./

Learning Report – Carry Speculative Adder

Course Code: <CODE>



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **Approved By** | **Remarks/Revision Details** |
| 1. | 16-09-20 | Aditya |  |  |  |
| 2. | 17-09-20 | Aditya |  |  |  |
| 3. | 18-09-20 | Aditya |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Document History**

# 

Contents-CHECKLIST 3

ACTIVITY AND TASKS 4

ACTIVITY 1– SYSTEM/SOFTWARE DEVELOPMENT 6

ACTIVITY 2 – AGILE ASPECTS 17

ACTIVITY 3 –CI WORKFLOW FOR C PROGRAMMING 15

# Activity and Tasks

## **Activity 1**– System/Software Development---------------------------------------------5

* Lists of tables
* Lists of figures
* Requirements
* SWOT Analysis
* UML Diagrams
* Test Plan
* References

## **Activity 2**– Agile Aspects ---------------------------------------------------------------------16

* Agile Methodology
* Companies which use Agile
* User Story-I
* User Story-II
* User Story-III
* References

## **Activity 3** –CI Workflow for C Programming ---------------------------------------------22

* Requirements
* Requirements Plan
* UML Diagram
* Test Plan

**Activity-1:**

**List of Tables:**

|  |  |
| --- | --- |
| **Name** | **Page no:** |
| Comparison of different types of adders | 1 |
| Cost | 1 |
| Analysis | 2 |
| Mapping requirements | 2 |
| Test Plan | 3 |

**List of Figures:**

|  |  |
| --- | --- |
| **Name** | **Page no:** |
| Low level Activity diagram | 4 |
| High level Activity diagram | 4 |
| Low level State diagram | 5 |
| High level State diagram | 5 |
| Low level Deployment Diagram | 6 |
| High level Deployment Diagram | 6 |
| Low level Component diagram | 7 |
| High level Component diagram | 7 |

**Objective:** To increase the speed of a carry speculative adder

**Requirements:**

**High level requirements:**

1. XILINX ISE Tool
2. Cadence Genus
3. Spartan 3 board
4. Adders ,Carry predictor and generator

**Low Level requirements:**

1. Enable pins
2. Multiplexers
3. Linux OS

**Comparison of different types of adders:**

|  |  |  |
| --- | --- | --- |
| Type of adder | Time of execution | Size of adder |
| Ripple carry adder | 321.3ns | 32 Bit |
| Carry select adder | 140.7ns | 32 Bit |
| CSPA | 50.3ns | 32 bit |
| CSPA with modified carry generators | 21.7ns | 32 bit |

**Cost:**

|  |  |
| --- | --- |
| **Equipment** | **Cost** |
| Spartan 3 board | 12,000 |
| XILINX ISE 14.0 | 56,000 |
| Cadence Genus | 60,000 |
| Total Cost | 1,28,000 |

**SWOT Analysis:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Strengths** | **Weakness** | **Opportunities** | **Threats** |
| 1. Better speed of execution.  2**.** Error free output.  Overall size of the circuit decreases because the modified carry generators when compared to CSPA(Carry Speculative Adder | 1. When compared to the traditional ripple carry adder, overall size of the circuit is increased as there are additional blocks for prediction and generation of the carry.  2. Power consumption is high when compared to a traditional Adder. | 1. Can be used in image processing because tedious multiplications are done in processing of the image.  2. Can also be used in an SOC. | 1. Will need more space on the PCB because of the additional blocks fabricated.  2.Power consumption when compared to the traditional adder is more. |

**Requirement table:**

|  |  |
| --- | --- |
| ID | Description |
| H\_01 | Xilinx ISE is a software tool produced by Xilinx for synthesis and analysis of HDL designs. |
| H\_02 | The Genus Synthesis Solution is a next-generation RTL synthesis and physical synthesis tool. |
| H\_03 | The **Spartan** **3** Starter **Board** provides a powerful, self-contained development platform for designs targeting the **Spartan**-**3** FPGA from Xilinx. |
| H\_01\_L\_01 | These enable pins are used to turn on or turn off the circuit. |
| H\_04\_L\_01 | The adders in the circuit will work according to the inputs given by the enable pins. |
| H\_04\_L\_02 | The multiplexers output is needed to select the correct value of sum as there are 2 sum blocks in the circuit. |
| H\_02\_L\_03 | Cadence Genus only runs on Linux operating system. |

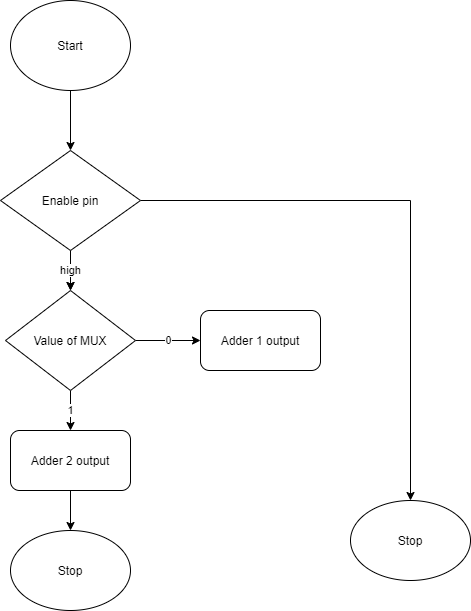
**Test Plan:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Description | Precondition | Expected Input | Expected Output | Actual Output |
| H\_04 | Adders | 0 | 123 | 123 | 123 |
| L\_02 | MUX | 0 | Sum1,Sum2,selection line | Sum1 | Sum1 |
| L\_02 | MUX | 1 | Sum1,Sum2,selection line | Sum2 | Sum2 |
| L\_01 | Enable pins | 0 | Any input | 0 | 0 |
| L\_02 | Enable pins | 1 | 180,220 | 400 | 400 |
| H\_04 | Error detection | 0 | 180,100 | 280 | 280 |
| H\_04 | Error detection | 1 | 180,100 | 280 | 280 |
| H\_01\_L\_02 | Error correction | 0 | 150 | 0 | 0 |
| H\_01\_L\_02 | Error correction | 1 | 150 | 150 | 150 |
| H\_04 | Carry predictor | 0 | 0 | 0 | 0 |
| H\_04 | Carry predictor | 1 | 1 | 1 | 1 |
| H\_04 | Internal carry generator | 0 | 1 | 1 | 1 |
| H\_04 | Internal carry generator | 0 | 0 | 0 | 0 |
| H\_04 | Internal carry generator | 1 | 0 | 0 | 0 |
| H\_04\_L\_01 | Adder\_Enable | 0 | Any Input | 0 | 0 |

