Full Stack Project Report

(2020-2021)

GIFFY



**Institute of Engineering and Technology**

Submitted by :

Tushar Saxena (181599762)

Vipul (181500801)

Umesh Pratap Singh (181500767)

**Supervised By:**

**Mr. Pankaj Kapoor**

**Technical Trainer**

**Department of Computer Engineering and Application**

****

**Department of computer Engineering and Applications**

**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,**

**Mathura – 281406**

**Declaration**

We here by declare that the work which is being presented in the Full Stack Project **“**GIFFY**”,** in partial full fillment of the requirements for FullStack Project viva voce, is an authentic record of our own work carried under the supervision of **Mr. Pankaj Kapoor, Assistant Professor, GLA University, Mathura.**

Tushar Saxena (181599762)

Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_

Vipul (181500801)

Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_

Umesh Pratap Singh (181500767)

Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_

Course: B. Tech (CSE)

Year: 3rd

Semester: V

Members Github Id’s:

* <https://github.com/vipulgupta22>
* <https://github.com/tusharsaxena3112>
* <https://github.com/thakurups>



**Department of computer Engineering and Applications**

**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,**

**Mathura – 281406**

**Certificate**

This is to certify that the project entitled “GIFFY” carried out in Full Stack Project is the work done by Tushar Saxena , Vipul , Umesh Pratap Singh and is submitted in partial full fillment of the requirements for the award of degree Bachelor of Technology (Computer Science and Engineering).

Signature of Supervisor:

Name of Supervisor: Mr. Pankaj Kapoor

Date:

**Acknowledgement**

It is our pleasure to acknowledge the assistance of a number of people without whose help this project would not have been possible.

First and foremost, We I would like to express our gratitude to **Mr. Pankaj Kapoor** our project mentor, for providing invaluable Encouragement, guidance and assistance. We would like to thank my co-team members for their complete support throughout in finishing the mentioned project accurately. After doing this project We can confidently say that this experience has not only enriched us with technical knowledge but also has unparsed the maturity of thought and vision, the attributes required for being a professional.

**Abstract**

The Covid-19 epidemic is spreading all over the world now. In late December 2019, a cluster of unexplained pneumonia cases has been reported in china. A few days later, the causative agent of this mysterious pneumonia was identified as a novel coronavirus. This causative virus has been temporarily named as severe acute respiratory syndrome coronavirus 2 and the relevant infected disease has been named as coronavirus disease 2019 (COVID -19) by the World Health Organisation.

The purpose of this covid-19 web tracker is that it helps in very effective way to understand the complexity of the situation. Therefore, in absence of pharmaceutical interventions, the implementation of precautions, hygienic measures will be essential to control and to minimise human transmission of the virus.

**Table of Contents**

**Declaration……………………………………………………………………………..2**

**Certificate……...…………………………………………………………………........3**

**Acknowledgement……………………………………………….…………………….4**

**Abstract………………………………………………………………………………...5**

**Table of Content………………………………………………………………...........6-7**

**1. Introduction**

1.1 Overview……………………………………..………………………………….... 13

1.2 Motivation….…………………………………...…………………………….……13

1.3 Project Plan………………………………………..……………………………….13 1.3.1 Objective

1.3.2 Scope

1.4 Drawbacks in existing system………………………..…………………………….14

**2. Software**

2.1 Hardware Requirements………………………………….………………………. 15

2.2 Software Requirements……………………………….…….…………………….. 15

2.3 Installation of VS Code………………………………………...…………………. 15

2.4 Specific Requirements……………………………………………………………..17

2.4.1 Language Used…………………………………………………………………..17

**3. Software Design**

3.1 Use Case Diagram……………………………………………………………..….. 20

3.2 Data Flow Diagram…………………………………..…………………………… 21

**4. Testing**

4.1 Introduction..…………………………………………………………….………….22

4.2 Error……...…………………………………….…………………………..……….22

4.3 Fault…………….……………………………………………………..….…..……..22

4.4 Failure……………………...………………………………………………..……...23

4.5 Functional Test……………..……………………………………………..………...23

4.6 Performance Test………………….………………………………………………..23

4.7 Stress Test…………………………………………………………………………..24

4.8 Structure Test…………………………………………………………………..…...24

**5. Implementation and User Interface……………………………………………..25-39**

**6. Contributions……………………………………………………………………….40**

**7. References/Bibliography………………………..………………………………….41**

**8. Data Collection……………………………………………………………………..41**

**Chapter-1**

**Introduction**

* 1. **Overview**

Coronaviruses are a group of related RNA viruses that causes diseases in mammals and birds. The earliest reports of a coronavirus infection in animals occurred in the late 1920s, when an acute respiratory infection of domesticated chickens emerged in North America. First Human coronaviruses were discovered in the 1960s.

