**1.Introduction :**

* 1. **Purpose :**

The dashboard provides daily updates of the pandemics COVID-19 using tableau.

Tableau makes it easier to create visual analytics and reports in the form of dashboards.

This dashboards can get updated in various time intervals.

* 1. **Acronyms & Abbrevations :**

Dashboard shows **total cases,total deaths,total morality,growth from last month** worldwide.

Dashboard also shows cases country wise, deaths country wise, graphical representation of daily counts, World map indicating country wise cases.

* 1. **Project Scope :**

A dashboard is a type of graphical user interface which often provides at a glance views of key performance indicators relevants to a particular objective or business process.The dashboard is often displayed on a web page which is linked to a database that allows the report to be constantly updated.

* 1. **Reference :**

**1.**Software Requirement Specification ,Computer Information System Program,Janusz Zalewski.

**2.**Software Engineering: A Practitioner’s Approach.

**3.** Software project management/Bob Hughes and Mike Cottrell,London.

1.5 Overview of Document

This section describes the overview of the entire document.  brief introduction is

provided here.

 Section 1: This section describes the purpose and scope of the project.

 Section 2:  This section describes the functions,product perspective, various user

characteristics, constraints, limitations, assumptions and dependencies.

Section 3: This section describes the requirements of the entire system. This

includes hardware requirements, user interfaces, use case diagram, design

constraints, functional and non-functional requirements.

Section 4: This section describes  about the system evolution

1.5 Overview of Document

This section describes the overview of the entire document.  brief introduction is

provided here.

 Section 1: This section describes the purpose and scope of the project.

 Section 2:  This section describes the functions,product perspective, various user

characteristics, constraints, limitations, assumptions and dependencies.

Section 3: This section describes the requirements of the entire system. This

includes hardware requirements, user interfaces, use case diagram, design

constraints, functional and non-functional requirements.

Section 4: This section describes  about the system evolution

1.5 Overview of Document

This section describes the overview of the entire document.  brief introduction is

provided here.

 Section 1: This section describes the purpose and scope of the project.

 Section 2:  This section describes the functions,product perspective, various user

characteristics, constraints, limitations, assumptions and dependencies.

Section 3: This section describes the requirements of the entire system. This

includes hardware requirements, user interfaces, use case diagram, design

constraints, functional and non-functional requirements.

Section 4: This section describes  about the system evolution

1.5 Overview of Document

This section describes the overview of the entire document.  brief introduction is

provided here.

 Section 1: This section describes the purpose and scope of the project.

 Section 2:  This section describes the functions,product perspective, various user

characteristics, constraints, limitations, assumptions and dependencies.

Section 3: This section describes the requirements of the entire system. This

includes hardware requirements, user interfaces, use case diagram, design

constraints, functional and non-functional requirements.

Section 4: This section describes  about the system evolution

1.5 Overview of Document

This section describes the overview of the entire document.  brief introduction is

provided here.

 Section 1: This section describes the purpose and scope of the project.

 Section 2:  This section describes the functions,product perspective, various user

characteristics, constraints, limitations, assumptions and dependencies.

Section 3: This section describes the requirements of the entire system. This

includes hardware requirements, user interfaces, use case diagram, design

constraints, functional and non-functional requirements.

Section 4: This section describes  about the system evolution

* 1. **Overview of Project**

This section describes the overview of the entire document. Brief introduction is provided here.

**Section 1:** This section describes the purpose and scope of the project.

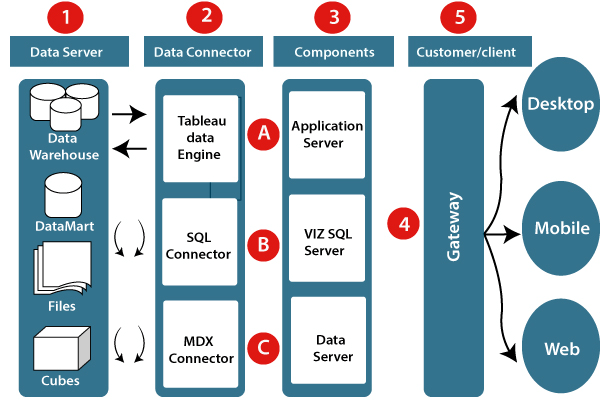
**Section 2:**This section describes the functions,product perspective,various user characteristics,constraints,limitations,assumptions and dependencies.

**Section 3:**This section describes the requirements of the entire system.

**Section 4:** This section describes about the system evolution.

1. **Description of Project**
   1. **Overall Description**

This section will give an overview of the whole system showing its interaction with other interrelated system.This basic functionality of the system will also be explained.This section will explain the types of stakeholder who will use the system and their respective available functionality.The constraints and assumption used for the system is also explained at the end.

