High Level Design (HLD)
Heart Disease Diagnostic Analysis

AAKASH BHUTE

Contents

Document Version Control……2

Abstract……3

1. Introduction……4

1.1 Why this High-Level Design Document?.....4

1.2 Scope……4

2. General Description……..5

2.1 Product Perspective & Problem Statement……5

2.2 Tools Used……5

3. Design Detail……..…...6

3.1 Functional Architecture……...6

3.2 Optimization……7

4. KPI……8

4.1 KPIs (Key Performance Indicators) ……..……....9

5. Deployment…….....9

Abstract

Heart disease is a broad term encompassing various disorders affecting the heart. In recent decades, the prevalence of heart diseases has become a significant concern, particularly in India, where it has emerged as the leading cause of death. According to a study, there has been a substantial increase in the number of deaths attributed to heart diseases in India. From 1990 to 2016, the mortality rate due to heart diseases has risen by approximately 34%, escalating from 155.7 to 209.1 deaths per 1 lakh population.

Heart disease is an umbrella term that encompasses a range of conditions impacting the heart. Over the past few decades, the prevalence of heart diseases has become a major worry, specifically in India, where it has taken the forefront as the primary cause of death. Research indicates a noteworthy surge in the number of deaths linked to heart diseases in India. Between 1990 and 2016, the mortality rate due to heart diseases witnessed an approximate 34% increase, with the number of deaths per 1 lakh population rising from 155.7 to 209.1.

1 Introduction

1.1 Why this High-Level Design Document?

The objective of this High-Level Design (HLD) Document is to provide additional elaboration to the existing project description, creating a comprehensive blueprint that can serve as a suitable model for coding. This document is designed to identify any inconsistencies or conflicts prior to the coding phase and can serve as a reference guide for understanding the high-level interactions between modules.

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 outlines the system's framework, including the database architecture, application architecture (layers), application flow (navigation), and technology architecture. It employs a language that ranges from non-technical to mildly-technical terms, ensuring that system administrators can easily comprehend the content.

2 General Description

2.1 Product Perspective & Problem Statement

The objective of this project is to analyze and predict the likelihood of heart disease occurrence by utilizing a combination of descriptive features related to the disease. Subsequently, we develop a Business Intelligence (BI) report that illustrates the influential factors contributing to the disease's occurrence, along with the magnitude of their impact. To accomplish this objective, we utilized a carefully curated dataset comprising information from 303 individuals. The problem at hand involves determining, based on the available individual information, whether an individual is prone to suffer from heart disease or not.

2.2 Tools used

Business Intelligence tools and libraries such as NumPy, Pandas, Seaborn, Matplotlib, MS-Excel, MS-Power BI, Jupyter Notebook and Python Programming Language are used to build the whole framework.

• Jupyter Notebook is used as IDE.

• Python is the Programming Language used.

• EDA is done using Numpy & Pandas.

• Visualizations were done using seaborn.

• Power BI is used for dashboard creation.

3 Design Details

3.1 Functional Architecture

Figure 1: Functional Architecture of Business Intelligence

Constraints

Our analysis is conducted using a restricted dataset that focuses on specific (15) features known to influence heart disease. It's important to note that the analysis does not incorporate external factors such as underlying medical conditions, medication usage, lifestyle choices, BMI value, and other related variables.

Assumptions

It is a task that is trivially performed by doctors, however the use of past cases and the potential cause for such cases has enabled data analysts to create a segment of symptoms/causes belonging to which significantly increases a risk of a potential heart disease in the future.

Our analysis assumes that all the data provided was true without any corruption and the features mentioned in the raw dataset are the only driving factors of a potential heart disease.

.

