

# High Level Design(HLD)

## FIFA 19 DATA ANALYTICS



Submitted By: [Abhishek Sachan](#)

Last Date of Revision:25/12/2022

# Content

## Abstract

<b>1. Introduction</b>	
1.1. Why this High-Level Design Document	<b>4</b>
1.2. Scope.	<b>4</b>
<b>2. General Description</b>	
2.1. Product Perspective & Problem Statement	<b>5</b>
2.2. Tools Used	<b>5</b>
<b>3. Design Details</b>	
3.1. Function Architecture	<b>6</b>
3.2. Optimization	<b>7</b>
<b>4. KPI</b>	
4.1. Key Performance Indicator	<b>9</b>
<b>5. Deployment</b>	<b>12</b>

## Abstract:

FIFA is one of the most popular football simulation video games of all times. I am sure that not only me but most of the people (I guess the majority would be males) probably played this game once in their life. Especially if you are a football fan, then you played this game not only once. I, as a football fan, have played FIFA's series for a long period of time, especially during my school years, and never got bored. :)

football fans also play football video games in order to maintain and live its excitement over and over again. EA's FIFA Football simulation video games have a very long history in the game industry and their FIFA series have been played by many users. Recently, while I was doing my research about finding a good dataset, I found FIFA 19 players dataset on Ineuron site and got very excited about it. Because I love football and my job in data space, I wanted to do my own analysis in this dataset.

# 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 help detect contradictions before coding and can be used as a reference manual for how the modules Interact at high level.

The HLD will:

- Present all 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

- o Reliability
- o Maintainability
- o Portability
- o Reusability
- o Application compatibility
- o Resource utilization
- o 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 to analyzed to top 10 strikers, top 10 goalkeeper and top 10 Mid Filder and their wages and their overall rating.

How wages of player will affect due to its weight and penalties and aggression.Sum of Wages of player will also depend on nationality and different club level leagues which player participates.

In jupyter notebook I analyzed how player age and height, and weight affect their overall rating and how many players are high paying and their positions can also affect their features like vision,accerlation,sprint,vision and long passes etc.

In SQL predict best playing squad of 30 players in which 3 goalkeeper,7 defender,10 mid fielder and 10 striker. Based on their various skilled factors like Penalites,LongShots,Finishing,Dribling,Ball Control,Long Passing,Short Passing etc.