Government of India is taking all necessary steps to ensure that we are prepared well to face the challenge and threat posed by the growing pandemic of COVID-19 the Corona Virus.

* 1. **Motivation**

As coronavirus cases are increasing with a tremendous rate, so for getting UpToDate with every information we all need something which helps us to keep. Here is a covid-19 tracker having information of almost everything related to coronavirus i.e. Covid-19 tracker, News, what precautions are to be taken during this pandemic and much more that helps everyone in an easy way.

* 1. **Project Plan**

**1.3.1. Objective**

To Build a Covid-19 Tracker System that take record of every number of confirmed cases, number of deaths and number of Recovered cases overall the world along with the voice Assistant support, and provides user an interface to get all data and facts regarding covid-19, including graphs, pie charts, bar chats and time series for showing the changing behaviour of covid-19 over the period.

**1.3.2. Scope**

User get all the information regarding Covid-19 cases, precautions, near medical stores and many more benefits.

**1.4. Drawbacks in Existing System**

* No Voice Assistant for better benefits.
* No Offline Updates.
* Not on HTTPS encryptions.

**Chapter-2**

**Software Requirement Analysis**

**2.1. Hardware Requirements**

* **Processor :**
* **Main Memory (RAM) :**
* **Cache Memory :**
* **Monitor :**
* **Keyboard :**
* **Mouse :**
* **Hard Disk :**

**2.2. Software Requirements**

* **System Software**

**- Operating System :** Windows 10 , Linux

* **Application Software**
* **Tools :** GitHub , PyCharm , VS Code
* **Front-end :**
* HTML
* CSS
* JavaScript
* **Back-end :** Flask
* **Programming :** Python

**2.3. Installation of VS Code**

VS Code is a free code editor, which runs on the macOS, Linux, and Windows operating systems.

VS Code is lightweight and should run on most available hardware and platform versions. You can review the [System Requirements](https://code.visualstudio.com/docs/supporting/requirements) to check if your computer configuration is supported.

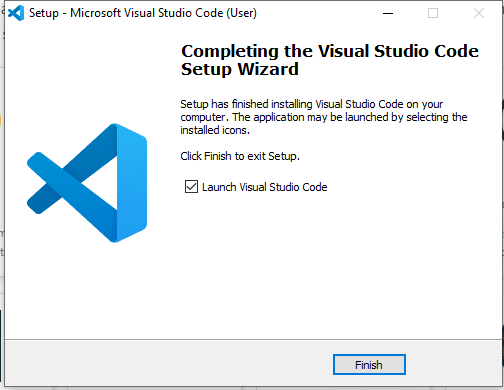


Fig1. Finish up Installing.