Design Details
Functional Architecture

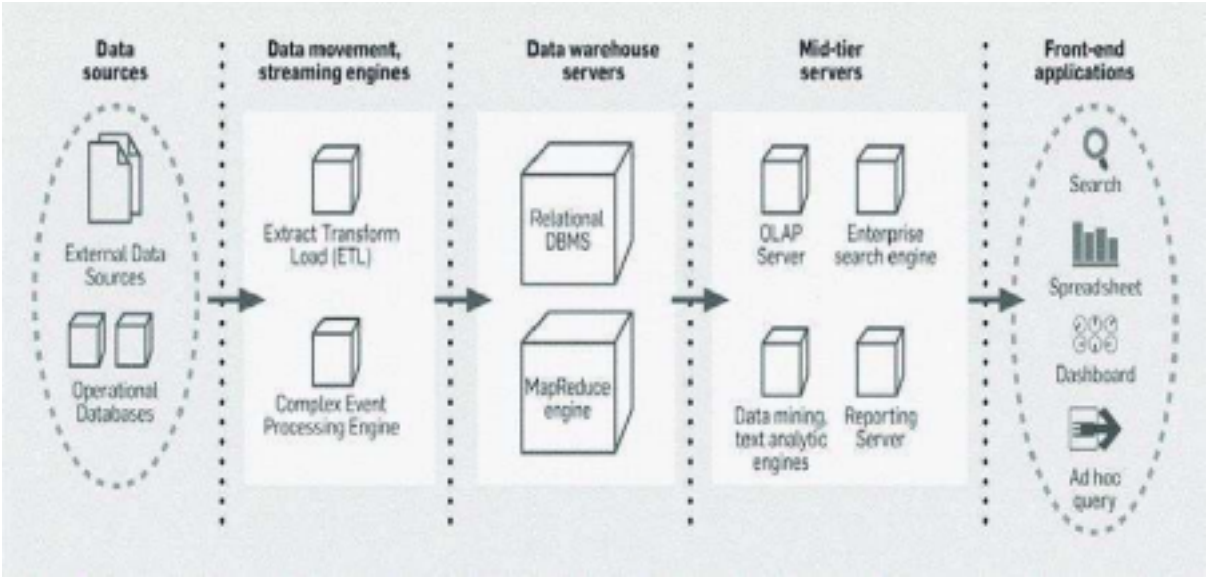

Figure 1: Functional Architecture of Business Intelligence

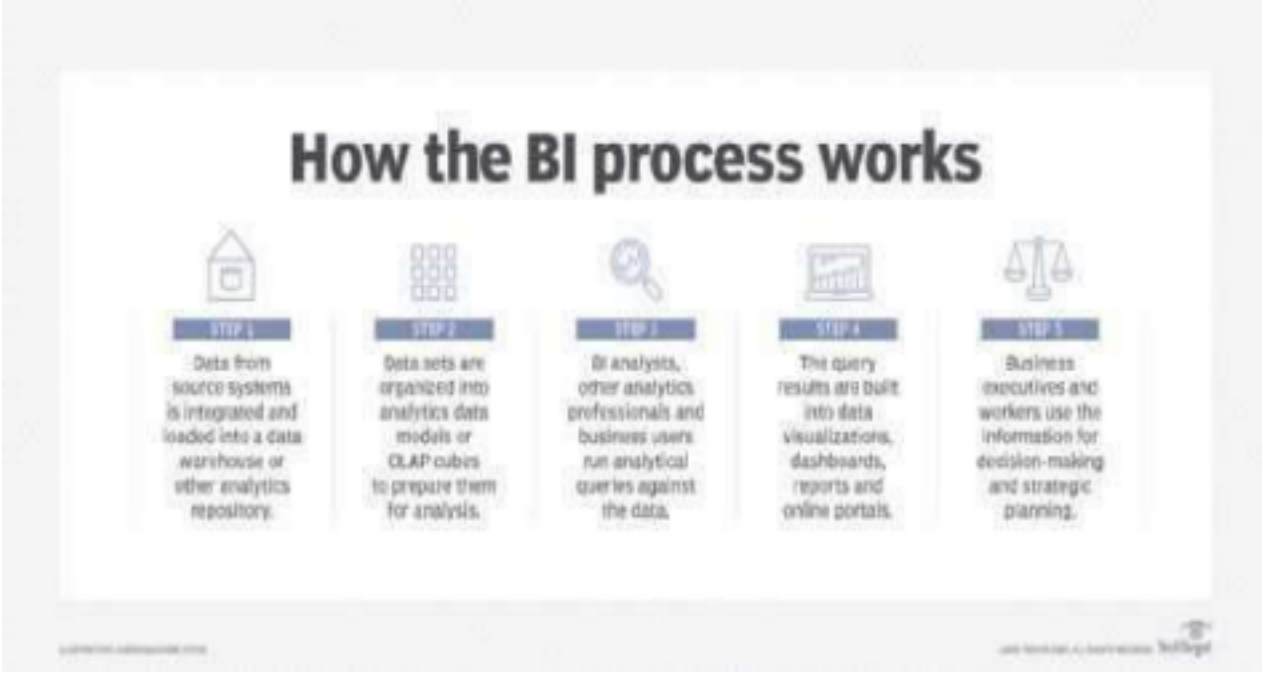

Figure 2: Working of BI process

BI Reporting Architecture

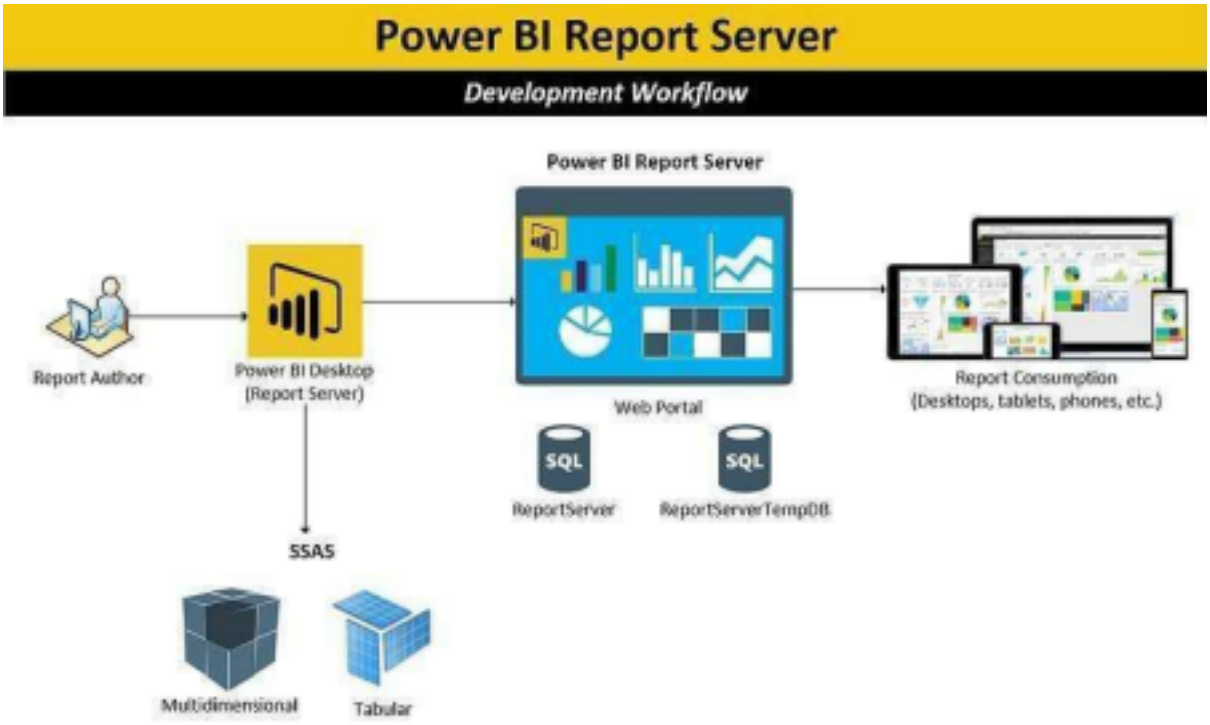

Figure 2: BI Reporting architecture

3.2 Optimization

1. 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

2. 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.

3. 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 including 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-ofdate filters) can take advantage of the indexing properties in your database and are faster than discrete data 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).

Performance

Healthcare analytics determines the presence or absence of a life threatening condition, it should be as accurate as possible. So that it will not mislead the user. Also, model retraining is very important to improve the performance.

Security

Since the Health care analysis consists of patient's data, the information should be secured.

Reusability

The code written and the components used should have the ability to be reused with no problems.

Resource utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

4 KPI

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

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)

Here are the key indicators summarizing the Heart Disease Analysis and its relationship with various metrics:

1. Percentage of People with Heart Disease
2. Age Distribution and Gender
3. Gender Distribution among Heart Disease Patients
4. Chest Pain among Heart Disease Patients across Age Categories
5. Blood Pressure, Cholesterol Level, and Maximum Heart Rate of Heart Disease Patients
6. ST Depression in relation to age and heart disease
7. ECG Measurement for Heart Disease Patients

These indicators provide a concise overview of the analysis and its findings related to heart disease and its associated metrics.

5 Deployment

Power BI emphasizes providing flexibility to align with your enterprise architecture rather than imposing restrictions. Both Power BI Desktop and Power BI Service seamlessly integrate with your existing technology investments, enabling a self-service analytics platform that suits your users' needs. Whether you prefer on-premises, cloud, or hosted options, Power BI offers a version that matches your specific requirements.

The Dashboard is published on the Power BI service and has been configured with an auto-refresh mode, ensuring that it continuously updates as real-time data loads into the log file.

The Dashboard showcases various insights derived from the log files, including:

1. Summary: This section of the dashboard provides a statistical overview of the dataset and visually represents the impact of different features on heart disease.

2. Critical Factors: Within this dashboard, the most significant feature contributing to the presence of heart disease is identified, along with the magnitude of its effect.

3. Patient Profile: This section presents a patient's profile who has been diagnosed with a heart disease, displaying average age, blood pressure level, and other relevant information.