Learning Report –

# SDLC and Testing

A Report By-

Name- Saubhagya Ashish

Ps No- 99002607

Table of Content

[Activity 1 3](#_bookmark0)

[Product Selected - Try On Augmented Reality Watch 3](#_bookmark1)

[Ageing 3](#_bookmark2)

[Requirements 3](#_bookmark3)

[High Level Requirements 3](#_bookmark4)

[Low Level Requirements 3](#_bookmark5)

[DESIGN 4](#_bookmark6)

[High Level Design 4](#_bookmark7)

[Structural Diagram 4](#_bookmark8)

[Behavioral Diagram 5](#_bookmark9)

[Low Level Design 6](#_bookmark10)

[Structural Diagram 6](#_bookmark11)

[Behavioral Diagram 7](#_bookmark12)

[Testing 8](#_bookmark13)

## ACTIVITY -1

**Product Selected -** Try On Augmented Reality Watch

### Ageing -

* Augmented reality technology was invented in 1968, with Ivan Sutherland’s development of the first head-mounted display system. However, the term ‘augmented reality’ wasn’t coined until 1990 by Boeing researcher Tim Caudell.
* A view of the physical real-world environment with superimposed

Computer-generated images, thus changing the perception of reality, is the AR.

* According to Apple CEO Tim Cooke, Augmented Reality is the core technology and will be big technological step forward, which is similar to the release of smartphones.

### Costing of Products-

* Earlier due to shortage of skills a basic augmented application cost around $100 - $200.
* After invention of software like Unity and Spark Augmented and Virtual Reality become quite easy to make.
* In present augmented reality is used almost every platform like e commerce, medical, and etc.
* It is one of the largest employment sector in the world right now.

**SWOT Analysis-**

Strength-

* Application creates the virtual objects.
* Virtual object well defines the original object.

Weakness-

* The application is virtual so can’t be filled or touched.
* The application only runs android platform

Opportunities-

* It gives customer a detail look of the product
* With new updates the product can generate a lot of revenue.

Threats-

* Security concerns are one of major issues.

### Requirements -

#### High Level Requirements -

* + Android Smartphone
  + Minimum Android Version- 8.0
  + Minimum Storage Space of 45 MB
  + In build Camera
  + Image Target
  + Virtual Object Formation

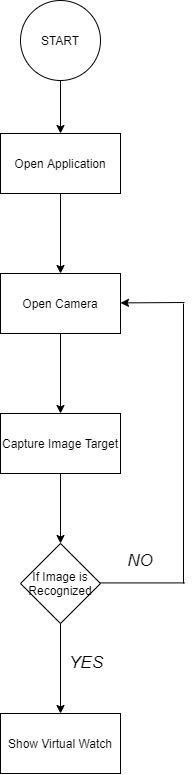
#### 

#### Low Level Requirements -

* + Creating Core Functionality -
    - Create 3 different Watch Models
    - Occlusion Of hand
  + Creating User Interface -
    - Create UI slide in Frame
    - Create Color switch buttons for watches
    - Create a Exit Button