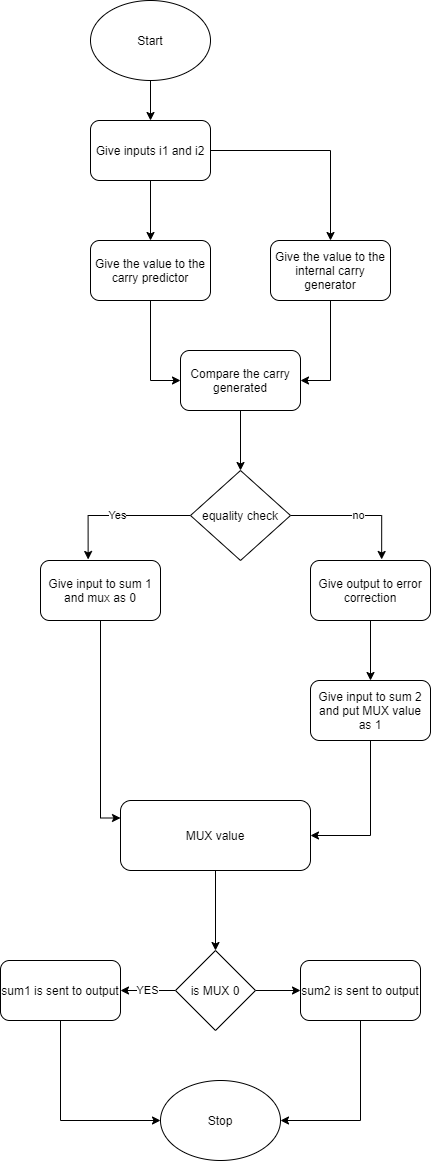
**Behavioural Model:**

**Activity Diagram:**

1. **Low level Activity diagram:**

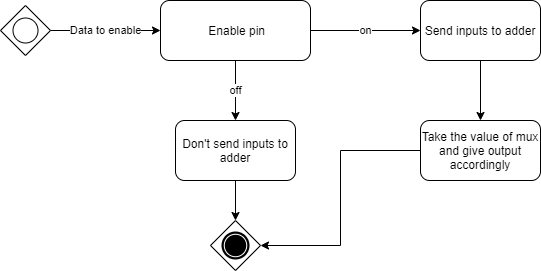
****

1. **High level activity diagram:**

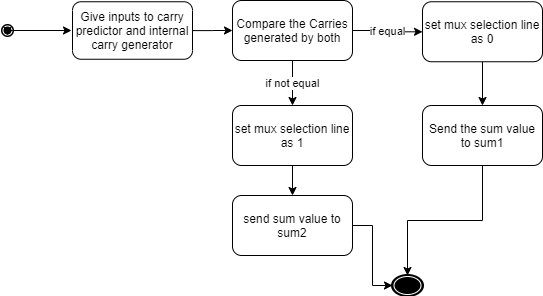
****

**State Chart diagram:**

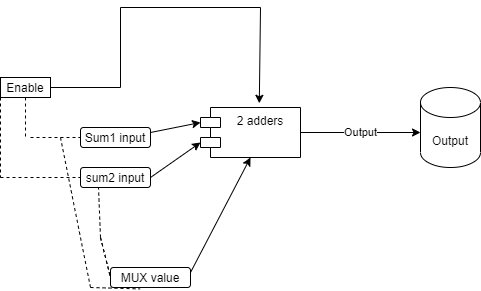
**1. Low level state diagram:**

****

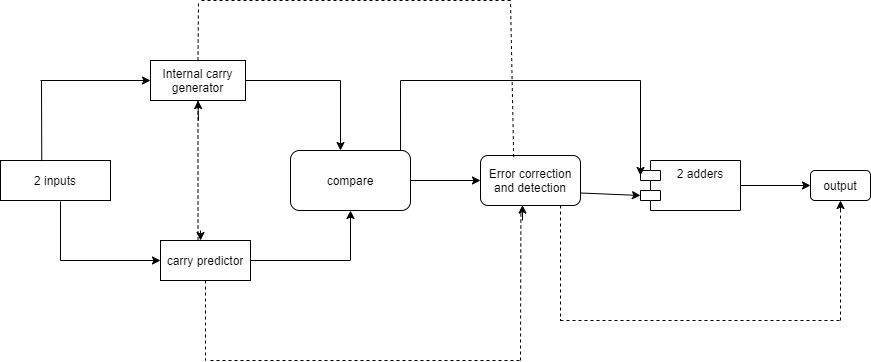
**2. High level state diagram:**

****

**Structural Diagram:  
1.Deployment Diagram:**

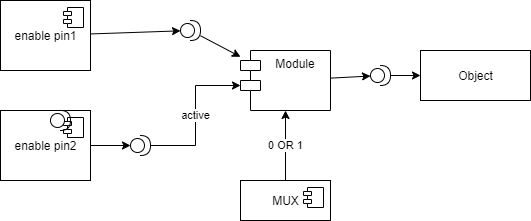
**Low level Deployment Diagram:**

**High level Deployment Diagram:**

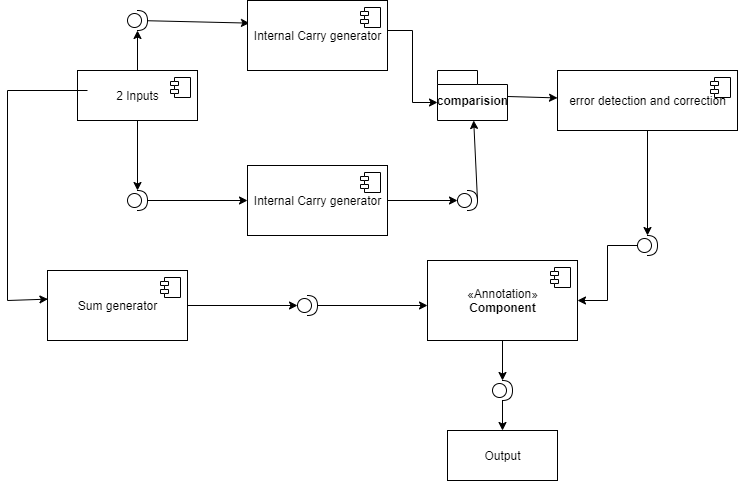
****

**Component Diagram:**

**Low level Component diagram:**

****

**High level Component diagram:**

****

**Test plan for High level requirement:**

1. **Requirement based:**

As the above project is a digital adder, the requirement would be 2 integers which are used for addition. The addition done is a binary addition so by adding 1 and 1 will result to sum as 0 and carry 1. This carry is propagated to the next binary digit. The adder must result the correct output of various positive integers.