**2.2 Product Perspective :** 

**1. Data server:-** The primary component of Tableau Architecture is the Data sources which can connect to it.

Tableau can connect with multiple data sources. It can blend the data from various data sources. It can connect to an **excel file, database**, and a **web application** at the same time. It can also make the relationship between different types of data sources.

**2. Data connector:-** The Data Connectors provide an interface to connect external data sources with the Tableau Data Server.

Tableau has in-built SQL/ODBC connector. This ODBC Connector can be connected with any databases without using their native connector. Tableau desktop has an option to select both extract and live data. On the uses basis, one can be easily switched between live and extracted data.

* **Real-time data or live connection:** Tableau can be connected with real data by linking to the external database directly. It uses the infrastructure existing database by sending dynamic **multidimensional expressions (MDX)** and SQL statements. This feature can be used as a linking between the live data and Tableau rather than importing the data. It makes optimized and a fast database system. Mostly in other enterprises, the size of the database is large, and it is updated periodically. In these cases, Tableau works as a front-end visualization tool by connecting with the live data.
* **Extracted or in-memory data:** Tableau is an option to extract the data from external data sources. We make a local copy in the form of Tableau extract file. It can remove millions of records in the Tableau data engine with a single click. Tableau's data engine uses storage such as **ROM, RAM**, and **cache** memory to process and store data. Using filters, Tableau can extract a few records from a large dataset. This improves performance, especially when we are working on massive datasets. Extracted data allows the users to visualize the data offline, without connecting to the data source.

**3. Components of Tableau server:** Different types of component of the Tableau server are:

* Application server
* VizQL server
* Data server

**A. Application server:** The application server is used to provide the authorizations and authentications. It handles the permission and administration for mobile and web interfaces. It gives a guarantee of security by recording each session id on Tableau Server. The administrator is configuring the default timeout of the session in the server.

**B. VizQL server:** VizQL server is used to convert the queries from the data source into visualizations. Once the client request is forwarded to the VizQL process, it sends the query directly to the data source retrieves information in the form of images. This visualization or image is presented for the users. Tableau server creates a cache of visualization to reduce the load time. The cache can be shared between many users who have permission to view the visualization.

**C. Data server:** Data server is used to store and manage the data from external data sources. It is a central data management system. It provides **data security, metadata management, data connection, driver requirements**, and data storage. It stores the related details of data set like **calculated fields, metadata, groups, sets**, and **parameters**. The data source can extract the data as well as make live connections with external data sources.

**4. Gateway:** The gateway directed the requests from users to Tableau components. When the client sends a request, it is forwarded to the external load balancer for processing. The gateway works as a distributor of processes to different components. In case of absence of external load balancer, the gateway also works as a load balancer. For single server configuration, one gateway or primary server manages all the processes. For multiple server configurations, one physical system works as a primary server, and others are used as worker servers. Only one machine is used as a primary server in Tableau Server environment.

**5. Clients:** The visualizations and dashboards in Tableau server can be edited and viewed using different clients. Clients are **a web browser, mobile applications**, and **Tableau Desktop**.

* **Web Browser:** Web browsers like **Google Chrome, Safari**, and **Firefox** support the Tableau server. The visualization and contents in the dashboard can be edited by using these web browser.
* **Mobile Application:** The dashboard from the server can be interactively visualized using mobile application and browser. It is used to edit and view the contents in the workbook.
* **Tableau Desktop:** Tableau desktop is a business analytics tool. It is used to **view, create**, and **publish** the dashboard in Tableau server. Users can access the various data source and build visualization in Tableau desktop.

**2.3 Assumptions and dependencies**

The architecture will be focusing on providing E-Health facilities online. So our assumption is that the internet facility, power supply, backup power, internet bandwidth is always up and functioning. Another assumption also a dependency for the architecture is hardware configuration of the central server. We assume that the hardware will be above specification, so that it will not need to face any problem in future.

**3. Specific Requirements**

This section contains all of the functional and quality of the system. It gives a detailed Description of the system and all its features.

**3.1 External Interface Requirements**

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware and software interfaces and provides basic prototypes of the user interface.

