Low-Level Design (LLD)

Health Analytics on Heart Disease Data

Last Revised Date: 31/08/2021

Document Version Control

Version	Date	Author	Comments
0.1	25-Aug-2021	Samruddhi Deshmukh Manish Kumar Subhabrat Nath	
0.2	31-Aug-2021	---	---

Reviews:

Version	Date	Reviewer	Comments

Approval Status:

Version	Review Date	Reviewed By	Approved By	Comments

Index

Abstract

Heart disease is the most major health issue that is suffering by many people all over the globe, some of the causes of heart diseases due to hypertension, diabetes, overweight, and an unhealthy lifestyle. This project of Healthcare Analysis on Heart Disease Data is aimed to explore the Heart Disease dataset .The objective is analyze the various features and their relationship with each other and find out their contribution towards getting a heart disease.

Various features such as **Age, Sex, Chest pain type, Blood pressure, Cholestrol,Fasting Blood sugar, Rest ECG, Thalach, Exercise enduced Angina,Major vessels,oldpeak,slope,thal** are present in the dataset.The goal of the project is to find all types of relationships between the features and come out with significant contributors to a heart disease.

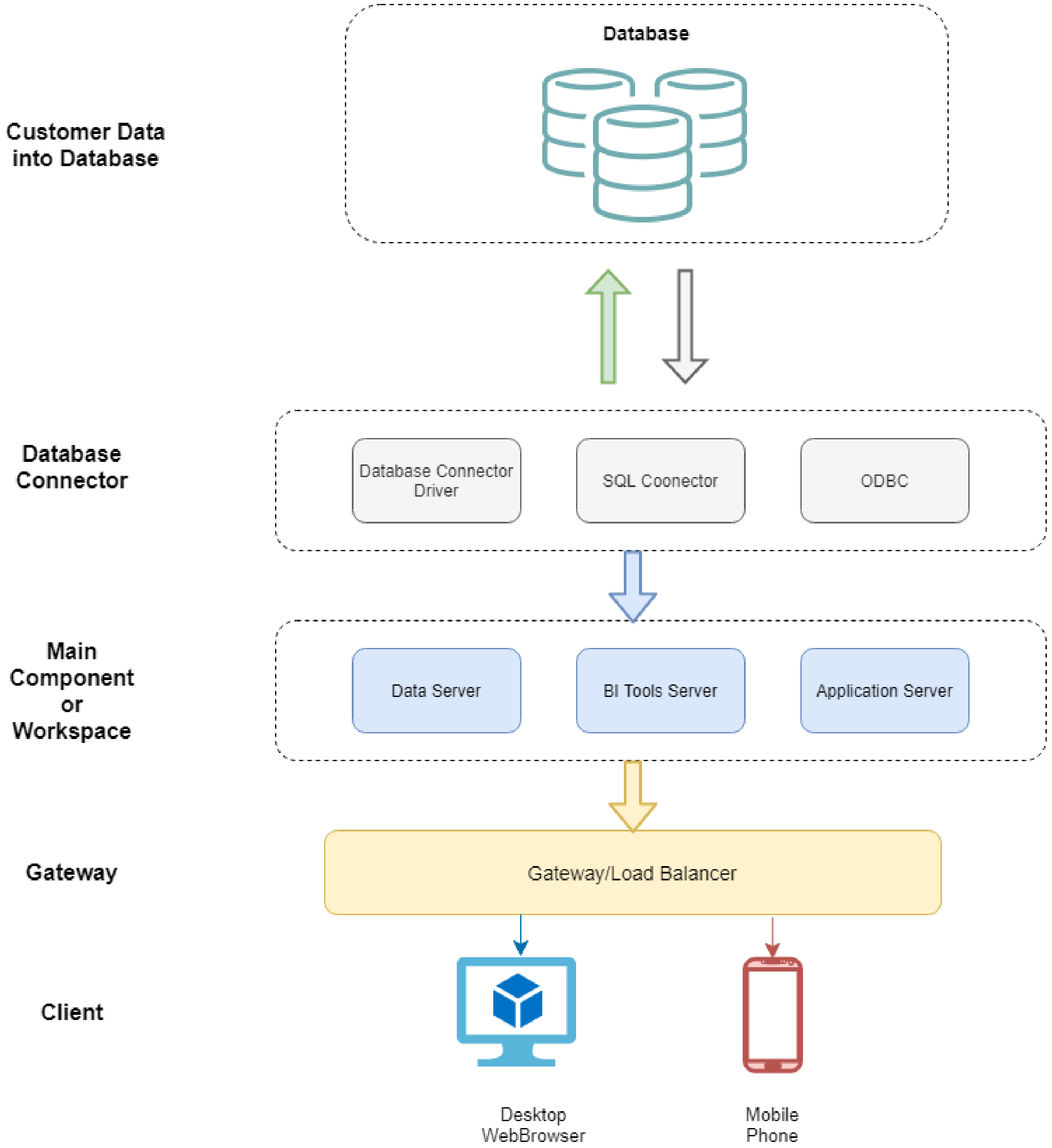
1. Introduction

1.1 Scope of the LLD:

The LLD system aims to present the internal functionality of the implemented system. The LLD documents will show the Dataset Description. Complete Architecture Description and working methodology of every component and tools are used to analyze Heart Diseases Analysis.

The LLD is the component-level design process that follows a step-by-step refinement process. The process can be used for design data structures, required software architecture, source code, and ultimately, performance algorithms, Overall, the data organization may be defined during requirement analysis and then refined during data design work.

