GENESIS – Calculator Mini-project Summary Report

./



**Details**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel.**  **No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **To be Approved** | **Remarks/Revision Details** |
| 1 | 7/12/2020 | Apoorva N |  |  |  |
| 2 | 7/12/2020 | Ritesh Prabhu |  |  |  |
| 3 | 7/12/2020 | Sai Kumar |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Contents

[CONTENTS 3](#_bookmark0)

1. [INTRODUCTION](#_bookmark1) [4](#_bookmark1)
2. [REQUIREMENTS](#_bookmark2) [4](#_bookmark2)
3. [THE CALCULATOR WHICH WE HAVE DESIGNED CAN PERFORM](#_bookmark3) [4](#_bookmark3)
4. [SWOT ANALYSIS](#_bookmark4) [5](#_bookmark4)
5. [REQUIREMENTS OF THIS PROJECT](#_bookmark5) [5](#_bookmark5)

[HIGH LEVEL REQUIREMENTS](#_bookmark6) [5](#_bookmark6)

[LOW LEVEL REQUIREMENTS](#_bookmark7) [5](#_bookmark7)

1. [DESIGNS](#_bookmark8) [6](#_bookmark8)

[BEHAVIORAL DIAGRAMS](#_bookmark9) [6](#_bookmark9)

[STRUCTURAL DIAGRAM](#_bookmark10) [7](#_bookmark10)

1. [TEST PLAN](#_bookmark11) [8](#_bookmark11)
2. [GITHUB WORKFLOW 8](#_bookmark12)

LIST OF TABLES

1. AGEING 4
2. COSTING 4
3. HIGH LEVEL INTEGRATED TEST PLAN 8
4. LOW LEVEL UNIT TEST PLAN 9

LIST OF FIGURES

1. SWOT ANALYSIS 5
2. USE CASE DIAGRAM OF CALCULATOR 6
3. SEQUENCE DIAGRAM OF CALCULATOR 6
4. STATE DIAGRAM OF CALCULATOR 7
5. CLASS DIAGRAM OF CALCULATOR 7
6. REPOSITORY VIEW 8
7. VALGRIND 9
8. UNIT TESTING 9
9. CPPCHECK 10

10. C/C++ CI 10

11. CODACY DASHBOARD 11

**Mini Project Calculator**

# Introduction

Calculator is basic yet very important application in today’s life as we cannot carry calculator everywhere, we go every mobile has calculator app in it. Even though the default calculator app in most Android smartphones today is fairly feature packed and capable enough to handle a few complex equations. If you need a scientific calculator that can help you with all types of calculations and math problems, you will have to go for a third-party calculator app.

# Requirements

## Research

|  |  |  |  |
| --- | --- | --- | --- |
|  | Past | Present | Future |
| Ageing | Electronic calculator:  This could perform only basic arithmetic operations. It had seven segment LCD display | Scientific Calculator:  Along with basic arithmetic operations this can perform scientific calculations | Voice enable input or  projecting facility, neural reading input |
|  |  | such as square root, exponential etc… |  |

Table 1. Ageing

|  |  |  |
| --- | --- | --- |
| **Costing** | **Electronic Calculator** | **Scientific Calculator** |
| Rs. 125 (approximate) | Rs. 540 (approximate) |

Table 2. Costing

# The calculator which we have designed can perform

* + Simple calculations like addition, subtraction, multiplication and division and Scientific calculations like factorial, square root, power, log functions, exponential
  + Conversions like celcius to farenheit vice versa and including to kelvin can be done. Measurement calculations like area of square, rectangle and volume of cone, cylinder and BMI can be done

**4W1H Question**

* **What:** A calculator is a small [hand-held computer](https://www.webopedia.com/TERM/H/hand_held_computer.html) that performs mathematical calculations.
* **When:** People do calculations usually use calculators in everyday life to save their time and for accurate answers.
* **Where:** Many times, in scientific and mathematical calculations involving complex operations, the calculation is not feasible because it will take a lot of time and there are many chances of errors when done manually.
* **Why:** To make calculations easier.
* **How:** The memory chips inside the calculator contain thousands or millions of bytes program code that allows the calculator to do work.

# SWOT Analysis

## STRENGTHS WEAKNESS





Environmentally

friendly application. Can perform operations anytime and every time.





Does not perform all operations.

