**EXECUTIVE SUMMARY**

The report traces an original of a jumbled source information heart that underpins the contention of information among a few purposes of examination, showing, and generation instruments. The facilities information includes an arrangement of works involving of framework extraction, setting, and perception centers for GE chiseling applications. The center element of the facilities information that it is an information diagram work to offer a dependable and direct sensible structure for keeping shared showing information in various purposes. Through support and purchase in from the showing association, the information layout forms into an intelligent or item arranged database to additionally improve the capacity to trade information across goals and facility tools. This server contains the information related to COVID-19 pandemic. The result of the test applications show that the facilities and area information arrived at the venture target of improving the lovely sign of ideal information alongside critical time savings. The process required the capacities. The aptitudes were created by application. The data originated from the file containing details of COVID-19 of all countries out of which I selected top 15 countries for case study. The path toward removing gaining from the data included data evacuation. Data extraction, the extraction of insightful information concealed in gigantic databases, Isa impelled system that enables associations to include the most fundamental information in their data dispersion focuses.

**TABLE OF CONTENTS**

**Pages**

CHAPTER 1.

* 1. Introduction 01

CHAPTER 2.

* 1. Service Oriented Architecture (SOA) 02

CHAPTER 3.

* 1. Data Cleaning and Merging 03
  2. RESTful Web Services 05
  3. Mashup Design 07

CHAPTER 4.

4.1 Demo Running Instructions 08

CHAPTER 5.

5.1 Conclusion and Recommendations 10

**TABLE OF FIGURES**

[Figure 1 Data in New.csv 3](#_Toc41303495)

[Figure 2 Data in XML file 4](#_Toc41303496)

[Figure 3 Output when you run RESTful Web Service 6](#_Toc41303497)

[Figure 4 data\_merger.py file 8](#_Toc41303498)

[Figure 5 Home Screen 9](#_Toc41303499)

## CHAPTER 1

## INTRODUCTION

The point of this report is to make Mashup Software, which will get the top 15 countries in the world that are affected by COVID-19. The system will show COVID-19 details corresponding to the country entered by the user. This is accomplished in three stages:

1.Use of petl module to concentrate and burden information from csv records to table and to change over XML document into the table which is later consolidated. Additionally, data cleaning is done to clean the missing information from the CSV document.

2.In following stage a RESTful WS is structured which depends on the Python Bottle Framework, it returns information in JSON position

3.In the end, a Mashup is created which gives the GUI to the client, where a client could enter country name to get COVID-19 details.

To accomplish this, text editor Sublime Text is utilized and the code is created utilizing Python, HTML, CSS, and JS.

## 

## CHAPTER 2

## 2.1 SERVICE ORIENTED ARCHITECTURE (SOA)

Service-Oriented Architecture (SOA) is a compositional methodology where applications utilize services accessible in the system. In this architecture, services are given to frame applications, through a correspondence bring over the web.

SOA permits clients to join countless offices from existing services to shape applications.

SOA envelops a lot of plan rules that structure framework improvement and give intends to coordinating parts into an intelligent and decentralized framework.

SOA based figuring bundles functionalities into a lot of interoperable services, which can be incorporated into various programming frameworks having a place with discrete business spaces.

There are two significant jobs inside Service-oriented Architecture:

1. Service supplier: The service supplier is the maintainer of the service and the association that makes accessible at least one services for others to utilize. To publicize services, the supplier can distribute them in a library, along with a service contract that determines the idea of the service, how to utilize it, the necessities for the service, and the expenses charged.
2. Service purchaser: The service customer can find the service metadata in the vault and build up the necessary customer parts to tie and utilize the service.

**CHAPTER 3**

**3.1 DATA CLEANING AND MERGING**

The data given has missing values, which in turn will lead to the inconsistent results. Also, the data given is in different formats which are merged later into one.

Files that needed to be analysed for this project:

**1. New.csv:** This is a Comma Separated value file. In it, there are complete details of top 15 countries affected by COVID-19. **Figure 1** shows the data in the New.csv file.