2. Architecture



3. Architecture Description

3.1 Database

In the database, we will store our data and will extract it as per need from the cloud through the database connector.

3.2 Database Connector

Database connecter is software or it can be assumed as a driver that creates a connection between cloud database to BI tool where we want to pull the data. It allows the BI tool to extract data directly from the database if the authentication is provided. Then the BI tool has access to the data and we can do analysis on the data.

3.3 Main Component/Workspace

This is the main workspace where the data will be extracted and all the major work will be done here. It can be SQL server or Online query editor or data analysis platform also can be BI tools like tableau in this case. Here we will store the data in memory and perform the analysis.

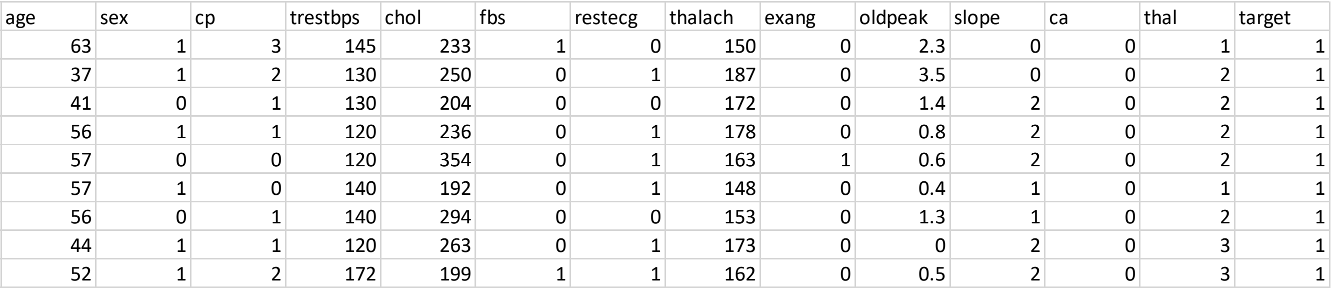
3.4 Gateway/Load Balancer

The cloud platform where the final application will be hosted. We will use Tableau Public for hosting our data.

4. Dataset Description

The data that will be used for the analysis is Heart Diseases Analysis founded on UCI Repository. A short description of the data is listed below.

4.1 Sample Data



age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
44	1	1	120	263	0	1	173	0	0	2	0	3	1
52	1	2	172	199	1	1	162	0	0.5	2	0	3	1

A Portion of data has been shown The dataset consist of **300 rows** and **14 columns** define an individual patient's health condition by targeting the chances of having heart diseases of that patient.