There may be some

incorrect results due to the invalid or wrong

SWOT

ANALYSIS

* Saves human power to calculate manually.
* Innovative technology.
* When some invalid user input is given, it may end the program

**OPPURTUNITIES THREATS**

Figure 1. SWOT ANALYSIS

# Requirements of this project

## High Level Requirements:

* + A mobile calculator app that should perform simple calculations, scientific calculations and conversions
  + The calculator was developed using standard C language and should run on all machines supporting gcc compiler.
  + Should display following menu bar to users like-1. Add, 2. Subtract, 3.Multiply, 4.Divide, 5. Square root, 6. Modulus, 7. Factorial, 8. Temperature, 9. Power, 10.Log, 11.Log10, 12. Exponential, 13.

Even/odd, 14. Area of square, 15. Area of rectangle, 16. Volume of cone, 17. Volume of cylinder,

18. BMI

## Low Level Requirements:

* + - Prevent users from divide by zero error
    - Can use either one or two operands
    - Results of operation should be accurate and has to be displayed on screen

# Designs

## Behavioral diagrams

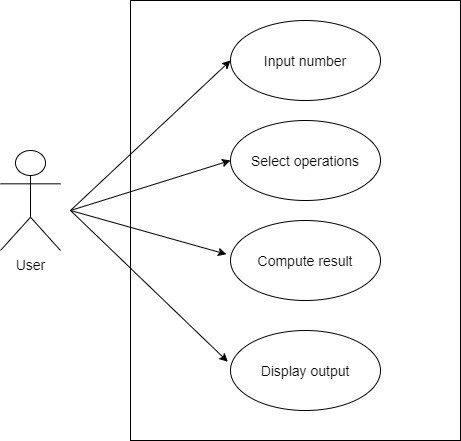
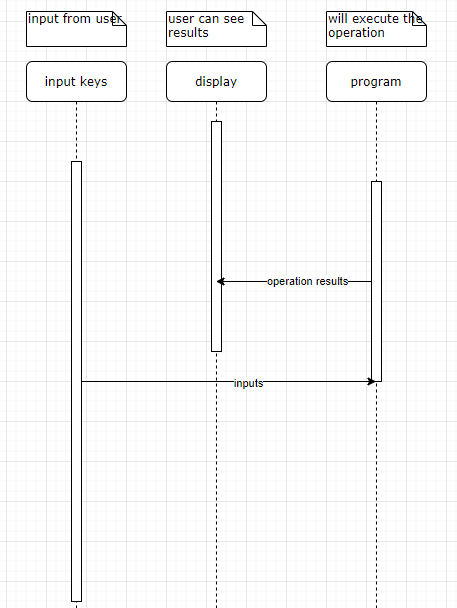