1. **Scenario based:**

As the inputs to the adder are dynamic, all the outputs should be error free and should be generated as soon as possible to generate a quick output.

1. **Boundary Conditions:**

The boundary conditions for this adder are, both the inputs are 0, which gives the sum 0 and carry while carry generated by MSB is 1 while addition, in which the carry out will be ‘1’.

**Test Plan for Low level requirements**:

1. **Requirement based:**

To put the circuit always working, both the enable pins should be high such that the circuit gives a proper output and the MUX is used to select the output from the 2 sum blocks present .The values possible for the enable pin and MUX are 0 or 1.

1. **Scenario based:**

To get the correct sum output the enable pin should always be high and the mux input which selects the correct output from the sum blocks.

1. **Boundary Conditions:**

Boundary conditions for the MUX are 0 or 1

Boundary conditions for enable pin are 0 or 1.

**References:**

1. Ing-Chao Lin, Senior Member, IEEE, Yi-Ming Yang, and Cheng-Chian Lin, “High-Performance Low-Power Carry Speculative Addition With Variable Latency”, IEEE transactions on very large scale integration (vlsi) systems,2014

2.Gai Liu, Ye Tao, Mingxing Tan, and ZhiruZhang,Computer Systems Laboratory, Electrical and Computer Engineering, “CASA: Correlation-Aware Speculative Adders”,2014

3.Junjun Hu and WeikangQian, Shanghai “A New Approximate Adder with Low Relative Error and Correct Sign Calculation”, 2014

4. V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, “Low-power digital signal processing using approximate adders,” IEEE Trans. Comput.- Aided Design Integr. Circuits Syst., vol. 32, no. 1, pp. 124–137, Jan. 2013.

5. A.B. Kahng and S. Kang, “Accuracy-configurable adder for approximate arithmetic designs,” in Proc. Design Autom. Conf. (DAC), 2012, pp. 820–825.

6.K. Du, P. Varman, and K. Mohanram, “High performance reliable variable latency carry select addition,” in Proc. Design Autom. Test Eur. Conf. Exhibit. (DATE), Mar. 2012, pp. 1257–1262.

7. N. V. Mujadiya, “Instruction scheduling on variable latency functional units of VLIW processors,” in Proc. Int. Symp. Electron. Syst. Design (ISED), 2011, pp. 307–312.

**Activity-2:**

**Agile methodology:**

**Agile methodology:**

The Agile Method and methodology is a particular approach to project management that is utilized in software development. This method assists teams in responding to the unpredictability of constructing software. It uses incremental, iterative work sequences that are commonly known as sprints.

A sprint is a period of time allocated for a particular phase of a project. Sprints are considered to be complete when the time period expires. There may be disagreements among the members of the team as to whether or not the development is satisfactory; however, there will be no more work on that particular phase of the project. The remaining phases of the project will continue to develop within their respective time frames.

Scrum is a subset of Agile. It is a lightweight process framework for agile development, and the most widely-used one.

* A “process framework” is a particular set of practices that must be followed in order for a process to be consistent with the framework. (For example, the Scrum process framework requires the use of development cycles called Sprints, the XP framework requires pair programming, and so forth.)
* “Lightweight” means that the overhead of the process is kept as small as possible, to maximize the amount of productive time available for getting useful work done.

**Companies that Use the Agile Method**

Although there is no official list of companies that use the Agile Method for their projects, IBM is one of the companies that openly uses this method to develop software. Many companies will adopt the use of this method within their development structure, but they aren’t always open about their choice to use it.

According to IBM, the use of the Agile Method means that significant organizational changes will take place. They believe that many Agile software development teams will increase their chances of success by partnering with a trusted guide. They help clients implement their own Agile software development strategies for their projects. They provide critical guidance that will help Agile software development teams to avoid common adoption, expansion, and implementation pitfalls.

**User story-1:**

**Development of webpages:**

Let us consider that we are initially making a webpage which collects name and phone number of a person and store it into a database. As the agile model is customer centric, we will release this web page for the customers to use.

**Epic-1:**

The information and feedback given by the customers is taken very seriously as they are the main users of this product. This web page is released into the market and feedback is collected, this will be the end of the cycle1. -5 Days

At the end of epic-1:

1. Webpage is created.
2. 2 columns data is noted.

**Epic-2:**

Now, the main important aspect of the product is updates. So, a new update is released into the website. Let’s say a new column is added into the data base as address and is released into the market. If any bugs are found, they will be corrected.-1 Week

At the end of epic-2:

1. According to the feedback, the bugs are corrected.
2. New feature is added and is rolled out for customers to try.

**Epic-3:**

Now, other column is added to collect hobbies and is released this cycle continuous for many iterations. -5 Days

At the end of Epic-3:

1. According to the feedback, the bugs are corrected.
2. New feature is added and is rolled out for customers to try that is hobbies column.

This cycle will continue for further more epics.

**User story-2:**

**Instant Messaging app:**

**Epic-1: Creating an app which can collect phone number, name and has an ability to message people in the same app.** -5 Days

Let us consider that we are initially making a Messaging app which collects phone number of a person and is used to message to the person using the same app. As the agile model is customer centric, we will release this app for the customers to use.

At the end of epic-1:

1. App is created.
2. Phone number is collected and ability to message is tested.

**Epic-2: Addition of voice call.-1Week**

Now, the main important aspect of the product is updates. So, a new update is released into the website. Let’s say a new feature is added into the app let’s say an audio call and is released into the market. If any bugs are found, they will be corrected.

At the end of epic-2:

1. According to the feedback, the bugs are corrected.
2. New feature is added that is a voice call and is rolled out for customers to try.

**Epic-3: Addition of video call-3Week**

For the next update a new feature of video calling is added and is released to customers. As stated above the customer plays a crucial role for using the products. According to the feedback, the new features will be added.

