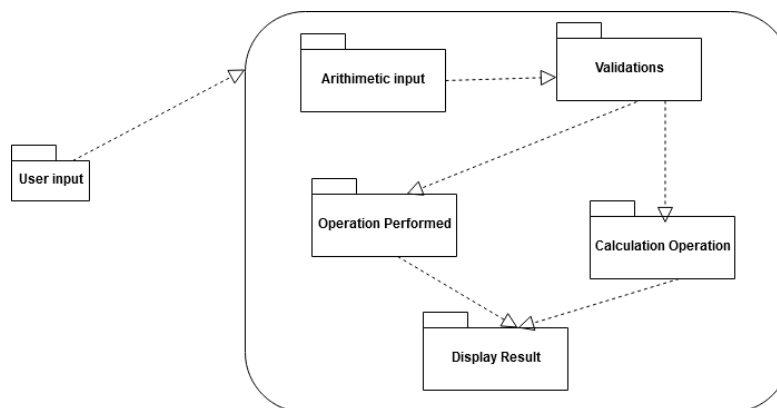
<i>ID_No.</i>	<i>Requirements</i>	<i>Description</i>	<i>Status</i>
LLR001	Input	Two different types of inputs integer type, float type	Implemented
LLR002	ARITHMETIC OPERATION (Addition, Subtraction, Multiplication, Division)	Addition (operand1, operand2) Subtraction (operand1, operand2) Multiplication (operand1, operand2) Division (operand1, operand2)	Implemented
LLR003	Log & Antilog	Log(x) Antilog(x)	Implemented
LLR004	Trigonometry Functions	To find Sine(x), Cosine(x), Tangent(x)	Implemented
LLR005	Permutation & Combination	To find Permutation (operand1, opeerand2) Combination (operand1, opeerand2)	Implemented

DESIGN

—

HIGH LEVEL DESIGN –

**Figure 1 –
CLASS DIAGRAM(HIGH LEVEL)**

**LOW LEVEL DESIGN****Package DIAGRAM (HIGH LEVEL)**

COMMUNICATION DIAGRAM (LOW LEVEL)



TEST PLANS –

Table no: High level test plan

Test ID	Description	Exp IN	Exp OUT	Actual Out	Type Of Test
L_01	Verify that BODMAS is applied in case of complex queries and the correct result is returned.	$24+3*2$	30	30	Requirement based
L_02	Addition of 2 or more numbers	$20+30$	50	50	Scenario based
L_03	Sequential incorrect parameter	$2+4*3.y$	20	Error	Boundary based
L_04	Out of range check	addition if exceed 15 digit numbers	any output	Result out of range	Requirement based
L_05	Sequential incorrect parameter	$2+4*3.y$	20	Error	Boundary based
L_06	Two symbol cannot be between numbers	$3++5$	2	Invalid Format	Scenario based

Table No 1.: Low level test plan

Test ID	Description	Exp IN	Exp OUT	Actual Out	Type Of Test
H_01	Out of Range check for Arithmetic	Max Addition upto 12 digit	Some value	Overflow Error	Requirement based
H_02	Check for long number	eg :- 1234567889023	Some value	Invalid output	Requirement based
H_03	bracket should be open and close if used	(2x-3=0	(2x-3=0)	Invalid output	Scenario based
H_04	Putting values in valid format	values to be performed	display valid output	Error if invalid	Scenario based
H_05	= sign in the expression	2x=3y=10	valid result	valid result	Boundary based

Table No 3.: Low level test plan

Test ID	Description	expected Input	Expected output	Actual output	Type of test
LLR_P_1	Finding the square of given number i.e. find power of number will be 2.	100^2	10000	10000	logic based
LLR_P_2	Finding the cube of given number i.e. find power of number will be 3.	10^3	1000	1000	logic based
LLR_P_3	Finding the power of given number i.e. find power of number will be any number	2^5	32	32	logic based
LLR_P_4	Finding the power of big number i.e. find power of number which results in digits more than 20	3^20	3.48x10^9	3.48x10^9	Boundary based

REFERENCES –

UML Diagram - <http://draw.io/><https://www.javatpoint.com/software-development-life-cycle><https://www.tutorialspoint.com/cprogramming/index.htm>GitHub Repo Link - https://github.com/99003721/AppliedSDLC_Calculator_N1

AGILE METHODOLOGY

THEME –

We have majorly focus to provide a efficient and simple Semi-engineering calculator for the students. It explore most such basic function in requirements. Our theme is to made a quick and friendly operational calculator. The range we have chosen is for medium range cost and features. Its similar to the scientific calculator and mostly used in calculation.

EPIC –

- Factorial
- Cube Root
- Exponential

USER STORY –

Exponential –

- Power should be valid input
- Exponential $F(x) = a(b^x)$ Where b is positive real no. & b not equal to 1

Factorial –

- Finding the factorial of numbers
- Input_number! Means $n(n-1)(n-2)....3,2,1$.
- First check the given number is positive. If the given integer is a negative the output is not defined
- Submit integer value at lower boundary [0]

Cube root –

- Finding the cube of given number i.e. find power of number will be 3
- Finding the power of given number i.e. find power of number will be any number
- Finding the power of big number i.e. find power of number which results in digits more than 20

REFERENCES –

<https://www.programiz.com/>

<https://www.javatpoint.com/c-programming-language-tutorial>

<https://creately.com/lp/uml-diagram-tool/>

<https://www.lucidchart.com/pages/examples/uml-diagram-tool>

GitHub Repo Link - <https://github.com/99003721/AppliedSDLC-Calculator-N1>