### DESIGN

### High Level Design -

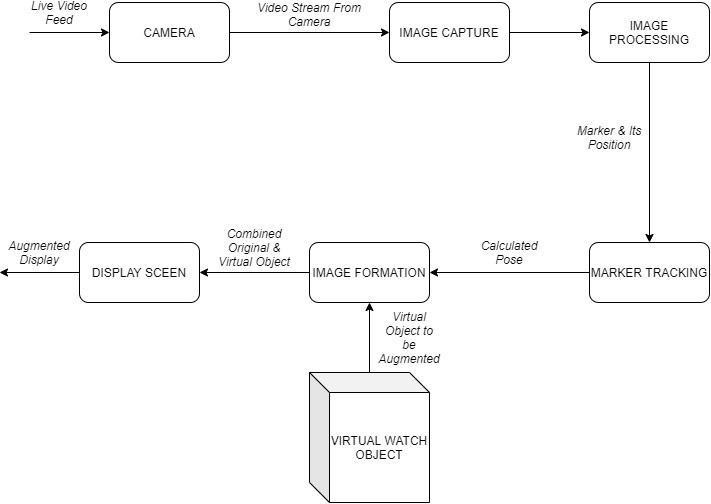


Structural Diagram -

Fig-1 Flow Chart Diagram

Behavioral Diagram

- State Chart Diagram



State Chart Diagram

### Low Level Design-

Structural Diagram -

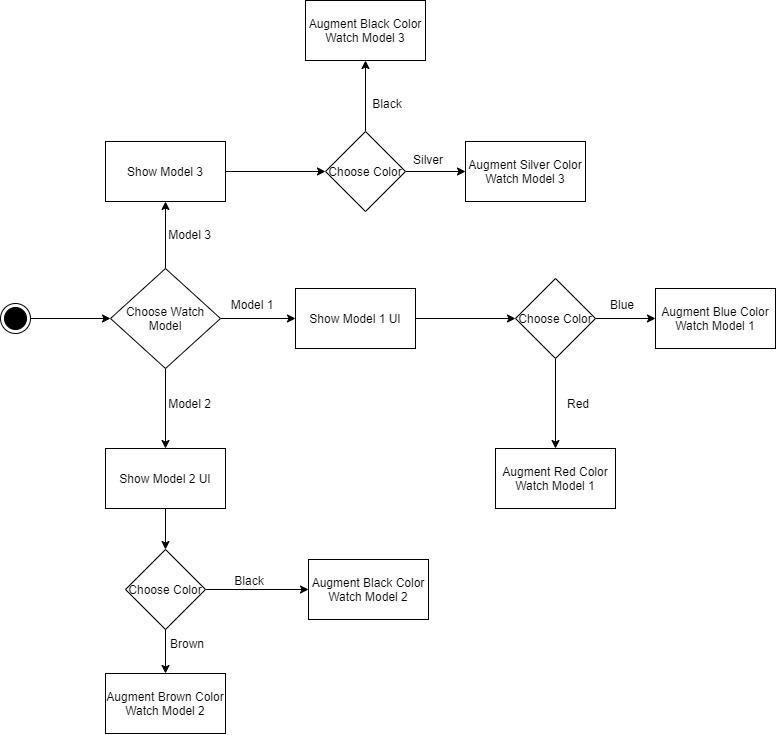
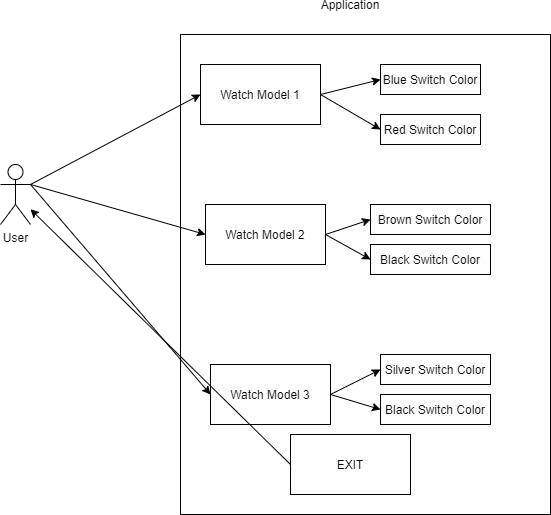


Fig 3: Flow Chart Diagram

Behavioral Diagram -

Fig 4: Use Case Diagram



### Testing -

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TEST ID** | **DESCRIPTION** | **EXPECTED INPUT** | **EXPECTED OUTPUT** | **ACTUAL OUTPUT** |
| T001 | Animation of Watch window 1 | Tap on Window 1 UI | Watch window 1 slides into the frame | Watch window  1 slides into the frame |
| T002 | Animation of Watch window 2 | Tap on Window 2 UI | Watch window 2 slides into the frame | Watch window 2 slides into the frame |
| T003 | Animation of Watch window 2 | Tap on Window 3 UI | Watch window 3 slides into the frame | Watch window 3 slides into the frame |
| T004 | Exit Button of Watch window 1 | Tap on Exit Button of Watch window 1 | Watch window 1 slides back from the frame | Watch window  1 slides back from the frame |
| T005 | Exit Button of Watch window 2 | Tap on Exit Button of Watch window 2 | Watch window 2 slides back from the frame | Watch window  2 slides back from the frame |
| T006 | Exit Button of Watch window 3 | Tap on Exit Button of Watch window 3 | Watch window 3 slides back from the frame | Watch window  3 slides back from the frame |
| T007 | Color Switch Buttons of Watch window 1 | Tap on the Red color switch buttons | Color of Watch Changes to Red | Color of the Watch Changes to Red |
| T008 | Color Switch Buttons of Watch window 1 | Tap on the Blue color switch buttons | Color of Watch Changes to Blue | Color of Watch Changes to Blue |
| T009 | Color Switch Buttons of Watch window 2 | Tap on the Black color switch buttons | Color of Watch Changes to Black | Color of Watch Changes to Black |
| T010 | Color Switch Buttons of Watch window 2 | Tap on the Brown color switch buttons | Color of Watch Changes to Brown | Color of Watch Changes to Brown |
| T011 | Color Switch Buttons of Watch window 3 | Tap on the Black color switch buttons | Color of Watch Changes to Black | Color of Watch Changes to Black |
| T012 | Color Switch Buttons of Watch window 3 | Tap on the Silver color switch buttons | Color of Watch Changes to Silver | Color of Watch Changes to Silver |