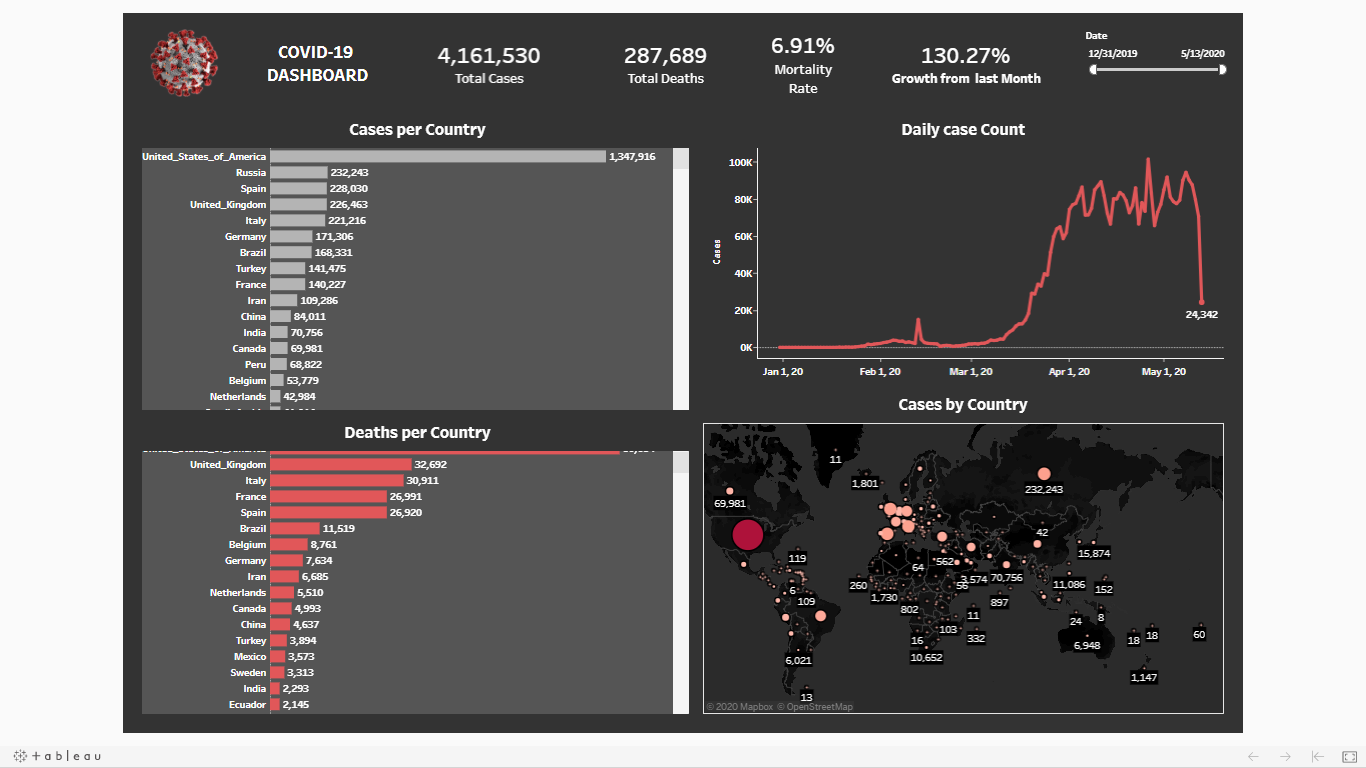
**3.1.1 User Interfaces**

The major interfaces of the system are:

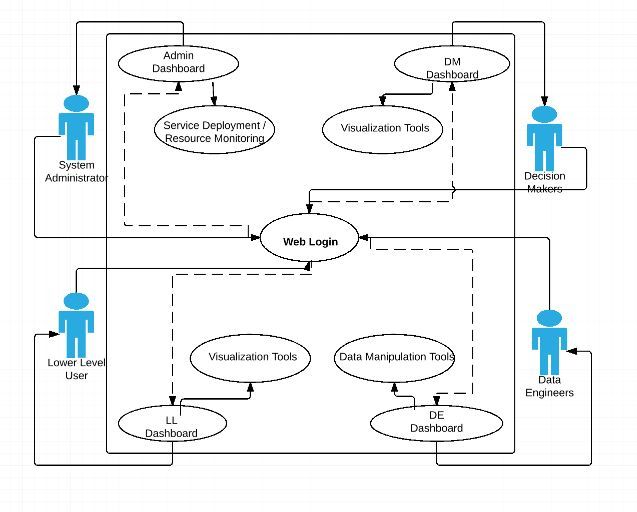
**Dashboard()**

The dashboard will be accessible with all user types. Here they will perform respective tasks

According to their permission level.



**Use Case Diagram:A use case diagram is shown below** **:**



**Figure:Use Case Diagram displaying user interactions**

In above use case diagram, four level of users system administrator, lower level users, data

engineers and decision makers are displayed. With all users the way they access their own

customized dashboard is displayed.

**3.1.2 Hardware Interfaces**

Even though this architecture is hardware-software integrated web architecture, we will not be any specific hardware interface to run the system. Our system is a web based system, so we will be launching it in several computers online.

**3.2 Design Constraints**

This section includes the design constraints on the architecture caused by the hardware.

**3.3 Software system attributes**

The requirements in this section specify the required reliability, availability, security & maintainability of the software system.

