HIGH LEVEL DESIGN

FIFA WORLD CUP ANALYSIS

SHUBHAM MEHAR

Version 1.0

Date- 11/12/2022

Project Title	**FIFA World Cup**
Technologies	**Business Intelligence**
Domain	**Sports**
Project Difficulties Level	**Advanced**

Document Version Control

Date Issued	Version	Description	Author
11/12/2022	1.0	Complete HLD	Shubham Mehar

INDEX

Sr. No.	CONTENT	Pg. No.
1.	Abstract	5
2.	High Level Design Document	6
3.	General Description	7
4.	TOOL USED	8
5.	DESIGN DETAILS (for Dashboard)	8
6.	Power BI Architecture	9
7.	Optimization	10-11
8.	Key Performance Indicators	11
9.	Deployment	12
10.	References	12-13

1. Abstract

The objective of this article is to explore the FIFA World Cup data and find the insights and trends. The **FIFA World Cup**, often simply called the **World Cup**, is an international association football competition contested by the senior men's national teams of the members of the *Fédération Internationale de Football Association* (FIFA, the International Federation of Association Football), the sport's global governing body. The championship has been awarded every four years since the inaugural tournament in 1930, except in 1942 and 1946 when it was not held because of the Second World War. The World Cups dataset show all information about all the World Cups in the history, while the World Cup Matches dataset shows all the results from the matches contested as part of the cups.

2. High Level Design Document

Commonly abbreviated as HLD, the high-level design is a general system of design. In simpler words, it refers to the overall design of a system and describes the overall architecture/ description of any application. It is also called system/ macro-level design.

A high-level design includes the overall description of a system architecture along with the design of its database and a brief description of its services, systems, platforms used, and the relationship between modules. A solution architect creates the HLD. It basically converts the overall client/ business requirements into a high-level solution. It comes into play before the low-level design.

Components of High-Level Design

In order to describe a bird's eye view of the overall solution, High level design should include two key components –

1. Functionalities and properties/attributes of components
2. Interactions and relations between various components.

Purpose of High-Level Design (HLD)

The purpose of this High-Level Design (HLD) Document is to add the necessary detail description to represent a suitable model. This document is designed to help in operational requirement and can be used as a reference manual for how the modules interact. Basically, HLD is a technical representation of functional requirements and flow of information across assets or components.

During the preliminary stages of a development, the need of the project is to identify those parts of the project that might be at risk or time consuming. HLD provides a brief description of how the various sub-systems and components of the system fit together.

Scope of HLD

The High-Level Design documentation presents the structure of the system as the application/database architecture, application flow and technology architecture. High-Level Design documentation may use some non-technical terms unlike Low Level design which should be strictly technical jargon.

Characteristics of HLD

- HLD presents all of the design aspects (taken from business requirements and expected outcome) and defines them in form of a diagram.
- It describes the user interface being implemented and description of hardware and software interfaces.
- It describes the performance requirements and flow of user's daily process.
- HLD includes design features and the architecture of the project.

Requirement of HLD

HLD briefly describes about the platforms/products/services/processes, flow of traffic that it depends on and includes any important changes that need to be made to them. HLD is the input for creating the LLD (Low Level Design) since the key communication items are displayed in HLD which are then converted to detailed communication in LLD, showing connectivity and physical level. HLD requires contribution from a number of experts, representing many distinct professional disciplines like SME, Design Architectures. Every type of end-user should be classified and identified in the HLD and each contributing design should give due consideration to the customer experience. Another important aspect is that HLD and LLD are source of reference in Operations stage also, after the project has been successfully implemented and now in BAU stage. Hence, design phase is of utmost importance.

3. General Description

Project Perspective

The World Cup is the most prestigious association football tournament in the world, as well as the most widely viewed and followed single sporting event in the world.

The project perspective is to find the insights from the given datasets. The dataset has FIFA world cup data from 1930-2014. There are three different datasets are given we have to use this data to find the insights using visualisations, tables, charts and to made it dynamic so we can check it easily using filters and slicers.

Problem Statement

The World Cups dataset show all information about all the World Cups in the history, while the World Cup Matches dataset shows all the results from the matches contested as part of the cups. Also give World Cup Players in which the data given of players who participated for their team.

