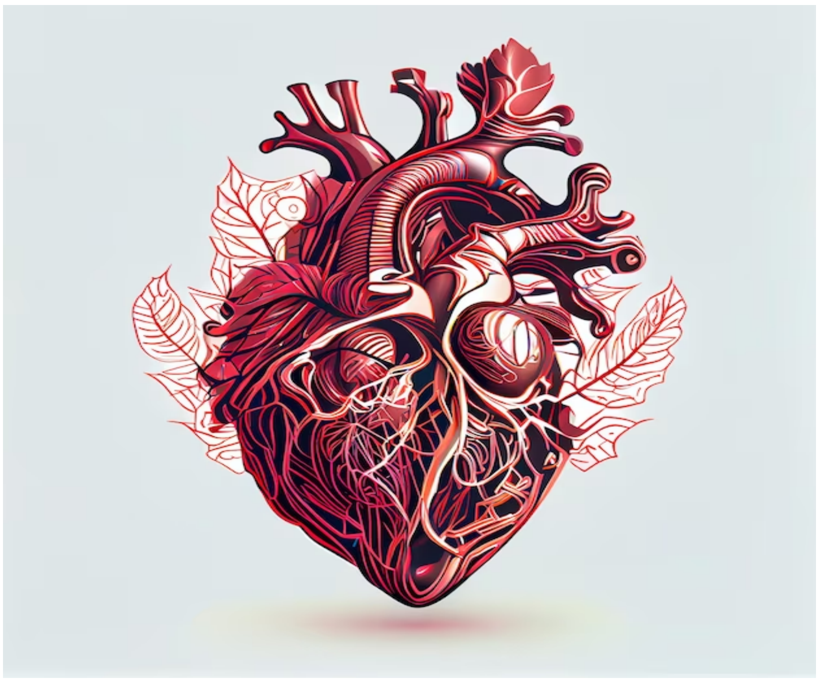


High-Level Design (HLD)
Heart Disease Diagnostic Analysis

Revision Number: 1.1

Last Date of Revision: 23/9/2023

Sameer Singh

Document Version Control

Date Issued	Version	Description	Author
22.09.2023	1.0	Abstract, Introduction, General Description, Design Detail	Sameer Singh
23.09.2003	1.1	KPI, Deployment, Final Revision	Sameer Singh

Contents

Abstract

The term 'Heart Disease' refers to any disorder related to heart. Heart diseases are the leading cause of death and disability in the world especially South-Asian region including India. As of the year 2017, heart disease was responsible for 26.6% (25.3%–27.4%) of total deaths in India, compared with 15.2% (13.7–16.2) in 1990. The India State-level disease burden study of the GBD study group reports that there has been a 2.3-fold increase in heart disease in the country between 1990 and 2016.

Thus, nowadays preventing heart disease become very necessary. Good data-driven systems for predicting heart diseases can improve the prevention process which can be helpful in preventing people from heart-related diseases.

1 Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (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 contradictions 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 in detail
- 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
    - Application compatibility
    - Resource utilization
    - 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 Product Perspective & Problem Statement

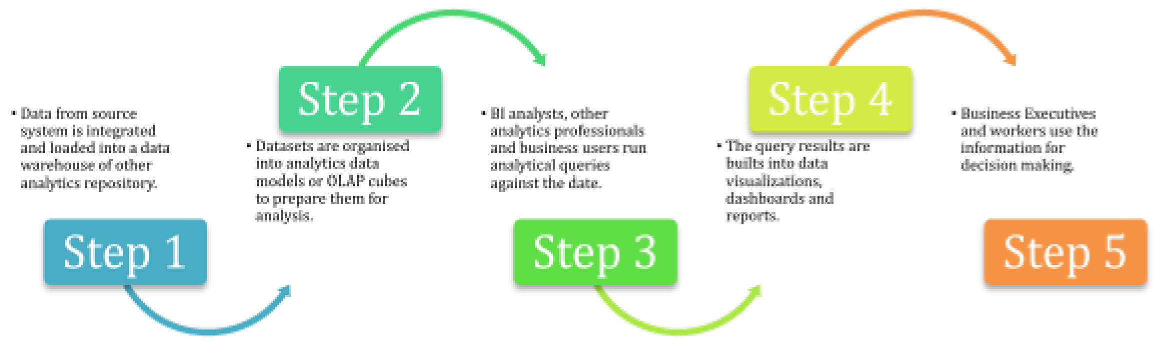
The goal of this project is to predict the probability of heart disease occurrence, based on a combination of features that describes the disease. To achieve the goal, we used a data set that is formed by taking into consideration the information of 303 individuals. Based on the given information we have to calculate that whether the individual will suffer from heart disease or not.