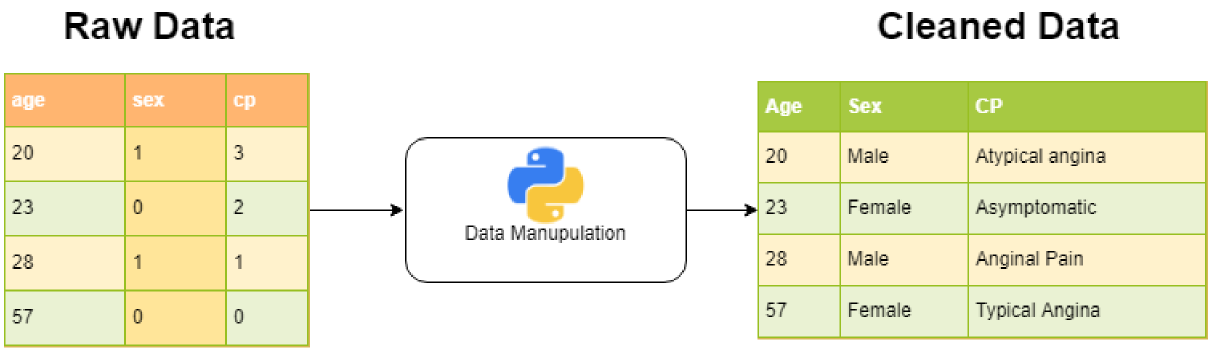
4.2 Dataset Description

Column Name	DataTypes	Description
age	Int	Age of Individuals
sex	Int	Gender of Individuals
cp	Int	Chest pain types

trestbps	Int	Resting Blood Pressure of individuals
chol	Int	Serum Cholesterol
fps	Int	Fasting Blood Sugar
restecg	Int	Resting ECG
thalach	Int	Maximum Heart Rate Achieved
exang	Int	Exercises Include Angina
oldpeak	Int	ST depression induced by exercise relative to rest
slope	Int	the slope of the peak exercise ST segment
ca	Int	Number of major vessels
thal	Int	Type of defect
target	Int	Having Heart Diseases or Not

4.3 Data Transformation/Manipulation

The dataset provided is not ready to start analysis, Data cleaning and Numerical Encoding are required for further analysis. We will use the python programming language for doing the data manipulation. In the Raw data, every data entry is an integer type and boolean type with 0 and 1. Finding meaningful insight and find the relationship between a patient's health report to chances of having heart disease. We will need further research for finding information behind the numerical encoding and using the python library we will transform raw data to encoding data that will be suitable for heart diseases analysis.



Raw Data

age	sex	cp
20	1	3
23	0	2
28	1	1
57	0	0

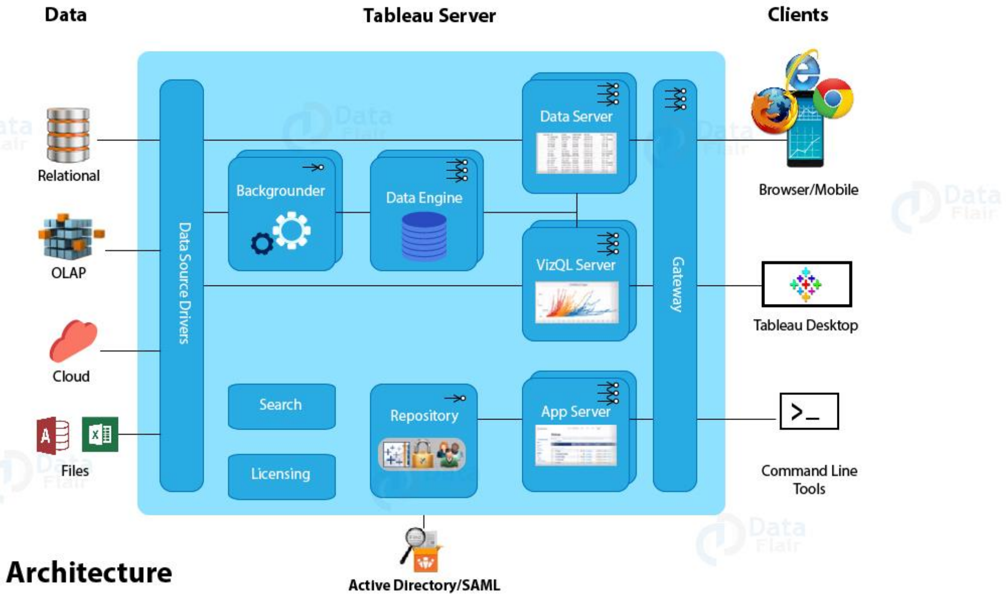
Cleaned Data

Age	Sex	CP
20	Male	Atypical angina
23	Female	Asymptomatic
28	Male	Anginal Pain
57	Female	Typical Angina

4.4 Data Insertion into Database

We will use Datastax Astra Database for our cloud data storage. We will export our data into a cloud database and will export it into the local system or directly can connect it to the BI tool we will use to Visualize the data for further analysis. The database is a flexible cloud storage solution and can access data from anywhere if having credentials for access. We will import data from the cloud to our BI tool for creating Dashboard and Analysis. The following architecture will describe the relation between clod and Client application for the flow of data.