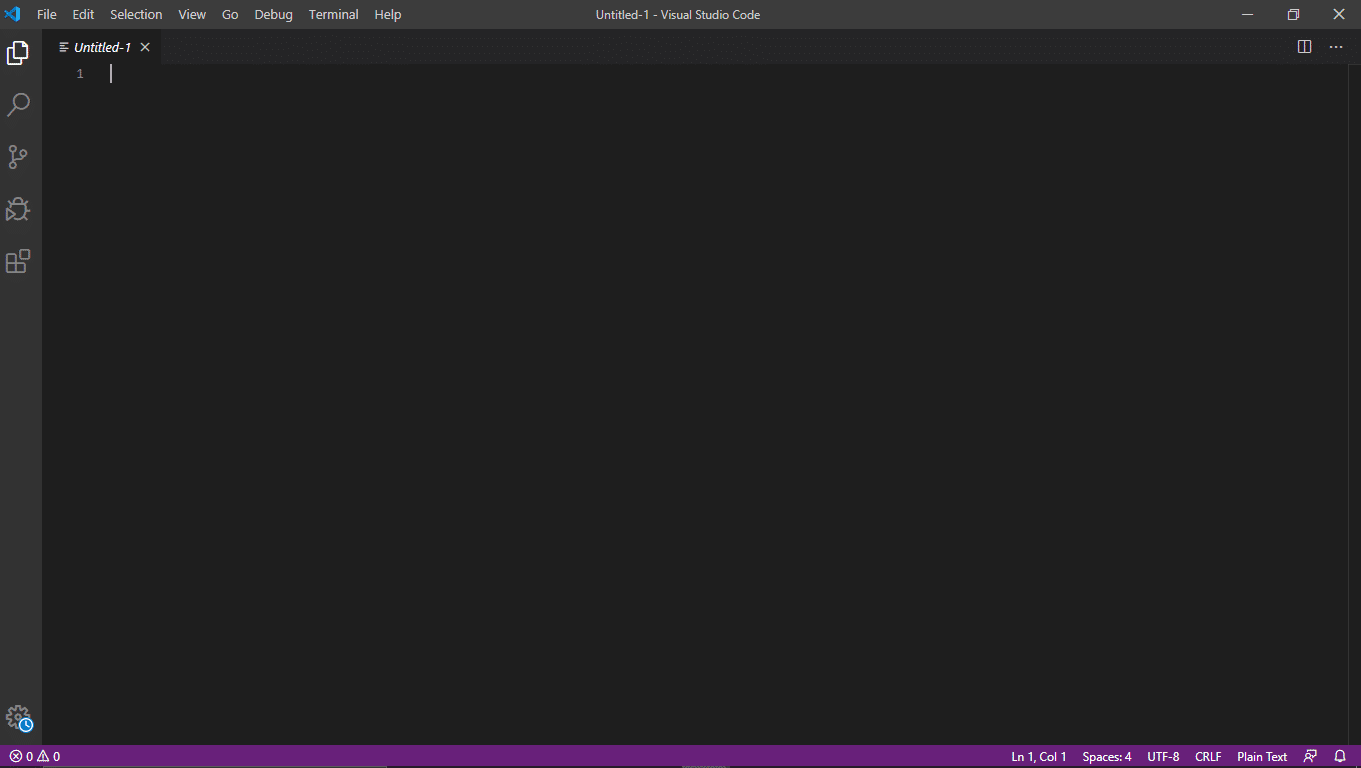


Fig2. VS Code Window.

**2.4. Specific Requirements**

**2.4.1 Languages Used**

**HTML:**

**HTML** stands for **Hyper Text Mark-up Language**, which is the most widely used language on Web to develop web pages. HTMLwas created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

I will list down some of the key advantages of learning HTML:

• **Create Web site** - You can create a website or customize an existing web template if you know HTML well.

• **Become a web designer** - If you want to start a carrer as a professional web designer, HTML and CSS designing is a must skill.

• **Understand web** - If you want to optimize your website, to boost its speed and performance, it is good to know HTML to yield best results.

• **Learn other languages** - Once you understands the basic of HTML then other related technologies like java script, php, or angular are become easier to understand.

**CSS:**

Cascading **S**tyle **S**heets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the colour of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the mark-up languages HTML or XHTML.

• **CSS saves time** − You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.

• **Pages load faster** − If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.

• **Easy maintenance** − To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

• **Superior styles to HTML** − CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

• **Multiple Device Compatibility** − Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.

**JavaScript:**

**JavaScript** is a lightweight, interpreted programminglanguage. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScriptis very easy to implement because it is integrated with HTML. It is open and cross-platform.



Fig3. HTML vs CSS vs JAVASCRIPT

**Chapter-3**

**Software Designs**

**3.1 Use Case diagram**

A **use case diagram** is a dynamic or behaviour diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform.