**At the end of Epic-3:**

1. According to the feedback, the bugs are corrected.
2. Video calling feature is rolled out for customers to try.

**User story-3:**

**Software updates in our gadgets: 1Week**

**Epic-1:Creating a contact with name and phone number-2 Weeks**

Let us consider that we are initially making a feature which collects name and phone number of a person and store it into a contact list. When ever we should make a call, the whole contact list should be scrolled for finding the contact.

At the end of epic-1:

1. Contact data base is created.
2. Name and his phone number must be saved. This ability is tested and is released into the market.

**Epic-2: Adding email option and search button.-1Week**

The information and feedback given by the customers is taken very seriously as they are the main users of this product. A small search button is added and an option for adding an email is also given to the contact list which makes the calling easy and the update is rolled out to customers.

**At the end of epic-2:**

1. Contact data base is created and the option of email is tested.
2. A search button is added to the contact list.

**Epic-3:**

Now instead of using the buttons a direct voice assist is made and given out such that a voice from the user can trigger a call to the contact. This is one of the example which we see in our daily life.

**At the end of epic-3: 3-weeks**

Voice assistant is enabled to make a call.

Agile methodology is an approach used for the development of a project which helps to respond to the unpredictability of building software through incremental, iterative work cadences. These methodologies are used to deal with the situations where the waterfall model fails. The biggest drawback of waterfall model is that it assumes that every requirement of the project can be identified before any design or coding occurs.

**References:**

1. [**https://www.atlassian.com/agile/project-management/program**](https://www.atlassian.com/agile/project-management/program)
2. [**https://www.tutorialspoint.com/sdlc/sdlc\_agile\_model.htm**](https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm)
3. [**https://www.guru99.com/agile-scrum-extreme-testing.html**](https://www.guru99.com/agile-scrum-extreme-testing.html)
4. [**https://en.wikipedia.org/wiki/Agile\_software\_development**](https://en.wikipedia.org/wiki/Agile_software_development)

**Activity-3:** **CI Workflow for C Programming**

**Requirements:**

**High level requirements:**

1. Operator1, Operator2
2. Addition
3. Subtract
4. Multiplication
5. Division
6. Percentage

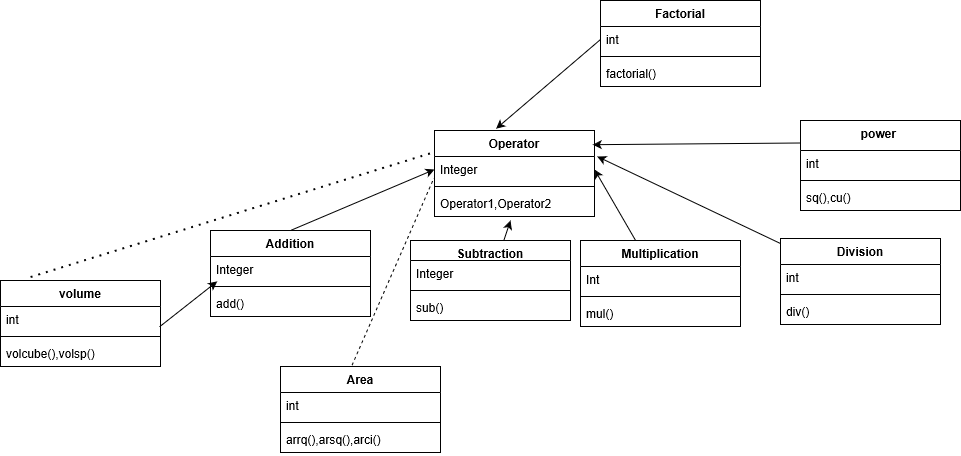
**Low level requirements:**

1. Prime
2. Perimeter of square
3. Perimeter of rectangle
4. Perimeter of circle
5. Area of square
6. Area of rectangle
7. Area of circle
8. Factorial
9. Armstrong number
10. Square of a number
11. Cube of a number
12. Reverse of a number
13. Volume of a cube
14. Volume of a sphere

**Requirement Table:**

|  |  |
| --- | --- |
| **ID** | **Description** |
| H\_01 | Operator which is used to do the operation |
| H\_02 | Performs Addition operation |
| H\_03 | Performs Subtraction operation |
| H\_04 | Performs Multiplication operation |
| H\_05 | Performs Division operation |
| H\_06 | Performs Percentage operation |
| H\_01\_L\_01 | Prime number operation |
| H\_02\_L\_01 | Prime number operation |
| H\_01\_L\_02 | Perimeter of square |
| H\_01\_L\_03 | Perimeter of rectangle |
| H\_01\_L\_04 | Perimeter of circle |
| H\_01\_L\_05 | Area of Square |
| H\_01\_L\_06 | Area of rectangle |
| H\_01\_L\_07 | Area of circle |
| H\_01\_L\_08 | Factorial |
| H\_01\_L\_09 | Armstrong number |
| H\_01\_L\_10 | Square of a number |
| H\_01\_L\_11 | Cube of a number |
| H\_01\_L\_12 | Reverse of a number |
| H\_01\_L\_13 | Volume of a cube |
| H\_01\_L\_14 | Volume of a Sphere |
| H\_02\_L\_02 | Perimeter of square |
| H\_02\_L\_03 | Perimeter of rectangle |
| H\_02\_L\_04 | Perimeter of circle |
| H\_02\_L\_06 | Area of rectangle |
| H\_03\_L\_05 | Area of square |
| H\_03\_L\_06 | Area of Rectangle |
| H\_03\_L\_07 | Area of circle |
| H\_03\_L\_08 | Factorial |
| H\_03\_L\_013 | Volume of a cube |
| H\_04\_L\_014 | Volume of a Sphere |