#### **ACTIVITY -2**

#### **CALCULATOR**

**Introduction**

#### A calculator is a machine which allows people to do [math](https://simple.wikipedia.org/wiki/Mathematics) operations more easily. For example, most calculators will [add](https://simple.wikipedia.org/wiki/Addition), [subtract](https://simple.wikipedia.org/wiki/Subtraction), [multiply](https://simple.wikipedia.org/wiki/Multiplication), and [divide](https://simple.wikipedia.org/wiki/Division_(mathematics)). Some also do [square roots](https://simple.wikipedia.org/wiki/Square_root), and more complex calculators can help with [calculus](https://simple.wikipedia.org/wiki/Calculus) and draw function graphs. Calculators are found everywhere. A [smartphone](https://simple.wikipedia.org/wiki/Smartphone) or other [computer](https://simple.wikipedia.org/wiki/Computer) can also act as a calculator.

Some calculators, like the [abacus](https://simple.wikipedia.org/wiki/Abacus), will work without [batteries](https://simple.wikipedia.org/wiki/Battery). Others, like the electronic calculator, require batteries. There are two types of electronic calculators: simple calculators, which can only [add](https://simple.wikipedia.org/wiki/Addition), [subtract](https://simple.wikipedia.org/wiki/Subtraction), [multiply](https://simple.wikipedia.org/wiki/Multiplication) and [divide](https://simple.wikipedia.org/wiki/Division_(mathematics)), and sometimes take [square roots](https://simple.wikipedia.org/wiki/Square_roots); and scientific calculators, which can do many other things, such as calculate [factorials](https://simple.wikipedia.org/wiki/Factorial) and [trigonometry](https://simple.wikipedia.org/wiki/Trigonometry) functions.

**Requirements**

**High Level Requirement**

* Performance - The performance of the calculator should be high.
* Speed -The speed of operations performed should be fast.
* It should perform all the arithmetic operations.
* It should find the area of square.
* It should perform conversion operations.
* It should find the factorial of number.
* It should check for prime number.

**Low level requirements**

* It should take operands as input and give the result of addition, subtraction, multiplication and division of the operands accordingly.
* It should take two operands as input and give the area of square.
* It should take an operand as input and make conversion from kilometer to meter, centimeter and millimeter.
* It should take the operand as input and give its factorial as output.
* It should take an operand as input check if it is prime number.

**1. Requirement Mapping**

|  |  |
| --- | --- |
| ID | Description |
| H\_01 | Performance  The performance of the calculator should be high. |
| H\_02 | Speed  The speed of operations performed should be fast. |
| H-03\_L\_01 | It should perform all the arithmetic operations.  It should take operands as input and give the result of addition, subtraction, multiplication and division of the operands accordingly. |
| H-04\_L\_02 | It should find the area of square  It should take two operands as input and give the area of square. |
| H-05\_L\_03 | It should perform conversion operations.  It should take a operand as input and make conversion from kilometer to meter, centimeter and millimeter. |
| H-06\_L\_04 | It should find the factorial of number.  It should take the operand as input and give its factorial as output. |
| H-07\_L\_05 | It should check for prime number.  It should take a operand as input check if it is prime number. |

