



# HIGH LEVEL DESIGN(HLD)

Healthcare Analytics on Heart Disease Data

Revision Number: 1.0

Last Date of Revision:18/11/21

-Nabeela Samar

# Document Version Control

Date issued	Version	Description	Author
18/11/21	1.0	First Version of Complete HLD	Nabeela Samar

# Contents

Document Version Control

Abstract

1. Introduction

1.1 Why this HLD Document?

1.2 Scope

2. General Description

2.1 Problem Statement

2.2 Tools used

3. Design Detail

3.1 Functional Architecture

3.2 Optimization

4. Key Performance Indicators

4.1 KPIs

5. Deployment

# Abstract

The term Heart Disease covers all possible heart disorders .Heart Diseases have become a major concern to deal with as studies show that the most of the deaths are caused due to heart diseases.

Thus prevention of Heart Disease has become necessary . Good data driven systems for predicting heart diseases can improve the entire research and prevention process , making sure that more people can live healthy lives.

# 1. Introduction

## 1.1 Why this High-Level Design Document

- The purpose of this HLD Document is to add the necessary detail to the current project description to represent a suitable model for coding.
- This document is also intended to help detect contradiction prior to coding , and can be used as a reference manual for how the modules interact at a high level

THE HLD will:

- Present all of the design aspects and define them
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:
  - Security
  - Reliability
  - Maintainability
  - Portability
  - Reusability
  - Resource Optimization
  - Serviceability

## 1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

## 2. General Description

### 2.1 Problem Statement

The goal of this project is to analyse the occurrence of heart disease in a person based on a combination of features that describe the heart disease. A dataset is formed taking into consideration information of some individuals.

Find key metrics and factors and show the meaningful relationships between attributes of the dataset.

### 2.2 Tools used



# 3. Design Details

## 3.1 Functional Architecture



### STEP 1

Data from source systems is integrated and loaded into a data warehouse or other analytics repository.



### STEP 2

Data sets are organized into analytics data models or OLAP cubes to prepare them for analysis.



### STEP 3

BI analysts, other analytics professionals and business users run analytical queries against the data.



### STEP 4

The query results are built into data visualizations, dashboards, reports and online portals.



### STEP 5

Business executives and workers use the information for decision-making and strategic planning.

# 3. Design Details

## 3.2 How BI Works

### ORGANIZATIONAL MEMORY

- Data Warehouse
- Enterprise resource planning (ERP)
- Knowledge Repository
- Content Management System (CMS)

### INFORMATION INTEGRATION

- Business Analytical Tools
- Data Mining
- Real Time Decision

### INSIGHT CREATION

- Text Mining Tool
- Web Mining Tool
- Environmental Scanning
- RFID

### ORGANIZATIONAL MEMORY

- Online Analytical Processing (OLAP) Tool
- Visualization Tool
- Digital Dashboard
- Score Card



## 3.2 Optimization

### **Your data strategy drives performance**

- Minimize the number of fields
- Minimize the number of records
- Optimize extracts to speed up future queries by materializing calculations, removing columns and the use of accelerated views

### **Reduce the marks (data points) in your view**

- Practice guided analytics. There's no need to fit everything you plan to show in a single view. Compile related views and connect them with action filters to travel from overview to highly-granular views at the speed of thought.
- Remove unneeded dimensions from the detail shelf.
- Explore. Try displaying your data in different types of views.

### **Limit your filters by number and type**

- Reduce the number of filters in use. Excessive filters on a view will create a more complex query, which takes longer to return results. Double-check your filters and remove any that aren't necessary.
- Use an include filter. Exclude filters load the entire domain of a dimension, while include filters do not. An include filter runs much faster than an exclude filter, especially for dimensions with many members.
- Use a continuous date filter. Continuous date filters (relative and range-of-date filters) can take advantage of the indexing properties in your database and are faster than discrete date filters.
- Use Boolean or numeric filters. Computers process integers and Booleans (t/f) much faster than strings.
- Use parameters and action filters. These reduce the query load (and work across data sources).

## Optimize and materialize your calculations

- Perform calculations in the database
- Reduce the number of nested calculations.
- Reduce the granularity of LOD or table calculations in the view. The more granular the calculation, the longer it takes.
  - o LODs - Look at the number of unique dimension members in the calculation.
  - o Table Calculations - the more marks in the view, the longer it will take to calculate.
- Where possible, use MIN or MAX instead of AVG. AVG requires more processing than MIN or MAX. Often rows will be duplicated and display the same result with MIN, MAX, or AVG.

Make groups with calculations. Like include filters, calculated groups load only named members of the domain, whereas Tableau's group function loads the entire domain.

- Use Booleans or numeric calculations instead of string calculations. Computers can process integers and Booleans (t/f) much faster than strings.  
Boolean>Int>Float>Date>DateTime>String

## 4. KPI

Dashboards will be developed to display and indicate certain KPIs and relevant indicators for the heart disease. As and when the system starts to capture the historical data the dashboards will be included to display charts over time with progress on various indicators or factors.



### 4.1 KPIs

Key indicators displaying a summary of the Heart Disease and its relationship with different metrics

1. Percentage of People of having Heart Disease
2. Ag Distribution
3. Gender Distribution
4. Chest Pain Experienced by People
5. Blood Pressure, Heart Rate, Fasting Blood Sugar

# 5. Deployment

Prioritizing data and analytics couldn't come at a better time. Your company, no matter what size, is already collecting data and most likely analyzing just a portion of it to solve business problems, gain competitive advantages, and drive enterprise transformation. With the explosive growth of enterprise data, database technologies, and the high demand for analytical skills, today's most effective IT organizations have shifted their focus to enabling self-service by deploying and operating Tableau at scale, as well as organizing, orchestrating, and unifying disparate sources of data for business users and experts alike to author and consume content.

Power BI prioritizes choice in flexibility to fit, rather than dictate, your enterprise architecture. Power BI Desktop and Power BI Service leverage your existing technology investments and integrate into your IT infrastructure to provide a self-service, modern analytics platform for your users. With on-premises, cloud, and hosted options, there is a version of Power BI to match your requirements.