- From the given datasets we have to find stats of FIFA world cup from 1930-2014.
- To clean the dataset where the dataset had spelling mistakes and correct them.
- To check data type and correct where it is required.
- Dropping unused columns an rows
- To create new columns and new measures required for finding the in-depth facts from the given datasets.
- To create different dashboard so, we can check the data for particular point and focus on it like team, player, stadium etc.

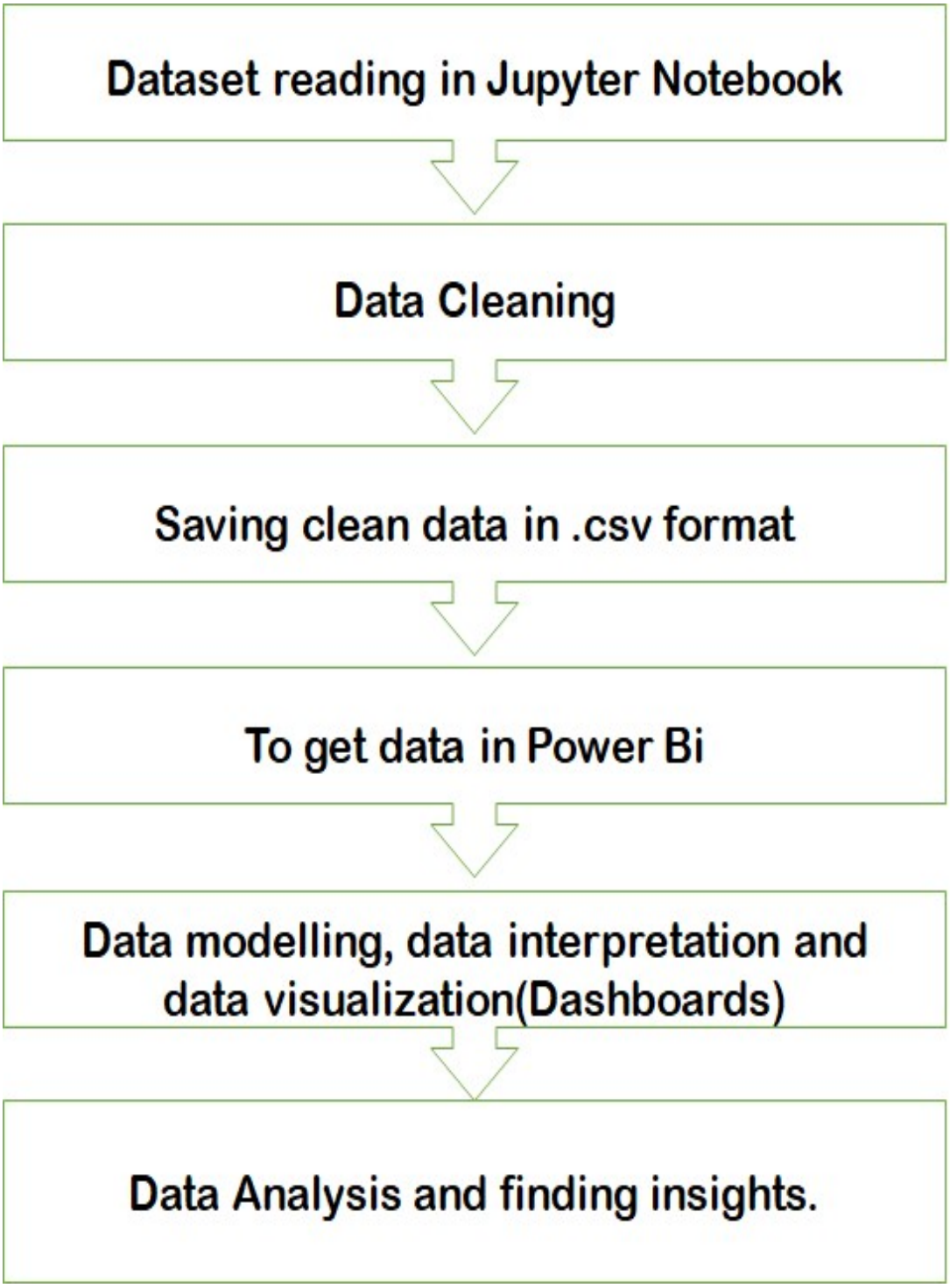
4. TOOL USED

Jupyter notebook for data cleaning, fixing error and data type corrections using python libraries such as NumPy and Pandas.