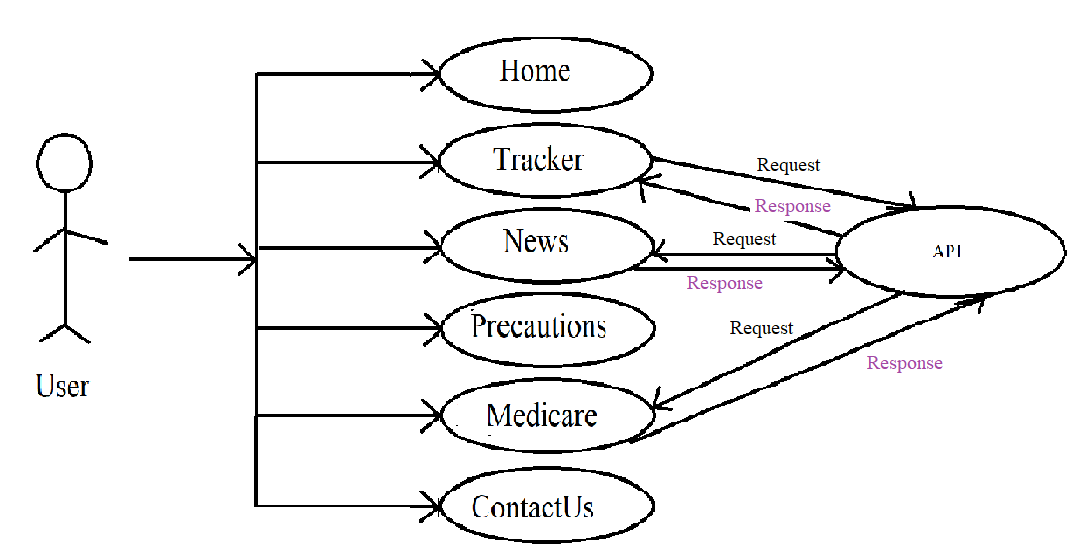
****

Fig 3.1 Use Case diagram

**3.2 Data Flow Diagram**

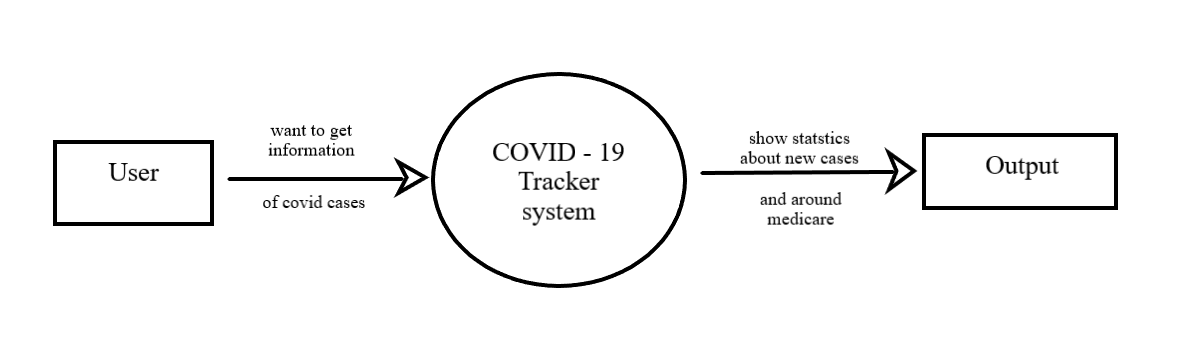
**Level 0:**

Fig. 3.2 0 Level DFD

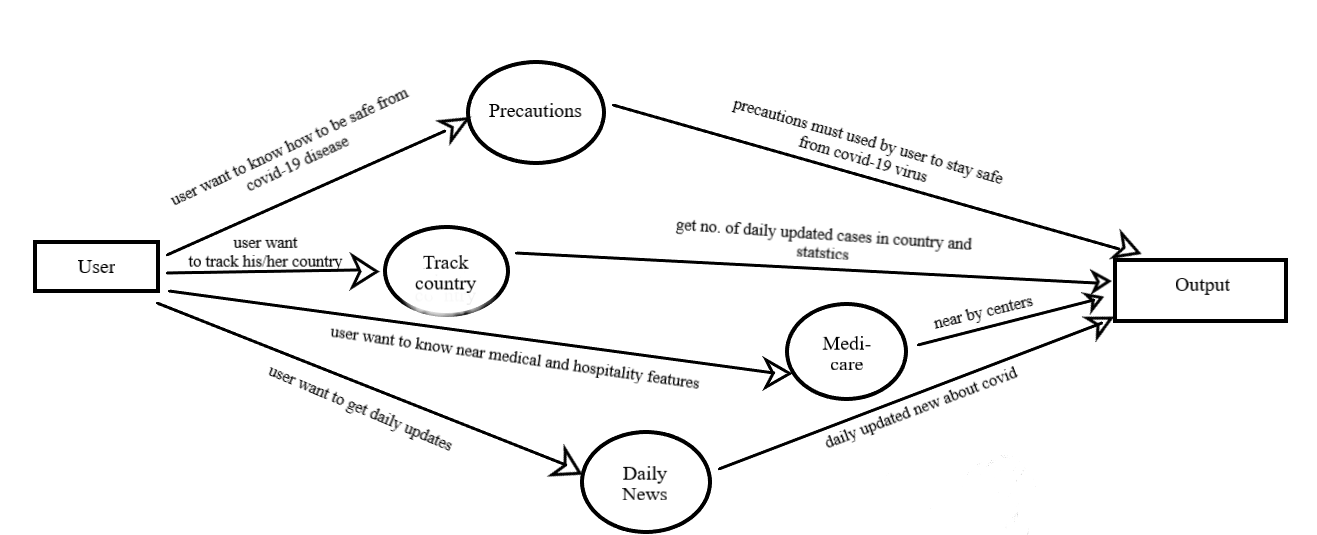
**Level 1:**

Fig. 3.3 1 Level DFD

**Chapter-4**

**Testing**

**4.1 Introduction**

The implementation phase of software development is concerned with translating design specification into source code. The preliminary goal of implementation is to write source code and internal documentation so that conformance of the code to its specifications can be easily verified, and so that debugging, testing and modifications are eased. This goal can be achieved by making the source code as clear and straightforward as possible. Simplicity, clarity and elegance are the hallmark of good programs, obscurity, cleverness, and complexity are indications of inadequate design and misdirected thinking.

Source code clarity is enhanced by structured coding techniques, by good coding style, by, appropriate supporting documents, by good internal comments, and by feature provided in modern programming languages.

The implementation team should be provided with a well-defined set of software requirement, an architectural design specification, and a detailed design description. Each team member must understand the objectives of implementation.



Fig5. Testing

**4.2. Error**

The term error is used in two ways. It refers to the difference between the actual output of software and the correct output, in this interpretation, error is essential a measure of the difference between actual and ideal. Error is also to used to refer to human action that result in software containing a defect or fault.

**4.3. Fault**

Fault is a condition that causes to fail in performing its required function. A fault is a basic reason for software malfunction and is synonymous with the commonly used term Bug.

**4.4. Failure**

Failure is the inability of a system or component to perform a required function according to its specifications. A software failure occurs if the behaviour of the software is the different from the specified behaviour. Failure may be caused due to functional or performance reasons.

**a. Unit Testing**

The term unit testing comprises the sets of tests performed by an individual programmer prior to integration of the unit into a larger system.

