

PROFESSIONAL TRAINING REPORT

at

Sathyabama Institute of Science and Technology

(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Technology Degree in Information Technology

By

Preethi S

38120063



DEPARTMENT OF INFORMATION TECHNOLOGY

SCHOOL OF COMPUTING

SATHYABAMA INSTITUTE OF SCIENCE AND

TECHNOLOGY

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,

CHENNAI – 600119. TAMILNADU

Jan 2021



SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of who carried out the project entitled "**COVID VIRUS TRACKER**" under our supervision from August 2021 to November 2021.

Internal Guide

Dr. Mary Gladence.L

Head of the Department

Dr. R. Subhashini M.E., Ph.D.,

Submitted for Viva voce Examination held on -----

Internal Examiner

External Examiner

DECLARATION

I hereby declare that the Project Report entitled “**Corona Virus daily case analysis**” done by me under the guidance of Dr.Mary Gladence.L at Sathyabama Institute of Science and Technology (Deemed to be University), Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600119 is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology.

DATE: 22/01/2021

Place: Chennai

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge our sincere thanks to Board of management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey our thanks to **Dr. T. SASIKALA M.E., Ph.D., Dean, School of Computing** and **Dr. R. SUBHASHINI M.E., Ph.D., Head of the Department, Department of Information Technology** for providing us the necessary support and details at the right time during the progressive reviews.

I would like to express our sincere and deep sense of gratitude to our Project **Dr.Mary Gladence .L.** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express our thanks to all Teaching and Non-teaching staff members of the Department of **INFORMATION TECHNOLOGY** who were helpful in many ways for the completion of the project.

CHAPTER NO	TITLE NAME	PAGE NO
	ABSTRACT	6
1	INTRODUCTION	7
	1.1 GENERAL	7
	1.2 OUTLINE OF THE PROJECT	7
2	AIM AND SCOPE	8
	2.1 AIM	8
	2.2 PROBLEM STATEMENT	8
	2.3 SCOPE	8
3	SYSTEM ANALYSIS	9
	3.1 PACKAGES USED	9
4	SYSTEM REQUIREMENTS	10
5	CONCLUSION AND FUTURE WORK	11
	5.1 CONCLUSION	11
	5.2 REFERENCE	11
6	APPENDIX	12
	6.2 SOURCE CODE	12
	6.2 SCREENSHOT	17

ABSTRACT:

Corona virus disease (covid-19) Daily Cases

· **Corona disease daily case analysis** this project contains dashboard(s) and other visualizations to monitor the outbreak.. The “Best current data source” is usually the state or territory’s public health department’s coronavirus-specific response page, and additional data sources are usually coronavirus-specific data public health department web pages. We are careful to make sure that we collect data from official state/territory sources.

The implementation of above model would help one to see the daily cases of covid in better way and can understand easily. COVID-19 data definitions change frequently across the 56 jurisdictions we track, so we must sometimes update our data definitions to reflect the realities we find in the data. As we track additional data points and adjust the definition of current fields to accommodate changes from states and territories, we will update the definitions here. Almost all of the data we compile is taken directly from the websites of state/territory public health authorities

CHAPTER 1

INTRODUCTION

1.1 GENERAL

I have made Analysis to simply know the daily confirmed cases death cases and recovered cases.

By clicking the Dropdown button we can choose the country and it will display the details of the daily case updates for different country .

1.2 OUTLINE OF THE PROJECT

Fetching the latest Country wise data of COVID-19 cases from the official website of ministry of health and visualizing it using Python libraries. This country analysis is unique for all different countries and will be updated with the most recent data as frequently . It will display the daily cases of COVID-19 for different countries.

CHAPTER 2

AIM AND SCOPE

2.1 AIM

The main objective of this project is to providing up to covid-19 daily cases from around the world. Showing the latest states and information related to COVID-19.

2.2 PROBLEM STATEMENT

If we want to see the data set of covid -19, it is tough to find the country wise one by one. For the fast going generation, they will not like to go through the data on by one. This dropdown analysis will make easy by clicking the country.

2.3 SCOPE

I have made analysis to simply know the daily confirmed cases,death cases and recovered cases.

I stored a list of countries in LIST data structure and i made it has dropdown button by clicking the country we know the daily covid cases update of that country.

CHAPTER 3

SYSTEM ANALYSIS

3.1.Packages Used:

- ❖ **Beautifulsoup**-A library for pulling data out of html and xml files.
- ❖ **Requests**-A library for making HTTP requests in Python.
- ❖ **GeoPandas**-A library for working With geospatial data in python.
- ❖ Other regular packages like pandas .