**Calculator UML Diagram:**

****

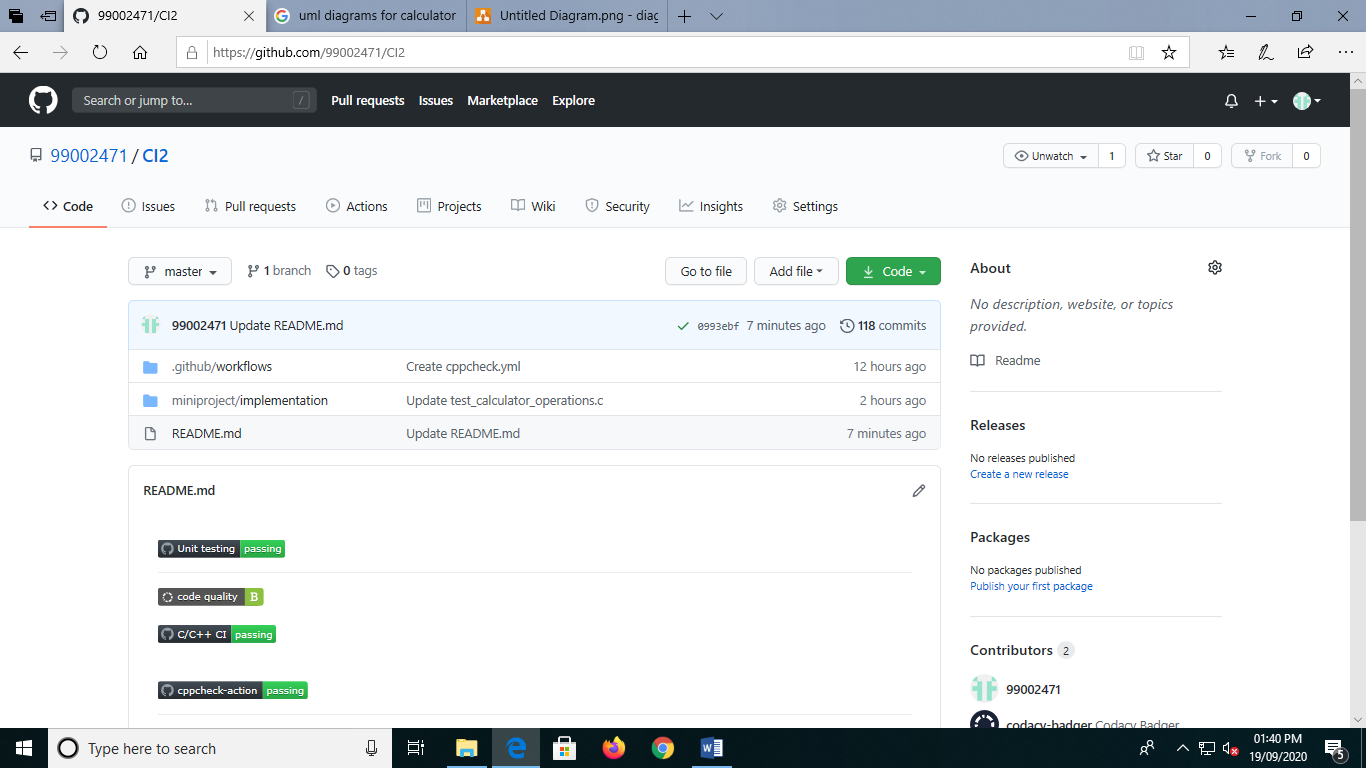
**Test Plan:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Description** | **Precondition** | **Expected input** | **Expected output** | **Actual output** |
| H\_02 | Addition | 0 | 12,22 | 34 | 34 |
| H\_03 | Subtraction | 0 | 11,1 | 10 | 10 |
| H\_04 | Multiplication | 1 | 2,5 | 10 | 10 |
| H\_05 | Division | 1 | 10,2 | 5 | 5 |
| H\_06 | Percentage | 0 | 19,100 | 19 | 19 |
| H\_01\_L\_01 | Prime number | 0 | 25 | -1 | -1 |
| H\_01\_L\_02 | Perimeter of square | 0 | 6 | 24 | 24 |
| H\_01\_L\_03 | Perimeter of rectangle | 0 | 4,5 | 18 | 18 |
| H\_01\_L\_04 | Perimeter of circle | 0 | 2 | 12 | 12 |
| H\_01\_L\_05 | Area of Square | 0 | 4 | 16 | 16 |
| H\_01\_L\_06 | Area of rectangle | 0 | 3,1 | 3 | 3 |
| H\_01\_L\_07 | Area of circle | 0 | 2 | 12 | 12 |
| H\_01\_L\_08 | Factorial | 1 | 5 | 120 | 120 |
| H\_01\_L\_09 | Armstrong number | 0 | 375 | 1 | 1 |
| H\_01\_L\_10 | Square of a number | 1 | 5 | 25 | 25 |
| H\_01\_L\_11 | Cube of a number | 1 | 3 | 27 | 27 |
| H\_01\_L\_12 | Reverse of a number | 1 | 123 | 321 | 321 |
| H\_01\_L\_13 | Volume of a cube | 0 | 5 | 125 | 125 |
| H\_01\_L\_14 | Volume of a Sphere | 0 | 2 | 32 | 32 |