**Test plan mapping**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Description | Precondition | Expected  input | Expected  output | Actual Output |
| H\_01 | Performance | Should be  90% and  above | - | High performance | - |
| H\_02 | speed | <20ms | - | Speed<20ms | <20ms |
| H\_03 | addition | Two operands  as input | 2 and 3 | 5 | 5 |
| H\_04 | subtraction | Two operands  as input | 3 and 2 | 1 | 1 |
| H\_05 | multiplication | Two operands  as input | 2\*3 | 6 | 6 |
| H\_06 | division | Two operands  as input | 4/2 | 2 | 2 |
| H\_07 | factorial | One operand | 3 | 6 | 6 |
| H\_08 | Conversion from kilometer to  meter | One operand | 100km | 100000m | 100000m |
| H\_09 | Area of square | Two operands  as input | 2 and 2 | 4 | 4 |

**UML Diagram**

1. Class Diagram

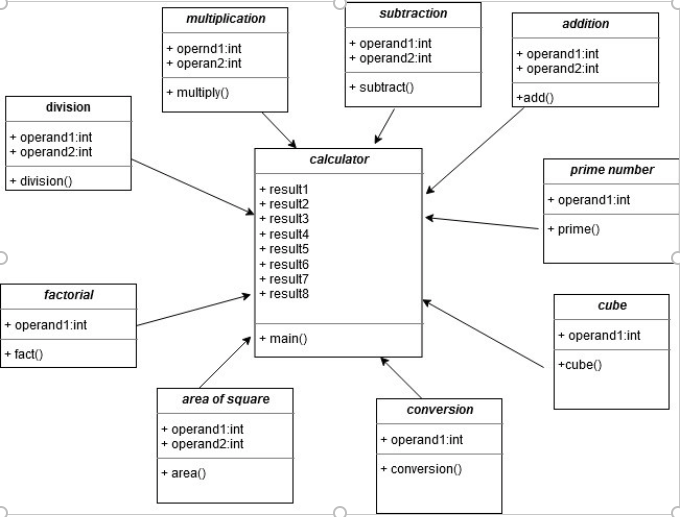


Fig. 5: Class Diagram

1. Use Case Diagram

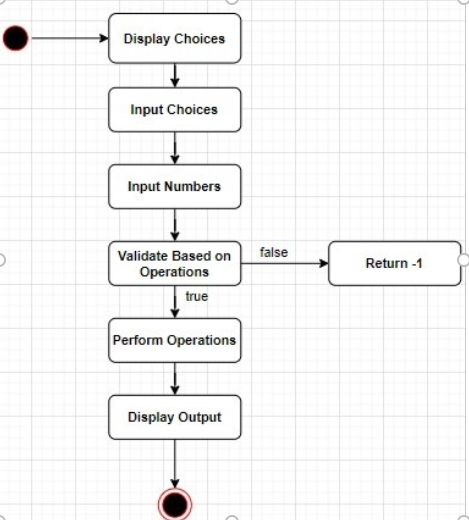


Fig 6: Use Case Diagram

1. Activity Diagram

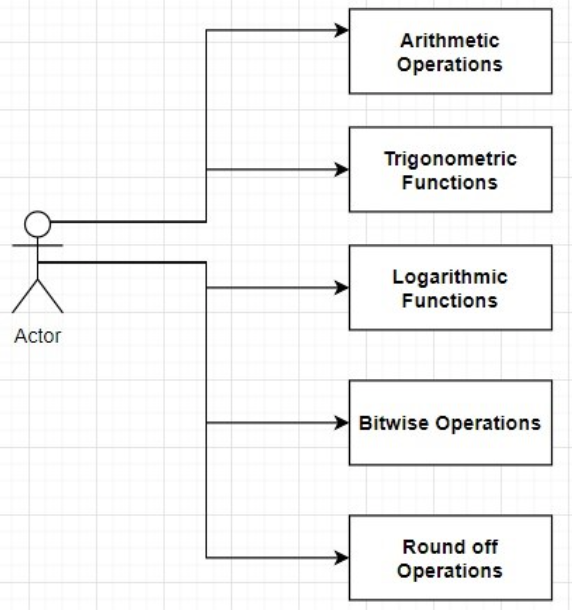
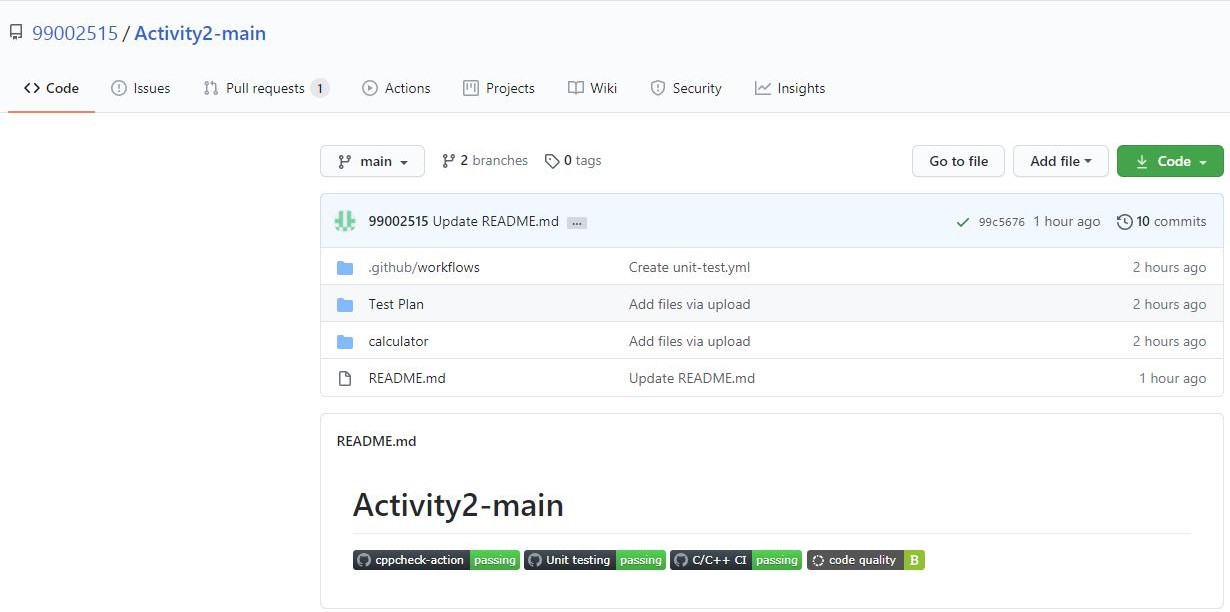


Fig 7: Activity Diagram

GitHub screenshots

Badges



GitHub link - <https://github.com/99002607/Applied_SDLC>

**ACTIVITY – 3 (Agile Implementation)**

**Theme: Calculation**

**Epic 1: Simple math operations.**

User Story1:

* As an accountant.
* I want to add 2 large numbers.
* I want to see the accurate result instantly.

Test Case:

* Given 2 numbers 9055 and 53245.
* When I add them.
* Result should be 62300 within no time.

User Story 2:

* As a Student.
* I need to perform mathematical operations.
* I need accurate and precise results.

Test Case 1:

* Given any number.