CHAPTER 4

RESULTS AND DISCUSSION

4.1 SYSTEM REQUIREMENTS

Hardware Requirement:

- ▶ Hard disk : 120 GB
- ▶ Monitor : 15' color with vgi card support
- ▶ Ram : Minimum 256 MB
- ▶ Processor : Pentium iv and above (or) equivalent
- ▶ Processor speed : Minimum 500 MHZ

Software Requirement:

- ▶ Operating system:Windows 10
- ▶ Language:Python
- ▶ IDE:Python IDLE

CONCLUSION AND FUTURE WORK

- ❖ The implementation of above module would help one to see the covid cases for different countries in better way and can understand easily.
- ❖ One can know the daily cases by clicking the country in the button.No need to refer dataset.

Readers are likely experiencing some latent anxiety, so do your best to neither make light of the situation nor be alarmist about it. One application of this is in your colour choice: You don't want your map's colour scheme or design to minimize the situation by being overly playful or light hearted. You also don't want to select colours that suggest the worst possible outcome. If a location does not appear in the visualizations , it may be considered a "dependent territory" and its COVID-19 activity has been aggregated

5.3 REFERENCE

1. : <https://www.w3schools.com/python/>

APPENDIX

Source code:

```
from PyQt5.QtWidgets import *
from PyQt5 import QtCore, QtGui
from PyQt5.QtGui import *
from PyQt5.QtCore import *
from bs4 import BeautifulSoup as BS
import requests
import sys

class Window(QMainWindow):
    def __init__(self):
        super().__init__()
        # setting title
        self.setWindowTitle("Python life ")
        # setting geometry
        self.setGeometry(100, 100, 400, 500)
        # calling method
        self.UiComponents()
        # showing all the widgets
        self.show()
        # method for widgets
    def UiComponents(self):
        # countries list // user can add other countries as well

        self.country = ["india","us","spain","china","russia", "pakistan",
"nepal", "italy", "spain", "uk","brazil"]
        # creating a combo box widget
        self.combo_box = QComboBox(self)
        # setting geometry to combo box
```

```

        self.combo_box.setGeometry(100, 50, 200, 40)
# setting font
        self.combo_box.setFont(QFont('Times', 10))
# adding items to combo box
        for i in self.country:
            i = i.upper()
            self.combo_box.addItem(i)

# adding action to the combo box
self.combo_box.activated.connect(self.get_cases) # creating label to show
the total cases

        self.label_total = QLabel("Total Cases ", self)
# setting geometry

self.label_total.setGeometry(100, 300, 200, 40)

# setting alignment to the text

        self.label_total.setAlignment(Qt.AlignCenter)
# adding border to the label

        self.label_total.setStyleSheet("border : 2px solid blue;")
# creating label to show the recovered cases
        self.label_reco = QLabel("Recovered Cases ", self)
# setting geometry

        self.label_reco.setGeometry(100, 350, 200, 40)

# setting alignment to the text

        self.label_reco.setAlignment(Qt.AlignCenter)
# adding border

        self.label_reco.setStyleSheet("border : 2px solid green;")

# creating label to show death cases

```

```

        self.label_death = QLabel("Total Deaths ", self)

# setting geometry
        self.label_death.setGeometry(100, 400, 200, 40)

# setting alignment to the text
        self.label_death.setAlignment(Qt.AlignCenter)

# adding border to the label
        self.label_death.setStyleSheet("border : 2px solid red;")

# method called by push
        def get_cases(self):

# getting country name
            index = self.combo_box.currentIndex()
            country_name = self.country[index]

# creating url using country name
            url="https://www.worldometers.info/coronavirus/country
/" + country_name + "/"

# getting the request from url
            data = requests.get(url)

# converting the text
            soup = BS(data.text, 'html.parser')

# finding meta info for cases
            cases = soup.find_all("div", class_="maincounter-number")

# getting total cases number
            total = cases[0].text

# filtering it
            total = total[1: len(total) - 2]

# getting recovered cases number
            recovered = cases[2].text

# filtering it
            recovered = recovered[1: len(recovered) - 1]

# getting death cases number
            deaths = cases[1].text

# filtering it
            deaths = deaths[1: len(deaths) - 1]

```

```
# show data through labels
        self.label_total.setText("Total Cases : " + total)
        self.label_reco.setText("Recovered Cases : " + recovered)
        self.label_death.setText("Total Deaths : " + deaths)

# create pyqt5 app
        App = QApplication(sys.argv)

# create the instance of our Window
        window = Window()
        window.show()

# start the app
        sys.exit(App.exec())
```

SCREENSHOTS