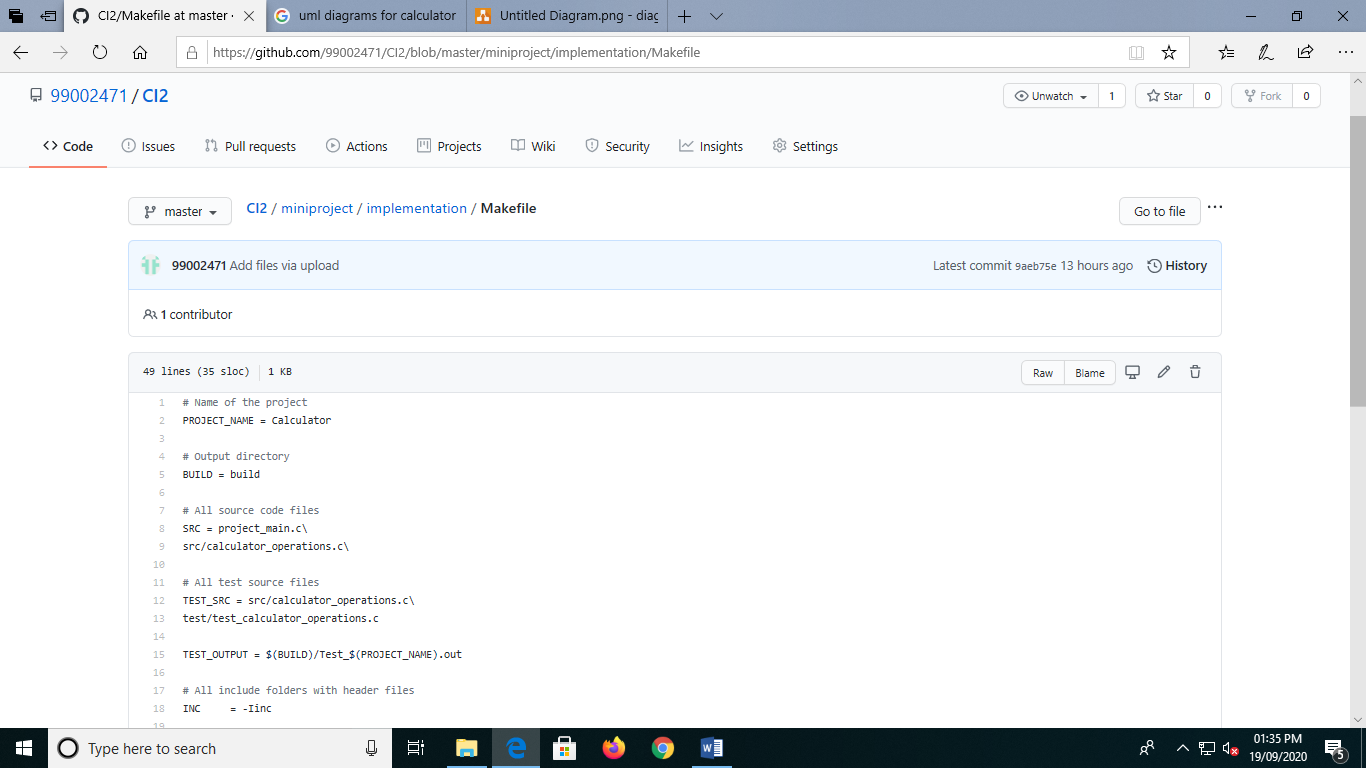
**Reports:**

GIT Repository link: https://github.com/Stepin104341/aditya104341

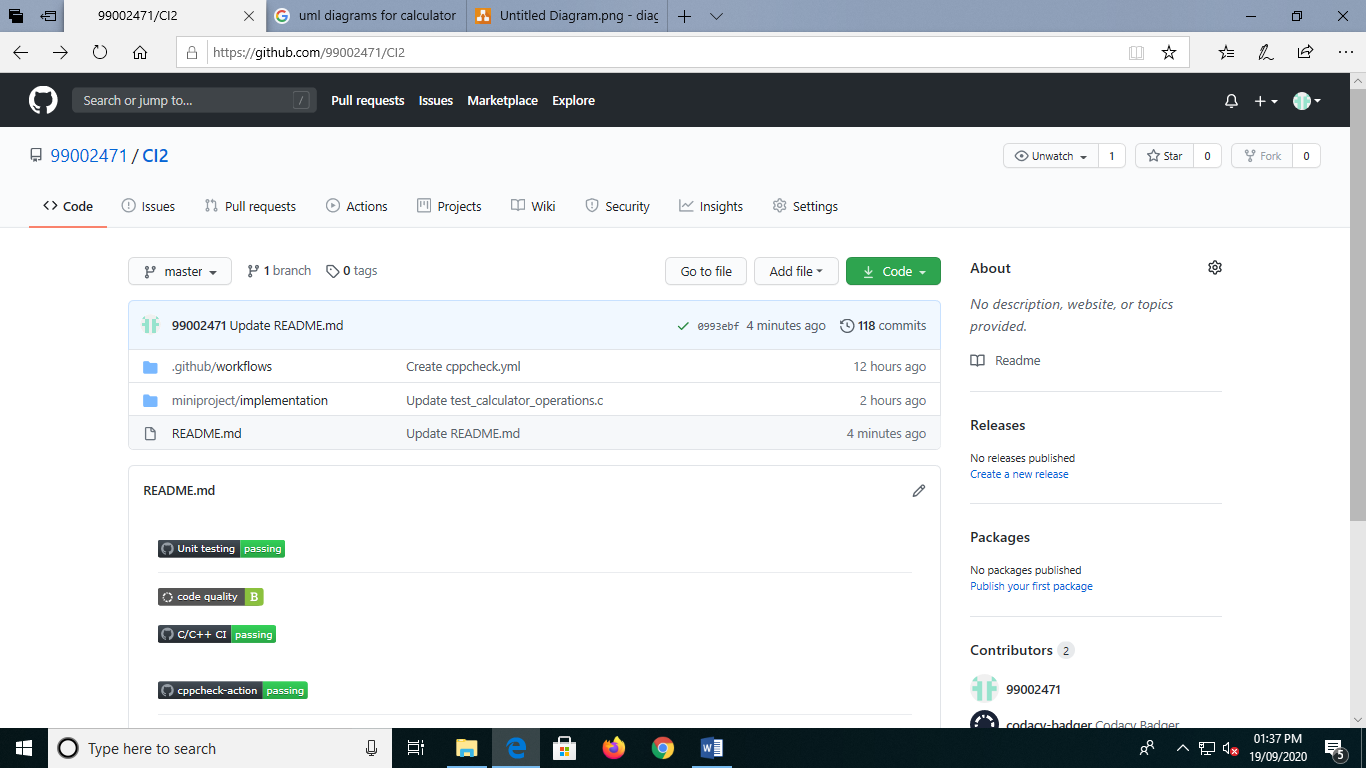
**GIT Repository:**