* I need explicit option to perform all basic operations.
* And the result should be accurate.

**Epic 2: Ease of Operation**

User Story 1:

* As a Student.
* I want to see all the calculations that I made from when I switched on the calculator.
* So that I can backtrack my calculations.

Test Case 1:

* Given I am in the middle of an operation.
* When I press ‘repeat’ key.
* Then the calculator must display all my previous calculations.

Test Case 2:

* Given that this is my first calculation.
* When I press 'repeat’ key.
* The calculator must display 0.

**Epic 3: Calculating Factorial**

User Stories:

* As an engineering student
* I want to do factorial of a number.

Test Cases:

* Given Inputs for number
* Generate factorial of that.

**Areas of Love and Challenge**

Love

=> New way of implementing same project

Challenge

=> Thinking of different epic was bit difficult

**Activity 4(MakeFile)**

GitHub Link:

* <https://github.com/99002607/Make_File>

References

* [https://www.programiz.com/c-programming/examples/calculator-switch-case](https://www.programiz.com/c-programming/examples/calculator-switch-case" \t "_blank)
* [https://www.tutorialspoint.com/git/index.htm](https://www.tutorialspoint.com/git/index.htm" \t "_blank)
* [https://www.codacy.com/products/how-to-do-code-review](https://www.codacy.com/products/how-to-do-code-review%E2%80%AF" \t "_blank)
* [https://www.cs.colby.edu/maxwell/courses/tutorials/maketutor/](https://www.cs.colby.edu/maxwell/courses/tutorials/maketutor/" \t "_blank)
* [https://www.cs.swarthmore.edu/~newhall/unixhelp/howto\_makefiles.html](https://www.cs.swarthmore.edu/~newhall/unixhelp/howto_makefiles.html" \t "_blank)