2.2 Tools used

Business Intelligence tools and libraries works such as Numpy, Pandas, Excel, Matplotlib, Seaborn, Power BI, Jupiter Notebook and Python programming are used to build the whole framework.

3 Design Details

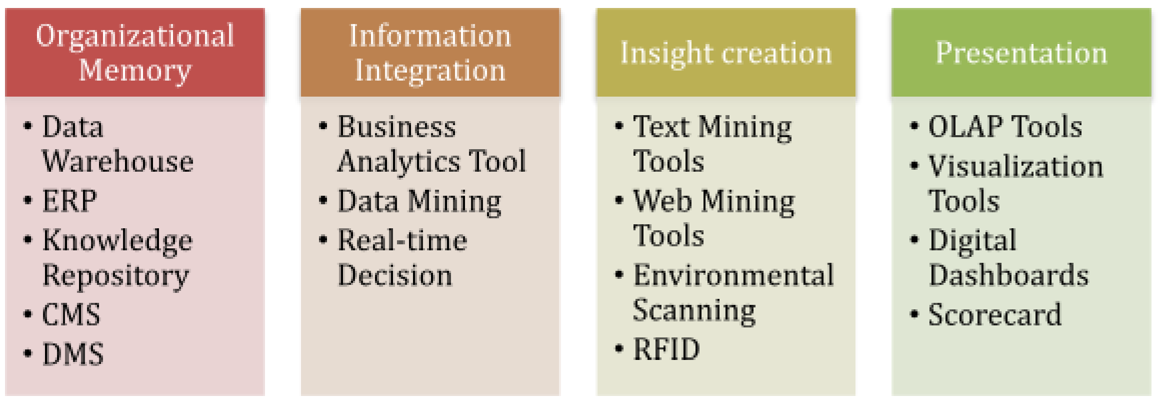
3.1 Functional Architecture



- Data from source system is integrated and loaded into a data warehouse of other analytics repository.
- Datasets are organised into analytics data models or OLAP cubes to prepare them for analysis.
- BI analysts, other analytics professionals and business users run analytical queries against the date.
- The query results are builts into data visualizations, dashboards and reports.
- Business Executives and workers use the information for decision making.

Figure: Functional architecture of Business Intelligence

How BI really works



Organizational Memory	Information Integration	Insight creation	Presentation
• Data Warehouse • ERP • Knowledge Repository • CMS • DMS	• Business Analytics Tool • Data Mining • Real-time Decision	• Text Mining Tools • Web Mining Tools • Environmental Scanning • RFID	• OLAP Tools • Visualization Tools • Digital Dashboards • Scorecard

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 including 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 KPIs

Dashboards will be implemented to display and indicate certain KPIs and relevant indicators for the disease.

Power BI

As and when, the system starts to capture the historical/periodic data for a user, the dashboards will be included to display charts over time with progress on various indicators or factors.