5. Tableau Server Components and Architecture-



5.1.Gateway-

Gateway is a kind of web-server that helps clients communicate to the server via HTTP or https. The server receives incoming client requests and directs them to the appropriate server for action. A gateway handles processes such as load balancing, traffic routing, URL rewriting, serving static files to clients, serving multi-thread processes etc.

5.2.Application Server-

The application server deals with login processes, domain authentication, data authorization, user or group permission management, content searches, etc. It works in close association with the server's repository and handles the data access operations.

5.3.Repository-

The repository in Tableau Server stores server metadata related to users, permissions, assignments, groups, and projects. Along with the metadata, it stores visualizations in flat files (TWS, TDS), and performance data for auditing.

5.4 VizQL Server- It loads visualizations to work on Tableau. It has an in-built caching for performance improvement and editing tasks. Whenever a user requests a visualization or wishes to update an existing one, the request received by VizQL is first converted into an SQL statement and sent down to the data sources via respective data source drivers. The requested data sent back from the data source comes to the VizQL server again, where it is processed with some final touches of additional calculations and sent to the user. Any new visualization coming from the data source is cached in the VizQL for further use.

5.5.Data Engine-

The data engine stores multiple TDEs (Tableau Data Extract) and can run on multiple servers maximum 2). It also attends multiple requests parallelly at a given point of time. The data engine hosts the piece of data in-memory extracted from the TDE upon getting a request from the user.

5.6.Backgrounder-

Backgrounder is a multi-process element that manages schedules for information refreshing and ensures proper functioning of the Tableau Server and Data Engine.

5.7.Data Server-

The data server helps in centralizing metadata management, driver deployment, and extract management. It also contributes to access control and serves as a proxy to the data sources. It hosts user queries and requests to prevent users from directly accessing the data source.

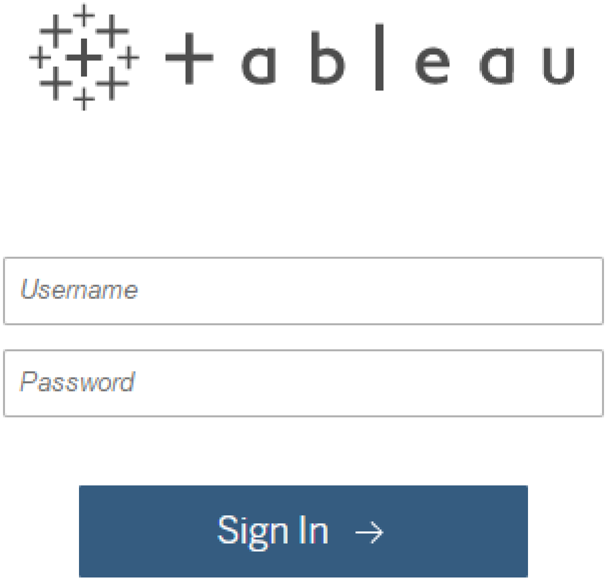
5.8.Search and License-

Two other important components are search and license. The search component manages the search indexing for the data in the repository. Whereas, the license component is responsible for the licensing and configuration of the Tableau server. Both these services run on the primary server of the Tableau's server cluster.

6. Deployment

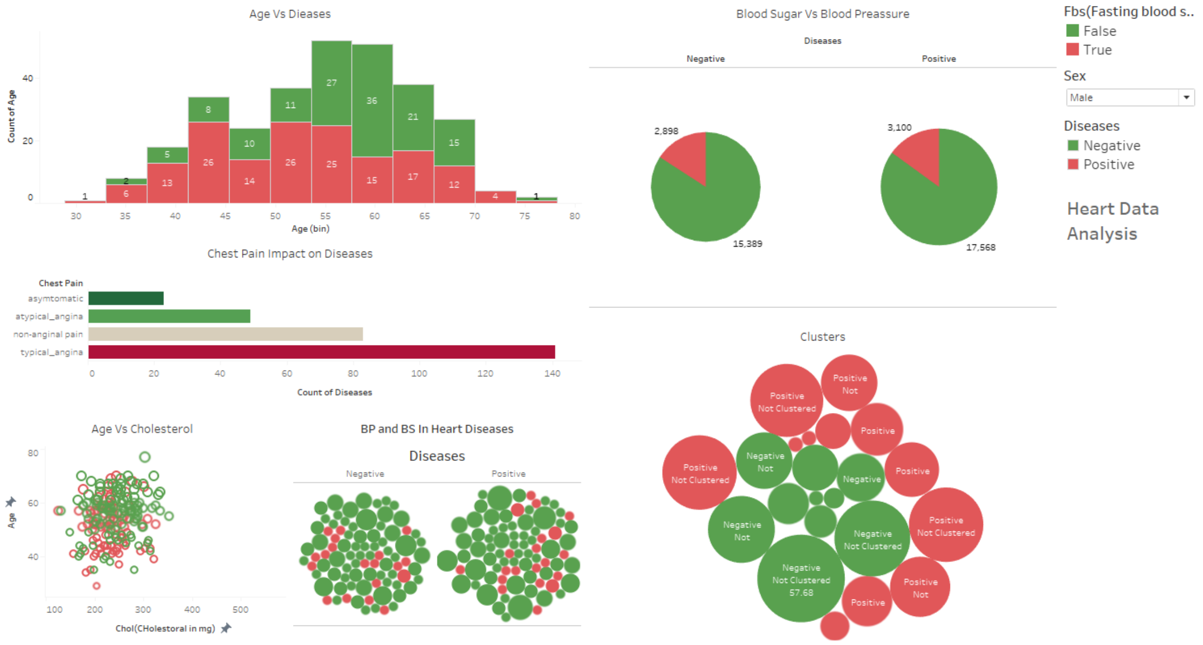
We can deploy our dashboard directly from our workspace, we must have registered on Tableau Public before that. And after saving the file it will automatically deploy on Tableau Public Server.