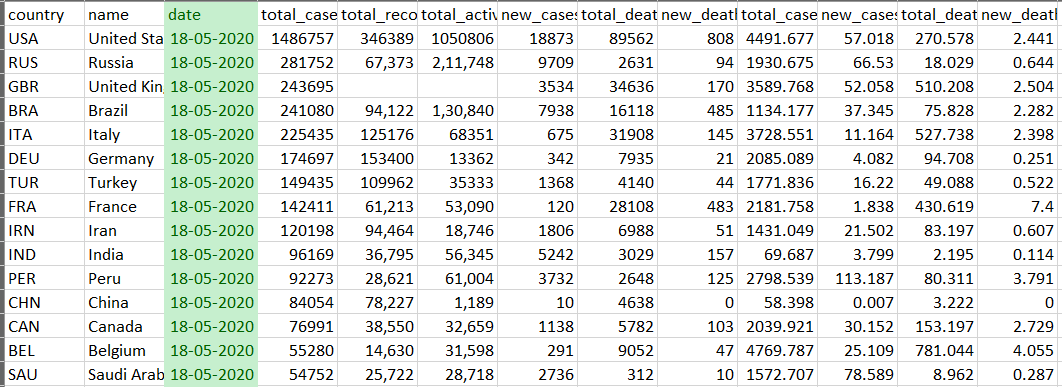


Figure 1 Data in New.csv

**2. Country\_location.xml:** Another record is a XML document in which contains Longitude and Latitude of Countries in XML record. As it is free of the stage in this way, no compelling reason to insert it with some other configuration. The underneath **figure 2** shows the XML information in the program.

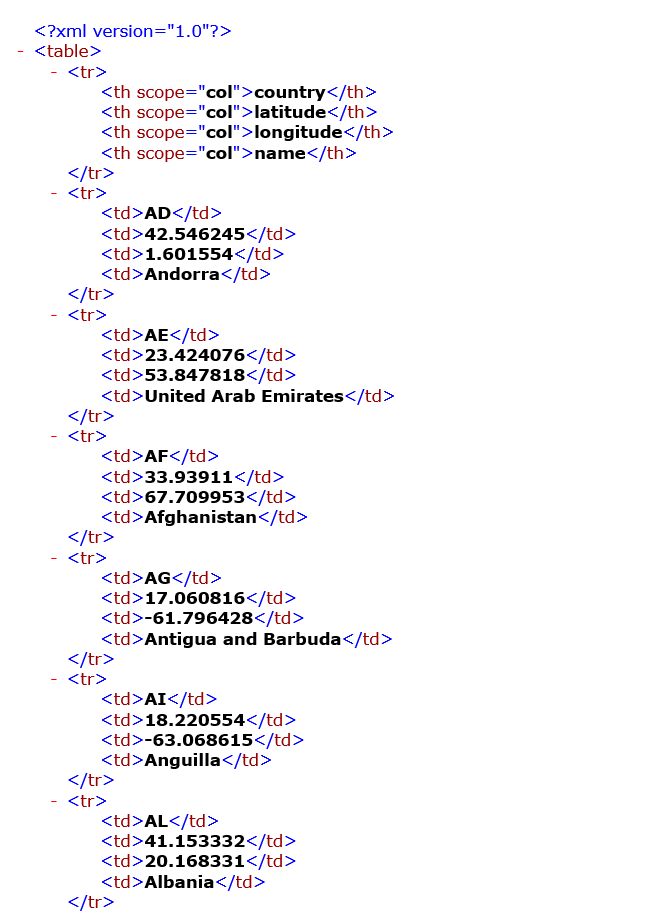


Figure 2 Data in XML file

**Data Cleaning:**

As said before, the information that is given as CSV document (Comma-Delimited Values) have missing qualities. Some information fields are missing which need to clean before the Merging procedure.

**Data Merging:**

In this undertaking of framework plan, we have two sorts of Data. To utilize them as one they should be handled as together. In this procedure, those two documents are combined as one named under "covid\_countries.csv". In this procedure, the two records are changed over into Tables utilizing the petl module in the python and afterward consolidated into one record.

**3.2 RESTFUL WEB SERVICES**

RESTful Web Service is implemented on web administration engineering executed by utilizing a Virtual structure in a Hierarchical design. In this venture, it is executed by utilizing **Bottle** Framework.

**Accessing the RESTful Web Service:**

The URL given in project is following:

http://localhost:8080/getcountry?country=xxxx (xxxx is the name of the country that the user will enter in the web browser to run this service)

To start the server there is a record called **“country\_locator.py”** open that document in PyCharm and run it. This container actualized system which will be running on default internet browser. The information will be taken from the recently made csv record **“country\_locator.py”**.