**3.4 Functional Requirement:**

1.Connect to the official Tableau COVID-19 Google sheet

<https://docs.google.com/spreadsheets/d/14guQPFErG-hIpsrNgYvX85vW&JMMK5X2vNZrafRcH8c/edit>

2. Make sure you authenticate Tableau in your browser and sign in via a google account.

3.Review the mock up dashboard on the next page.You will needto create the following:

1. A headline card representing total cases of the country
2. A headline card representing total deaths of the country
3. A headline card representing mortality rate of the country
4. A line graph showing confirmed cases per day
5. A line graph showing confirmed deaths per day
6. A bar chart/line graph showing thw mortality rate per day
7. A pie chart showing the cases confirmed and deaths per day

4.When you are done with your dashboard click on “server” and publish to taleau Public.

5.Upload a screenshot of your dashboard and a link to your Tableau Public dashboard so everyone taking the course can see your waorl!

**Some Pointers:**

1)Remember to filter on date = “yesterday”

2)For Mortality rate, remember it’s a two-step process. Step 1 is to calculate deaths as its own separate field. Step 2 will be to calculate the mortality rate which is:

SUM(deaths)

SUM(cases)

**APPENDIX 1:CONFORMITY ASSESSMENT ACTIVITIES AND STAKEHOLDERS(CAAS)**

Conformity Assessment Activities and Stakeholders as shown in above diagram

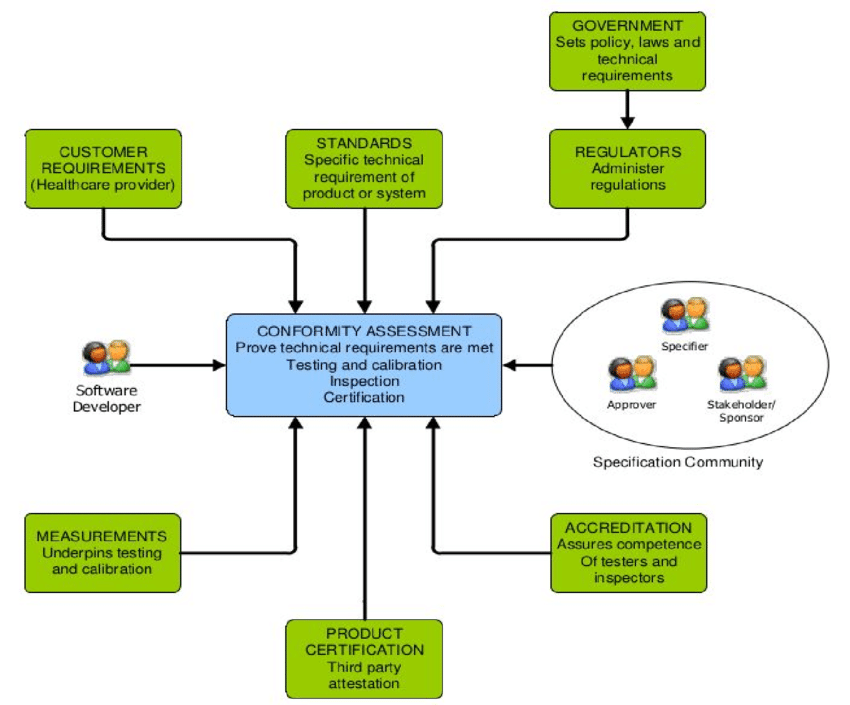
focuses on stability of a system considering various constraints like: measurements,

product certifications, standards and customer's requirements. CAAS also incorporates

developers, governments, regulators, approver, specifiers, stakeholders and sponsors

to provide an organisational framework in which inter-related artefacts are verified for

consistency,traceability and compatibility of the system.



**Consistency** is a characterisation of the logical coherence of the artefacts that are defined in the specification viewpoints.

**Traceability** defines relationships that link an attribute or other feature of a particular Specification artefact with another artefact in the other (or same) cell of the specification matrix.

**Compatibility** is a relationship between two or more conformance statements involving two or more specification instances. The relationship identifies whether two or more implementations claiming to be conformant to the specification instances can achieve interoperability.

The figure shown above is the interface of a dashboard. Almost all the interfaces like UploadSoftware(), ViewSoftware(), GenerateReport(), DataExport(), SysAdmin() will be accessed from the dashboard. It it the system administrator who assigns the access to each type of users. From the dashboard all types of users will access the softwares. System Administrator will upload a project, data engineers will export the data and prepare report and the high level officials will view them. And all activities will be done through the Dashboard shown above.

**UploadSofware()**

This interface is accessible to system administrator who will upload E-Health Projects to the architecture.

**ViewSoftware()**

Here all the software uploaded by system administrator is collected. Users will view application only assigned to them.

**GenerateReport()**

This interface will be used mainly by Data Engineer to analyze data and produce report. Also High level managers will see this interface.