Steps 1 – Login



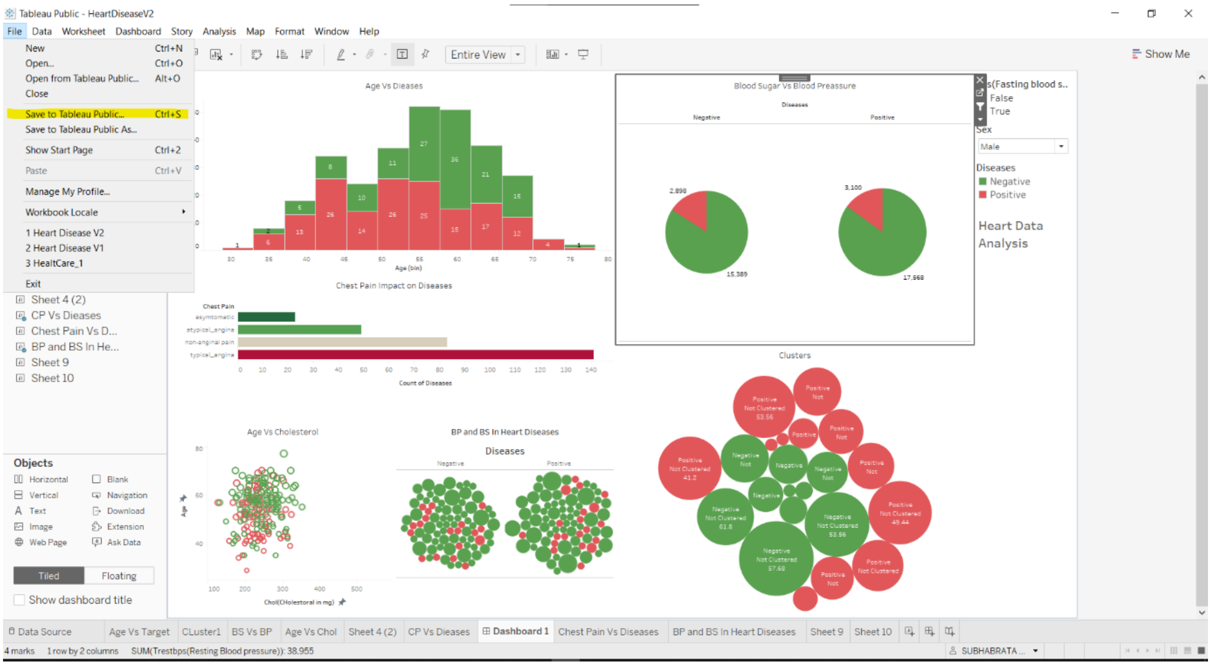
The user has to enter his registered email address and password for the login.

Step 2 – Building the Dashboard.



Users need to Build the Dashboard before Deploying on tableau public. And have to save the dashboard on the tableau workspace. We have to select and have to present on the desire dashboard or worksheet that we want to deploy.

Step 3 – Save the Dashboard from Tableau Workspace.



Step 4 – Save the Dashboard

After click on the save the Tableau dashboard will save and deploy on the tableau public server.