Figure 2. Use case diagram of calculator

Figure 3. Sequence Diagram of calculator

## Structural Diagram

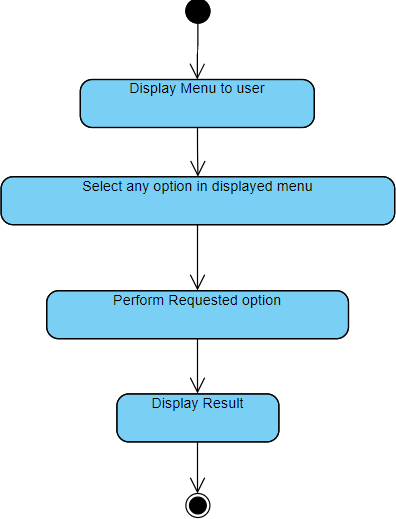


Figure 4. State diagram of calculator

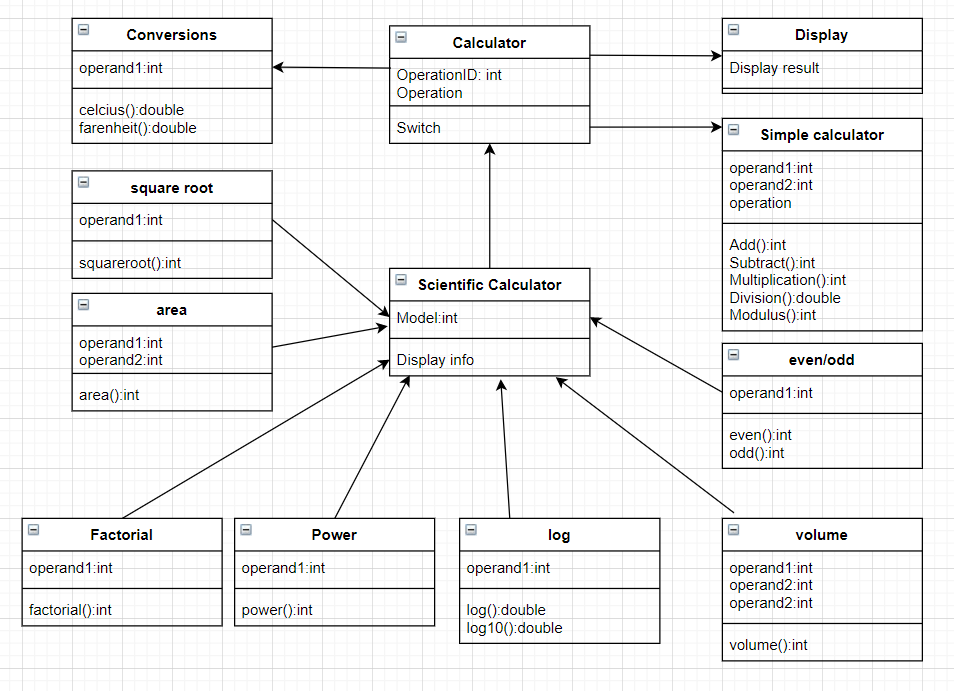


Figure 5. Class diagram of calculator

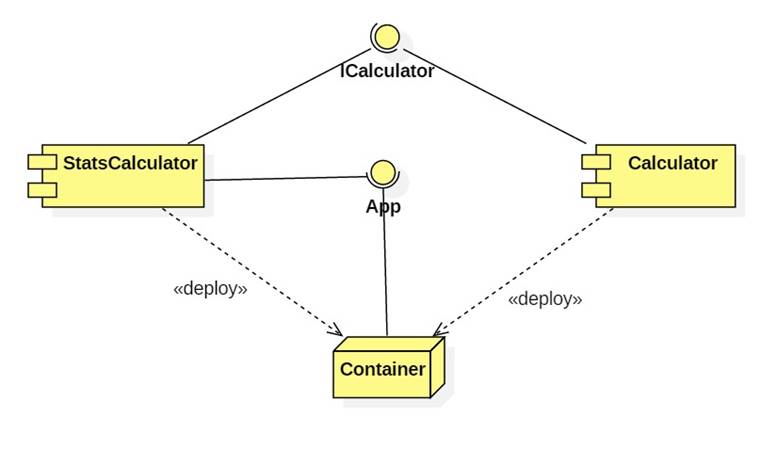


Figure 6. Component diagram of calculator

# Test Plan

**High Level Integrated Test Plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Description** | **Precondition** | **Expected I/P** | **Expected o/p** | **Actual o/p** |
| **1** | Get input method | Accept the input from Key | Correct input | Will show the inputs in display | Displays |
| **2** | Math operation | Correct Calculation | Correct input  symbol as pressed  by user | Will do correct  calculation | calculates |
| **3** | Display | Display output | Proper display of output which  can be interpreted  by human | Will display results clear visibility | Display results |

Table 3. High Level Integrated Test Plan

**Low Level Unit Test Plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test ID** | **Description** | **Expected I/P** | **Expected o/p** | **Actual o/p** | **Result** |
| **1** | Addition | Enter 2 and 3 | 5 | 5 | passed |
| **2** | Subtraction | Enter 5 and 4 | 1 | 1 | passed |
| **3** | Multiplication | Enter 5 and 6 | 30 | 30 | passed |
| **4** | Division | Enter 6 and 2 | 3 | 3 | passed |
| **5** | Modulus | Enter 8 and 4 | 0 | 0 | passed |
| **6** | Factorial | Enter 5 | 120 | 120 | passed |
| **7** | Area of square | Enter 2 | 2 | 2 | passed |
| **8** | Area of rectangle | Enter 4 and 5 | 20 | 20 | passed |
| **9** | Volume of cone | Enter 5 and 6 | 82.89 | 82.89 | passed |
| **10** | Volume of cylinder | Enter 5 and 6 | 471.00 | 471.0 | passed |
| **11** | BMI | Enter 80 and 1.83 | 23.88 | 23.88 | passed |
| **12** | Temperature conversion  (F to C vice versa) | Enter 100 | 37.77 | 37.77 | Passed |
| **13** | Temperature conversion  (K to F vice versa) | Enter 298.706 | 78 | 78 | passed |
| **13** | Power | Enter 5 and 2 | 25 | 25 | passed |
| **14** | Log10 value | Enter 6 | 0.778 | 0.778 | passed |
| **15** | Log value | Enter 25 | 3.218 | 3.218 | passed |
| **16** | Exponential | Enter 4.5 | 90.01 | 90.01 | passed |
| **17** | Even/Odd | Enter 4 | 0 (even) | 0 | passed |