This web server will yield the output in the organization of JSON object, which will be brought later by the following procedure (talked about in subsequent stage). To run this procedure the system must have Bottle framework installed in it.

The **Figure 3** shows the output that we get after running it in PyCharm. The output is in JSON object format because it will send the data over network in a structured way. Also, JSON objects are the most used format in web applications.

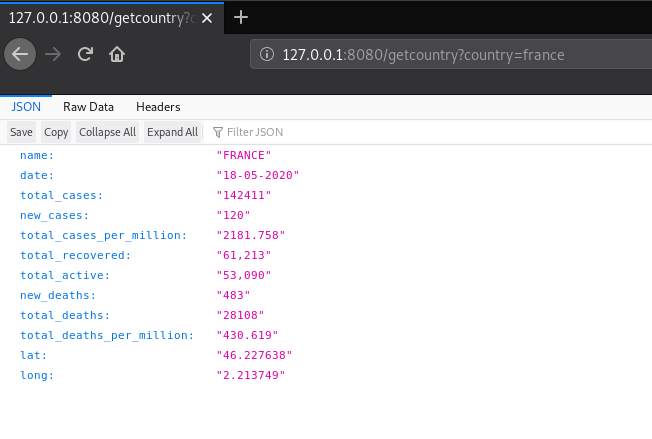


Figure 3 Output when you run RESTful Web Service

The above figure shows the details corresponding to the name of country passed.

The output has several values

1.Country Name: FRANCE

2.Total Cases: 142411

3.Total Deaths: 28108

4.Cases per million: 2181.758

**3.3 MASHUP DESIGN**

Mashups is a half and half application that is created from at least two sources like web application or site pages. Mashups utilize those applications to make a solitary application or in a solitary graphical interface. Like in this task

1. A RESTful Web Service is made which takes contribution from the yield of the python code composed of the on framework preparing of the information

2. The Output of the RESTful Web Service is taken care of as a contribution to the Mashup application as JSON information and further handling is finished.

In this, HTML document is made with the assistance of CSS and JavaScript without any preparation. The record made in this procedure is “country\_map.html”. The Objective of this Mashup is:

1. The Created HTML site page will take contribution from client as country name.

2. The HTML page will bring the information from the RESTful web administration made in the past advance in the JSON structure and will show the area on the Google map.

For demonstrating the Google map the Google API is utilized. From this API I could show the details of the country that is entered by the user.

There are two sources of this Mashup:

1. The covid\_countries.csv file which I have created in the first step after Data Cleaning and Data Merging processes.
2. The RESTful Web Service that is performed in the previous step in order to start the Web Service.

**CHAPTER 4**

**4.1 DEMO RUNNING INSTRUCTIONS**

The project is executed in three Phases:

1. Data Cleaning and Merging
2. Making RESTful Web Service utilizing Python Bottle
3. Creating Mashup utilizing over two stages