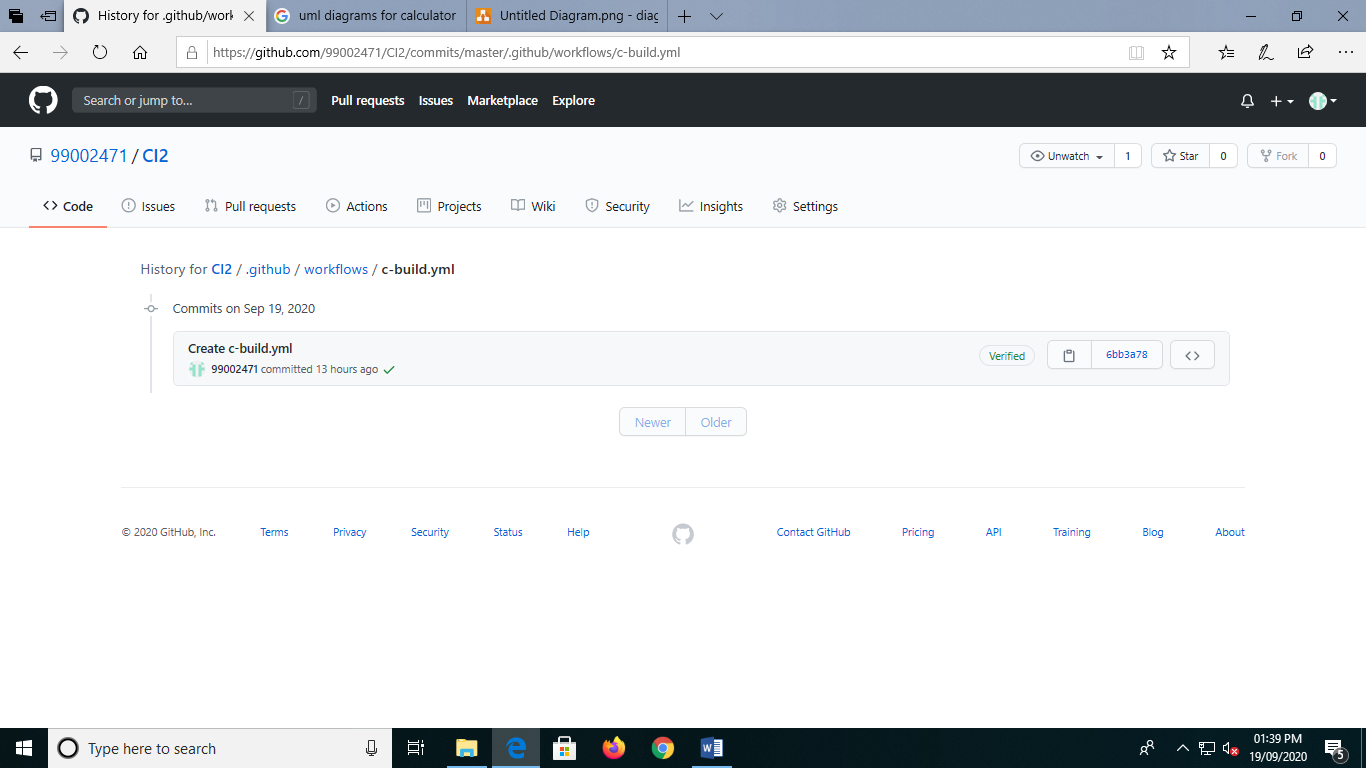
**Make File:**



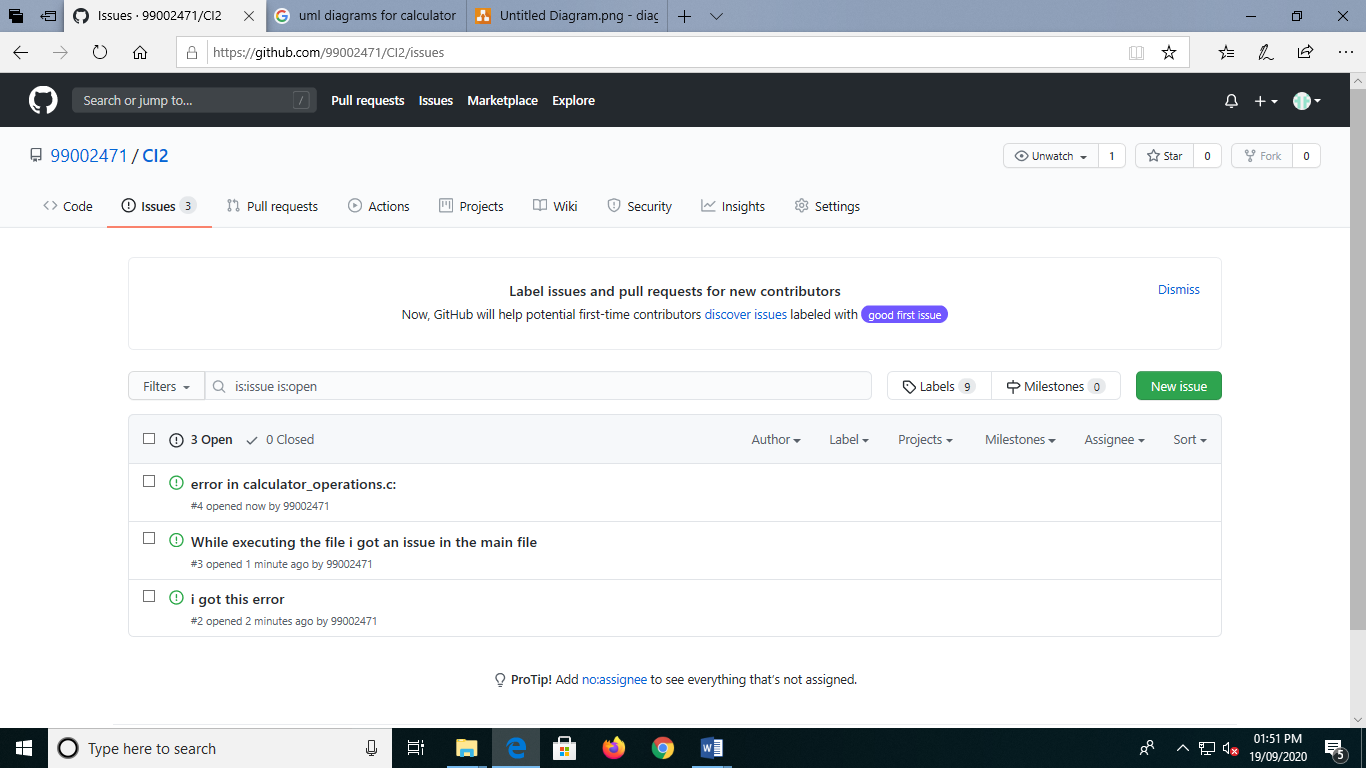
**Commit History:**



**C-Build:**



**Issues:**

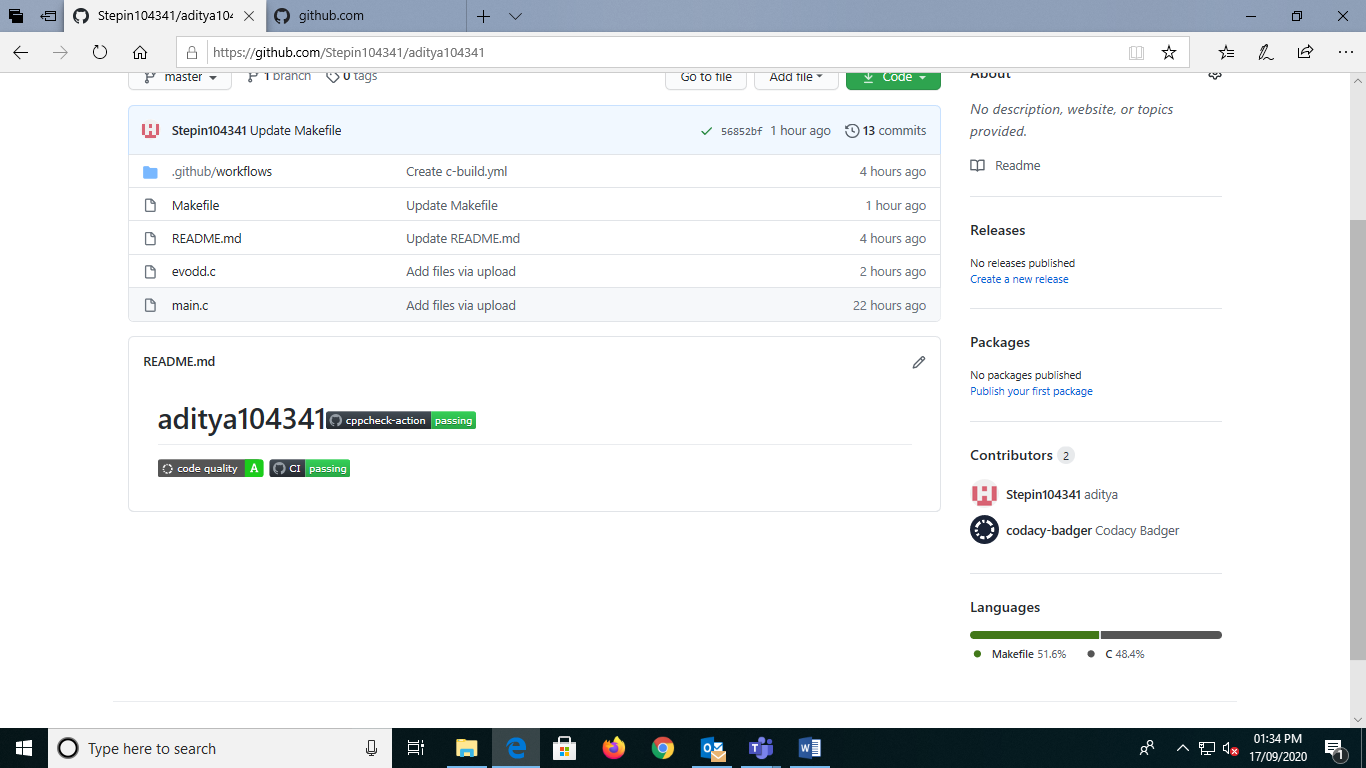