## GitHub workflow:

Link: <https://github.com/99003187/Scientific_Calculator>

1. Repository view

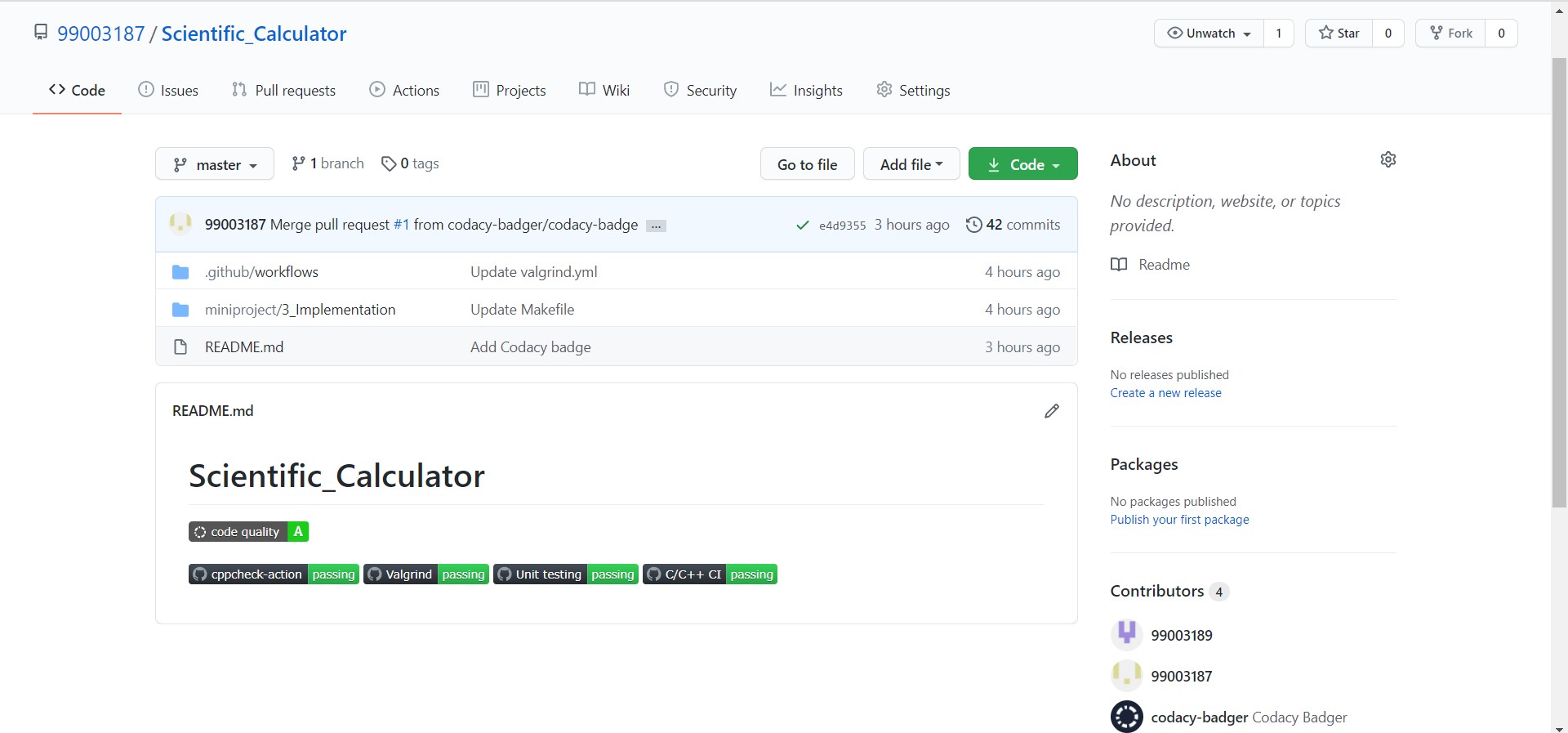


Figure 7. Repository view

1. Valgrind

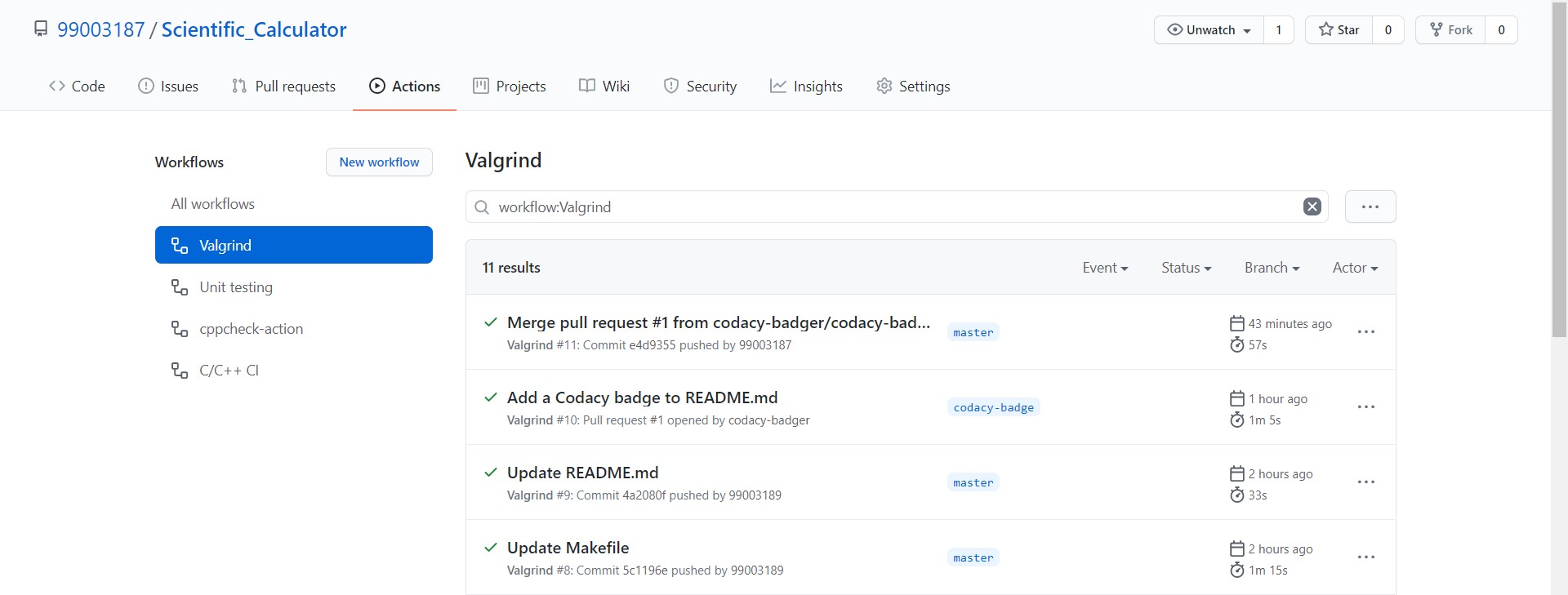