### Perquisites:

1. PyCharm or any other compatible IDE
2. Petl framework
3. Bottle must be installed

### For Data Cleaning and Data Merging:

### First open PyCharm IDE

### Open file Called “data\_merger.py” and hit run. You can run through terminal as well.

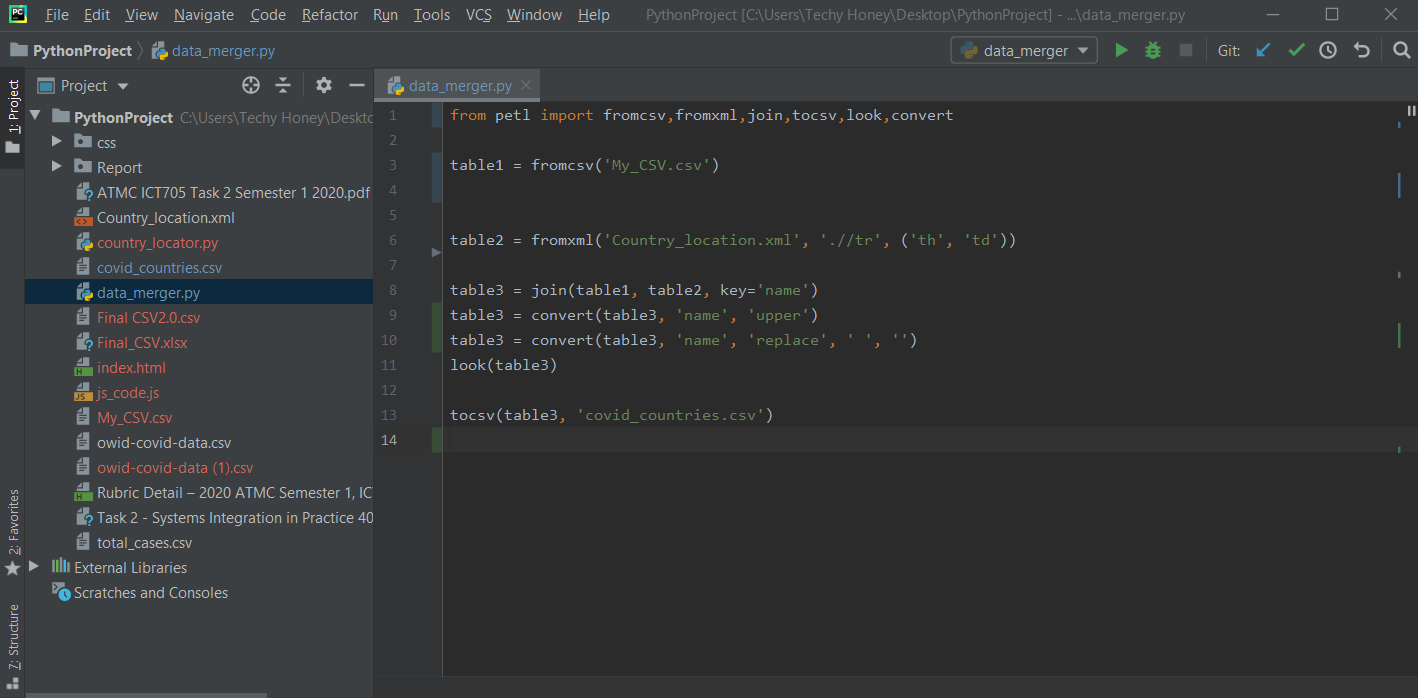


Figure 4 data\_merger.py file

### For Running RESTful Web Service:

### Open file Called “country\_locator.py” and hit run.

### User must have an internet connection for the Google maps to fetch.

### Now open “country\_map.html”. File will be opened in your default browser. You will see page shown below.

### 

Figure 5 Home Screen

**CHAPTER 5**

**5.1 CONCLUSION & RECOMMENDATIONS**

In the Nutshell, this report centres around making a Mashup application utilizing which the users can discover the details related to COVID-19. The country name is searched in the csv and coordinate of the country is marked on the map if the country is present in the case study, if not present the error message will be shown.

To accomplish this, few things are finished. From the outset, the information that is given is cleaned for missing qualities and afterward converged with the diverse record types. The output is then put away in a record which is utilized by both “country\_locator.py” and “country\_map.html” as an information document. This is a significant advance as it will change the whole outcome.

In the subsequent advance, a web server is structured utilizing Bottle system. It takes “covid\_countriess.csv” as info and returns the JSON object information which is later utilized by “country\_map.html” to look and show the areas.

In the last advance, a Mashup application is created utilizing over two procedures. Utilizing the documents made above as info this Mashup application shows the details related to COVID-19 of country getting the country name from the client.

At the stage of making RESTful Web Service we may encounter some difficulties such like returning JSON Object or any other. This problem was resolved by referring the Bottle Framework documentation and some helpful articles.

**Recommendations:**

Service Oriented Architecture is highly beneficial to OSL as they deal with trades all over the world.

Some of the Advantages of (SOA) are:

1. **Easy Maintenance**: SOA can be easy maintained as services are independent and can be easily updated.
2. **Scalability**: As OSL has a world-wide shipping services so in SOA services can run on different servers, this increases scalability.
3. **Reliability:** SOA applications are more reliable and is easy to debug the code.

Some of the Disadvantages of (SOA) are:

1. **High Investment**: A huge initial investment is required in SOA
2. **Complex Service Management**: At the point when services communicate, they trade messages to errands. the quantity of messages may go in millions. It turns into an awkward undertaking to deal with an enormous number of messages.

**REFRENCES**

Petl Documentation: <https://petl.readthedocs.io/en/stable/>

Bottle Framework Documentation: <https://bottlepy.org/docs/dev/tutorial.html>