4.1 KPIs (Key Performance Indicators)

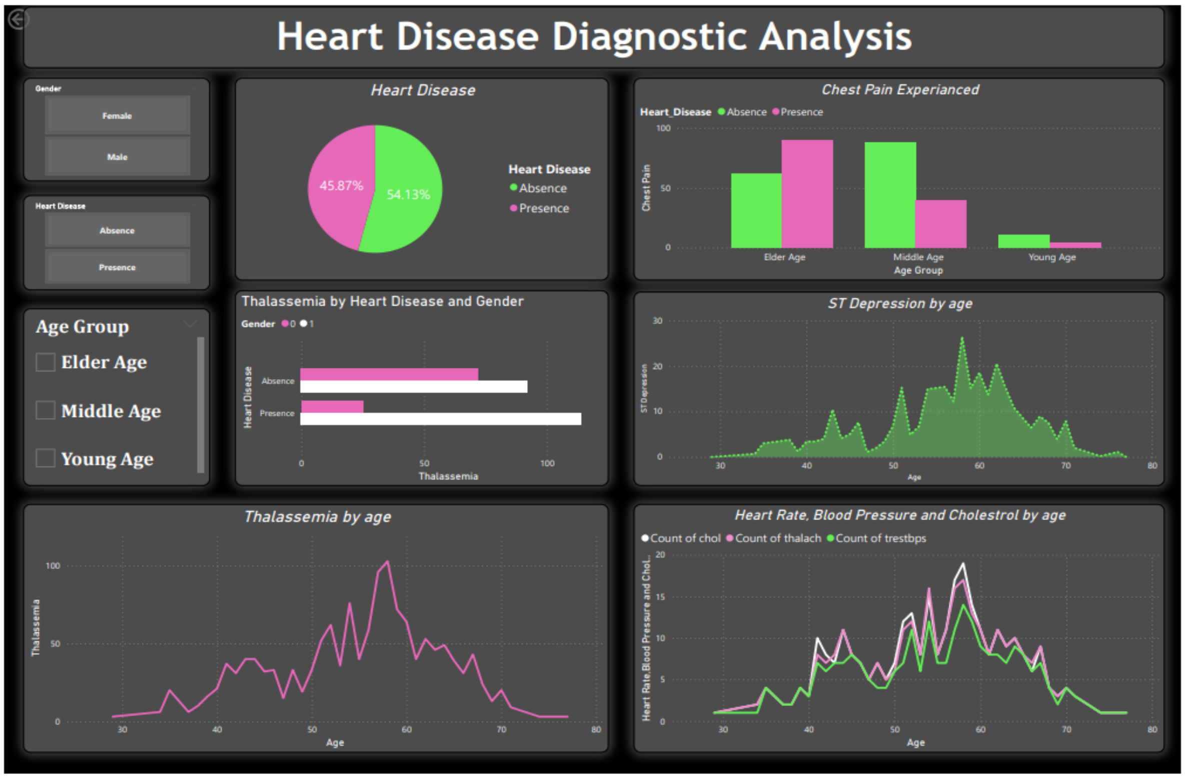
Key indicators displaying a summary of the Heart Disease Diagnostic Analysis and its relationship with different metrics.

- Percentage of people having Heart Disease.
- Age and Gender distribution based on Heart Diseases.
- Chest Pain experienced by the Heart Disease patients.
- ST Depression experienced by the people.
- Presence of Thalassemia in Heart Disease patients.
- Heart Rate, Blood Pressure and Cholestrol level of people as per their age.

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.

PowerBI prioritizes choice in flexibility to fit, rather than dictate, your enterprise architecture. PowerBI leverage your existing technology investments and integrates 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 Tableau to match your requirements.


Heart Disease Diagnostic Analysis
Gender
Female
Male
Heart Disease
Absence
Presence
Age Group
Elder Age
Middle Age
Young Age
Heart Disease
45.87%
54.13%
Heart Disease
● Absence
● Presence
Chest Pain Experianced
Heart_Disease ● Absence ● Presence
Chest Pain
Elder Age
Middle Age
Young Age
Age Group
Thalassemia by Heart Disease and Gender
Gender ● 0 ● 1
Heart Disease
Absence
Presence
Thalassemia
ST Depression by age
ST Depression
Age
Thalassemia by age
Thalassemia
Age
Heart Rate, Blood Pressure and Cholestrol by age
● Count of chol ● Count of thalach ● Count of trestbps
Heart Rate,Blood Pressure and Chol...
Age