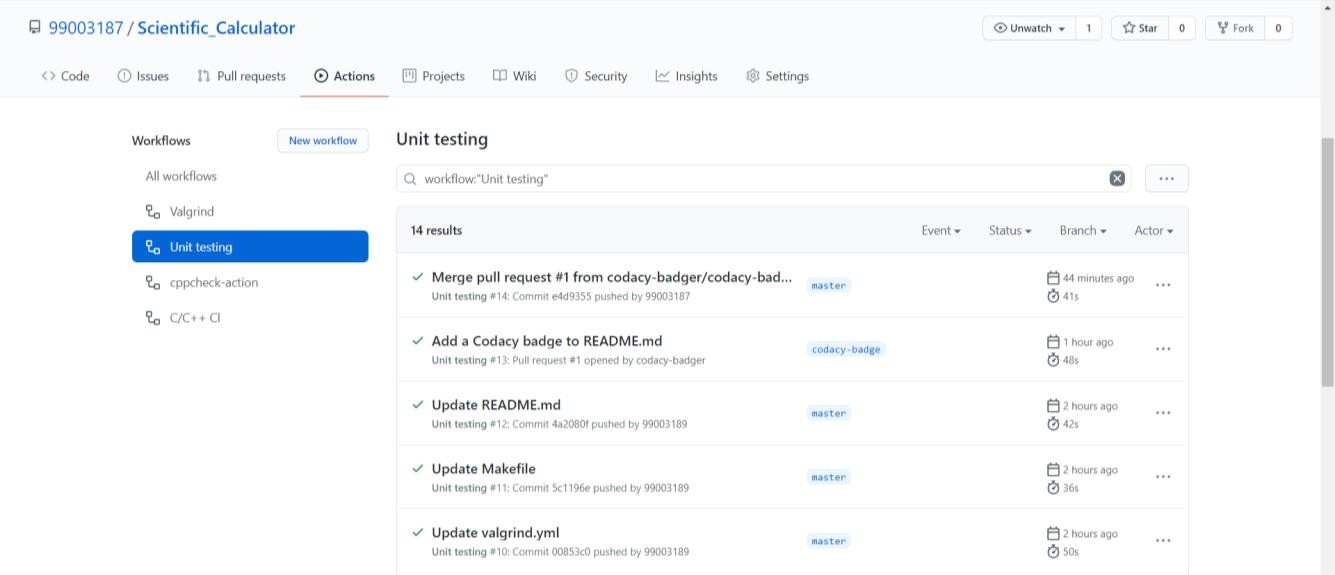
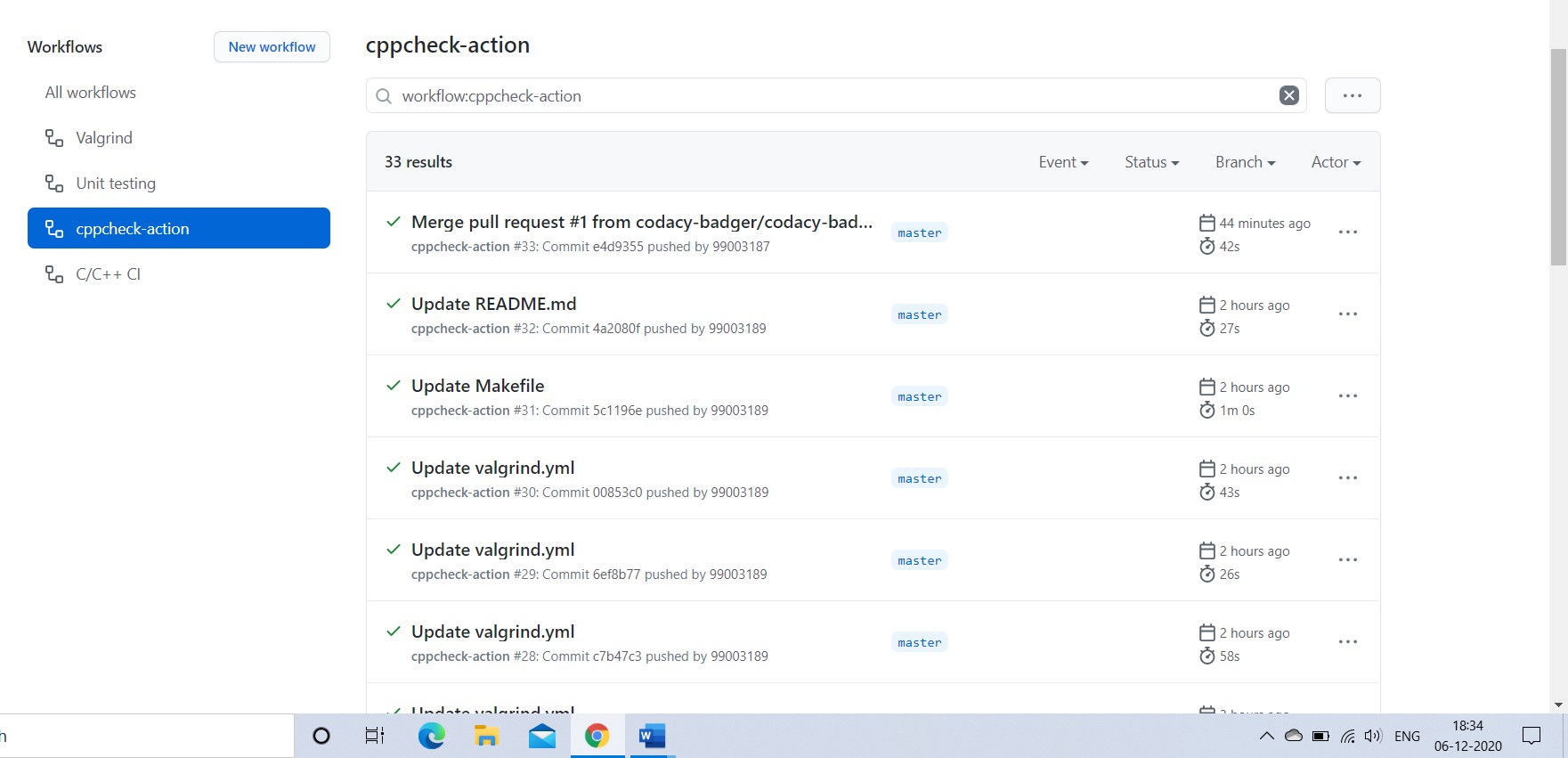