A program unit is usually small enough that the programmer who developed it can test it in great detail, and certainly in greater detail than will be possible when the unit is integrated into an evolving software product. In the unit testing the programs are tested separately, independent of each other. Since the check is done at the program level, it is also called program teasing.

**b. Module Testing**

A module and encapsulates related component. So can be tested without other system module.

**c. Subsystem Testing**

Subsystem testing may be independently design and implemented common problems are sub-system interface mistake in this checking we concenton it. There are four categories of tests that a programmer will typically perform on a program unit.

i Functional test

ii Performance test

iii Stress test

iv Structure test

**4.5 Functional Test**

Functional test cases involve exercising the code with Nominal input values for which expected results are known; as well as boundary values (minimum values, maximum values and values on and just outside the functional boundaries) and special values.

**4.6 Performance Test**

Performance testing determines the amount of execution time spent in various parts of the unit, program throughput, response time, and device utilization by the program unit. A certain amount of avoid expending too much effort on fine-tuning of a program unit that contributes little to the overall performance of the entire system. Performance testing is most productive at the subsystem and system levels.

**4.7 Stress Test**

Stress test are those designed to intentionally break the unit. A great deal can be learned about the strengths and limitations of a program by examining the manner in which a program unit breaks.

**4.8 Structure Test**

Structure tests are concerned with exercising the internal logic of a program and traversing particular execution paths. Some authors refer collectively to functional performance and stress testing as “black box” testing. While structure testing is referred to as “white box” or “glass box” testing. The major activities in structural testing are deciding which path to exercise, deriving test date to exercise those paths, determining the test coverage criterion to be used, executing the test, and measuring the test coverage achieved when the test cases are exercised.

**Chapter-5**

**Implementation and User Interface**

**Chapter-6**

**References**

* [www.javatpoint.com](http://www.javatpoint.com)
* [www.w3school.com](http://www.w3school.com)
* [www.tutorialspoint.com](http://www.tutorialspoint.com)
* [www.youtube.com](http://www.youtube.com)
* www.giphy.com

**Chapter-7**

**Data Collection**