Business intelligence tool Power Bi will be used for Data Modelling, data interpretation and data visualization. To find insights from the given datasets.

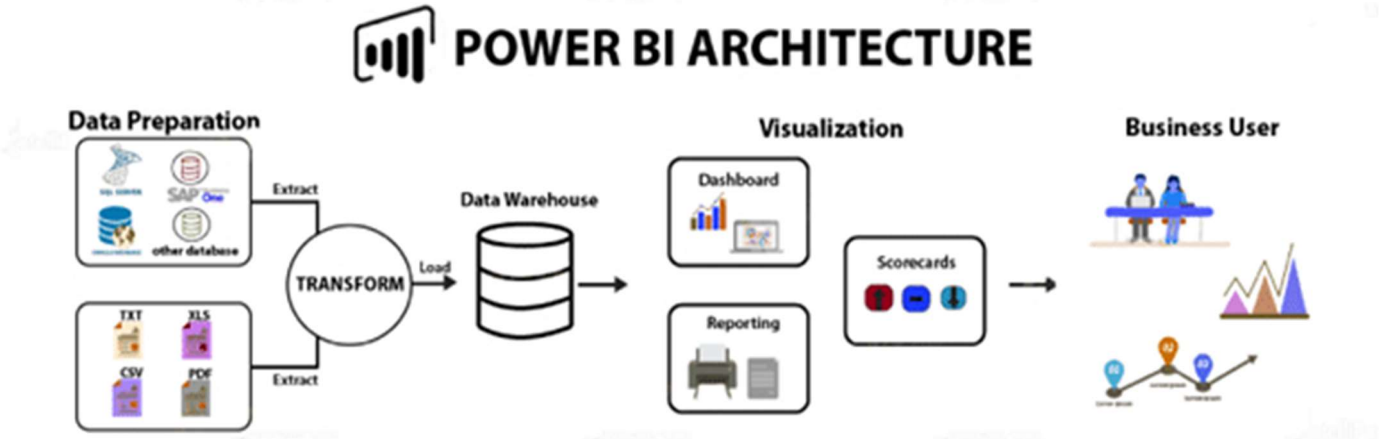
5. DESIGN DETAILS (for Dashboard)

Flowchart



6. Power BI Architecture

MS Power BI architecture consists of four major steps that explain the whole process from data sourcing to the creation of reports and dashboards. Various technologies and processes work together to get the required results with extreme precision. Let's see those steps further.



- **Sourcing data:** Power BI extracts data from various servers, Excel sheets, CSV files, and databases. The extracted information can be directly imported to Power BI, or a live service link is established to receive it. If you directly import the data in Power BI, it will only be compressed up to 1 GB. Post that, you can only run live queries on your chunky datasets.
- Transforming the data: Before visualizing the data, cleaning and pre-processing it should be done. This means removing useless or missing values from rows or columns. Following that, certain rules will be applied to transform and load the datasets into the warehouse.
- **Report and publish:** After cleaning and transforming the data, reports will be created based on requirements. A report is a visualization of the data with different filters and constraints presented in the form of graphs, pie charts, and other figures.
- **Creating dashboards:** Power BI Dashboards are created by pinning individual elements or pages of live reports. Dashboards should be created after you have published your reports to the BI service. When the reports get saved, the visual maintains the filter settings chosen so that the user can apply filters and slicers.

7. Optimization

You can optimize your solution at different architectural layers. Layers include:

- The data source(s)
- The data model
- Visualizations, including dashboards, Power BI reports, and Power BI paginated reports
- The environment, including capacities, data gateways, and the network

Optimizing the data model

The data model supports the entire visualization experience. Data models are either external-hosted or internal-hosted, and in Power BI they are referred to as *datasets*. It's important to understand your options, and to choose the appropriate dataset type for your solution. There are three dataset modes: Import, Direct Query, and Composite. For more information, see Datasets in the Power BI service, and Dataset modes in the Power BI service.

Limit Data Model

The fastest way to optimize your Power BI report is to limit the number of columns to only the ones you need in your data model. Go through your tables in Power Query and determine what fields are being used. Delete these columns if they are not being used in any of your reports or calculations. Another way to limit your data model is to use Row Level Security when applicable.

A side note, limiting columns can be far more effective than limiting the rows in your data set. As an example, if you have a data set with 1M rows and 20 columns, removing 10 columns will remove 10M data points from your model. If you were to parse down the number of rows in half, you would only be eliminating 5M data points. While filter down your dataset to only the applicable data is a good idea, eliminating unused columns will be much more effective in minimizing your dataset.