Figure 8. Valgrind

1. Unit Testing



Figure 9. Unit Testing

1. cppcheck

Figure 10. cppcheck

1. C/C++ CI

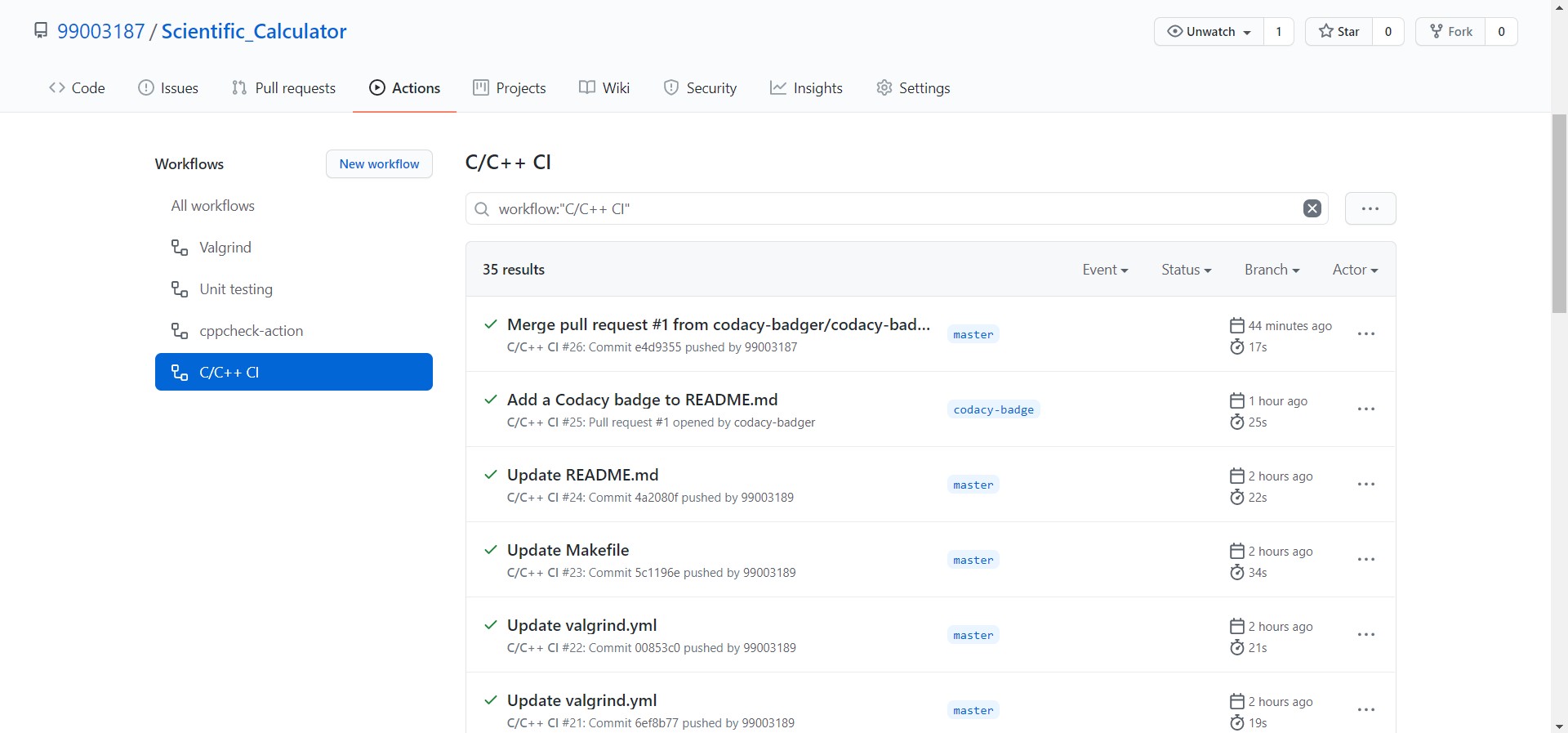