**Appendix**:

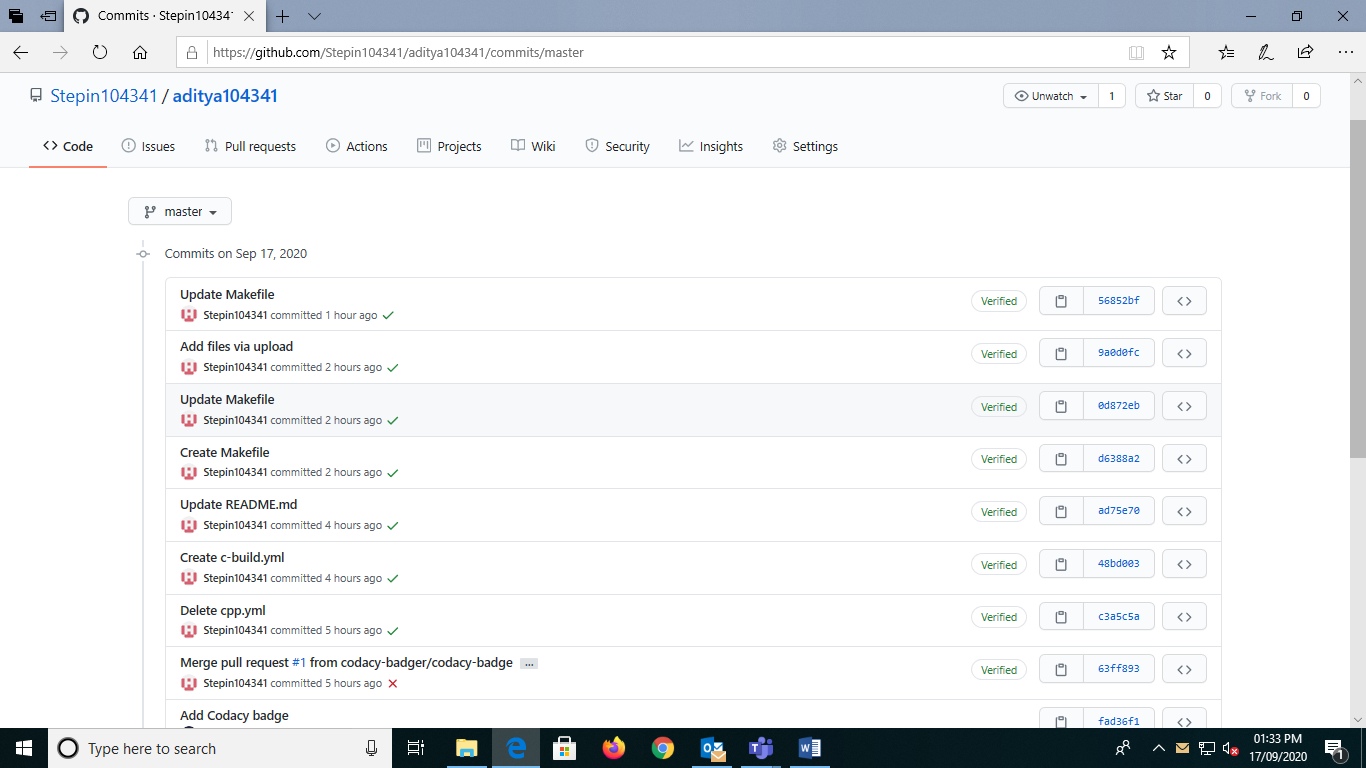
GIT Repository link: <https://github.com/99002471/CI2>

**Old Repository screenshots:**

**Code quality and Batches:**



**Commits:**



**Actions:**