Push Data Transformations Upstream

Building on the previous step, if you are able to move your data manipulations to your query, this will help your performance. Whether you're doing calculations, formatting, or grouping at the appropriate granularity in your SQL, this will eliminate Power

BI spinning its wheels on these items. If you end up doing your transformations in Power Query, try Query Folding.

Move Row Level Logic to Power Query

Move any logic that needs to be calculated at the row level to Power Query. Instead of using DAX to create calculated columns containing the more basic IF THEN calculations, use M to do the same thing. These fields will be included in your permanent data model brought in from Power Query instead of them being performed in Power BI Desktop.

Use Measures Instead of Calculated Columns

For beginning Power BI users, it can be tempting to create calculated columns instead of using measures as you're able to see the row-by-row output for each calculated column. The issue is that calculated columns can be more burdensome for the data model as they actually create a new column of data. Instead, measures are simply aggregations of the fields in the data model, so they do not actually add new data to your model. Using measures where applicable will help reduce data model size as well as increase computation efficiency.

Convert Multiple Measures to Variables

Instead of creating metrics that require using multiple measures, use variables. You can perform multiple calculations within a single measure using the VAR and RETURN functions. This will minimize the number of measures you use as well as increase performance for the metrics being calculated. If you are going to be repeating a variable multiple times then you can still create an independent measure for that value.

Amend Dates and Column Types

Although this is probably the least impactful of the ideas in the list, editing your data's formatting can also help limit model size and performance. As an example, date will show as mm/dd/yyyy 12:00:00am by default in Power BI. To cut down on characters in the data model, change the date types to Date to eliminate the Time portion. If you have integers with a large number of decimal points, reduce the decimal places showing. Lastly, if you have text showing TRUE or FALSE, you can change these to be binary, so 1 and 0.

Use Performance Analyzer to Analyse

Power BI has a built-in way for you to analyse the performance of your reports. In the View ribbon you can find the Performance Analyzer. By opening this pane and clicking

Record, this will show you how fast your report renders when performing different functions in your report. You can also see what specific sections of your report are causing performance delays so that you can work on improving those sections.

8. Key Performance Indicators

Key Performance Indicators (KPIs) are the critical (key) quantifiable indicators of progress toward an intended result. KPIs provide a focus for strategic and operational improvement, create an analytical basis for decision making and help focus attention on what matters most.

Managing with the use of KPIs includes setting **targets** (the desired level of performance) and tracking progress against those targets.

Managing with KPIs often means working to improve performance using **leading indicators**, which are precursors of future success, that will later drive desired impacts indicated with **lagging measures**.

9. Deployment

A **Deployment Pipeline** is the process of removing code from version control and making it automatically available to users of your application. When a team of developers is working on a project or feature, they need a reliable and efficient way to build, test, and deploy their work.

Power BI Deployment Pipelines allow creators to create and test content in the Power BI service before it is consumed by users. In simple terms, these Power BI Deployment Pipelines are in charge of managing the lifecycle of organizational content. Reports, paginated reports, dashboards, datasets, and dataflows are among the content types.

A Power BI Deployment Pipeline has three stages:

1) Development

This stage is used to collaborate with other creators to design, build, and upload new content. This is the initial stage of a Power BI deployment pipeline.

2) Test

After making the necessary changes, in the testing stage, you can upload the modified content in order for it to be moved to this testing stage. Some of the examples of what can be accomplished in a testing environment are as follows:

- Distribute content to testers and reviewers.
- Load and run tests on larger amounts of data.
- Test your application to see how it will appear to your customers.

3) Production

After testing the report content, you can move to the production stage. In this stage, you can share the final version of your content with the stakeholders across the organization during the production stage.

10. References

https://byjus.com/gate/difference-between-high-level-and-low-level-design/

https://ipwithease.com/what-is-a-high-level-design-hld/

https://learn.microsoft.com/en-us/power-bi/guidance/power-bi-optimization

https://intellipaat.com/blog/power-bi-architecture/

https://www.phdata.io/blog/optimizing-power-bi-reports/#:~:text=The%20fastest%20way%20to%20optimize,of%20your%20reports%20or%20calculations.

https://hevodata.com/learn/power-bi-deployment-pipelines/