Figure 11. C/C++ CI

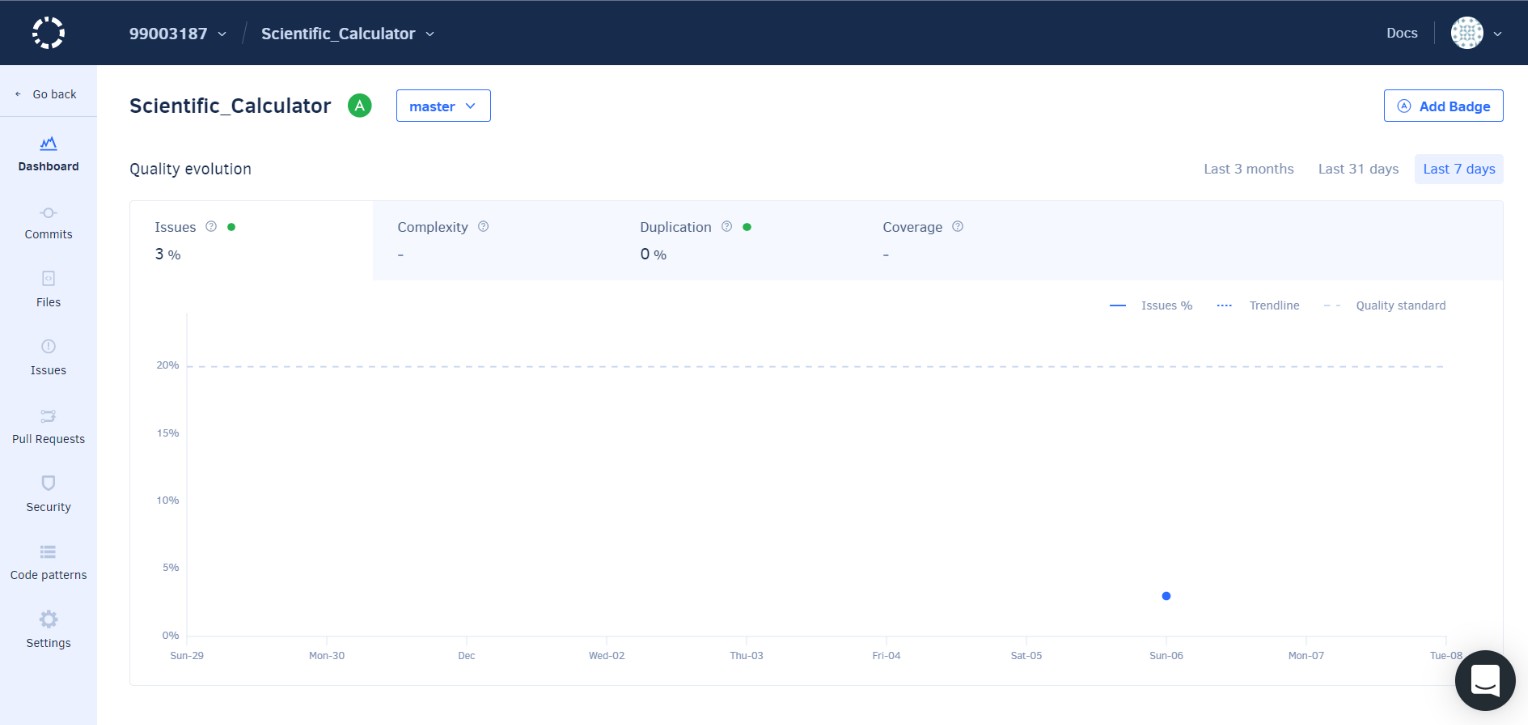
1. Codacy dashboard

Figure 12. Codacy Dashboard