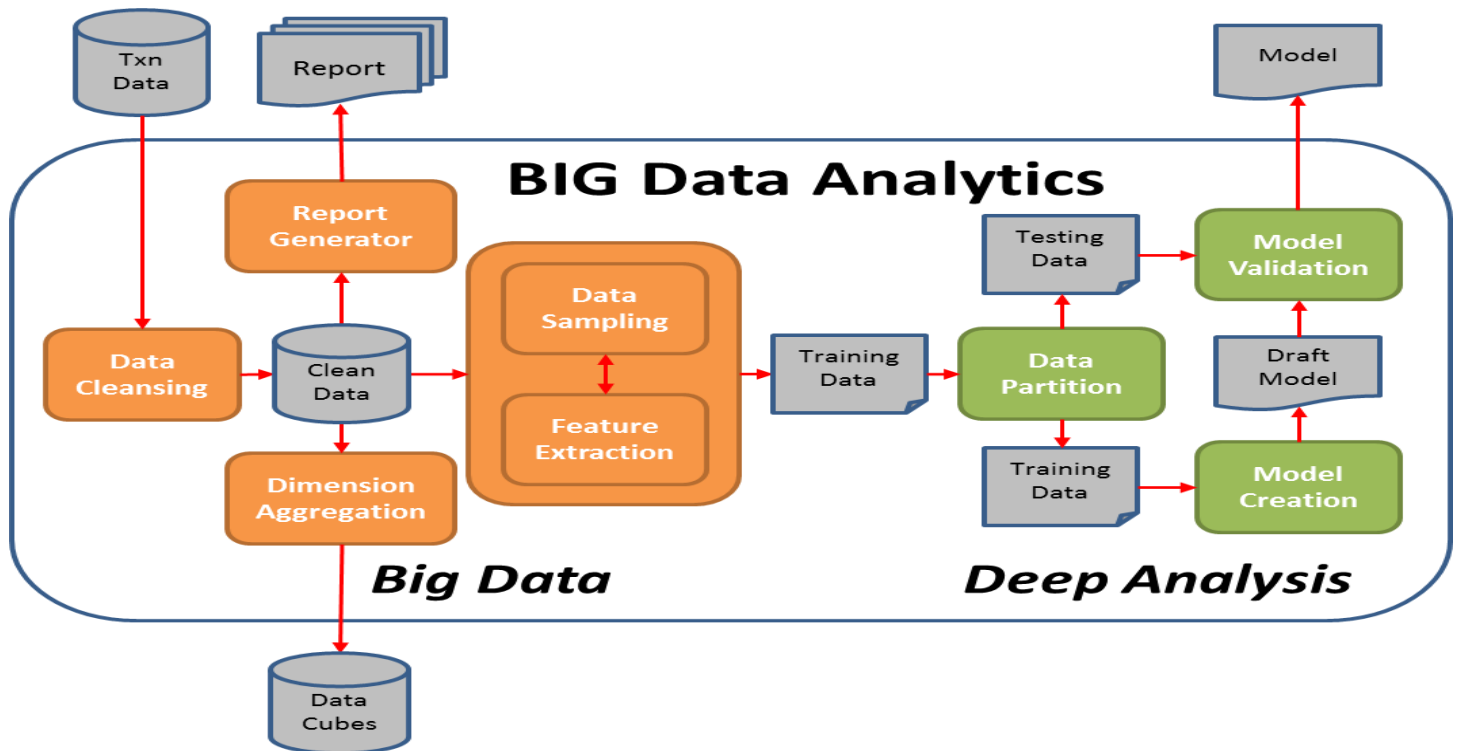
### 2.2 Tools Used:

Business intelligence tools like Tableau Public and libraries works such as NumPy ,Pandas, Matplotlib, Seaborn, requests, OS, Jupyter-Notebook, MS Excel, MySQL and Python are used to build the whole framework.

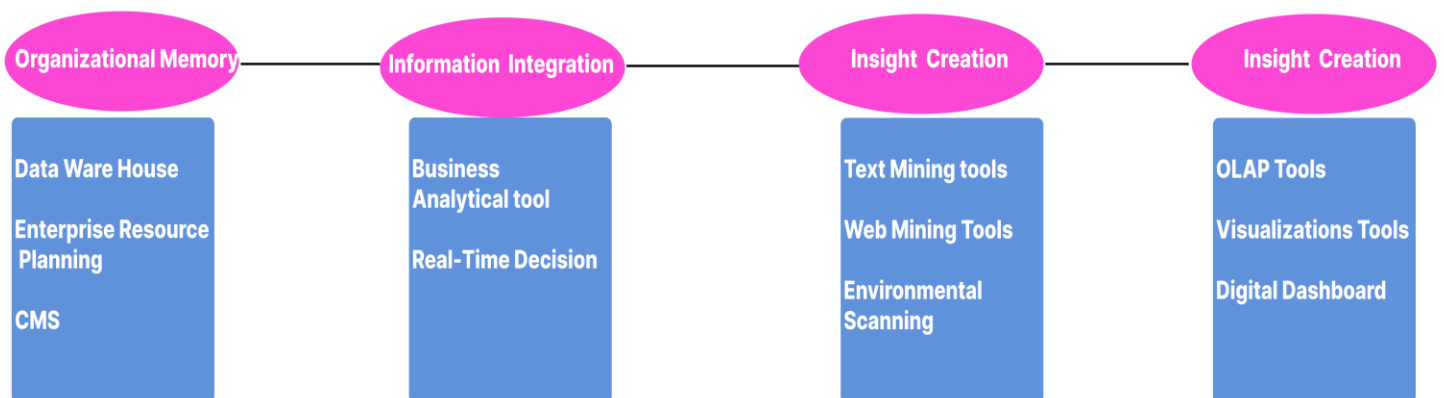


### 3. Design Details:

#### 3.1 Functional Architecture:



#### 3.2 How BI Tool Worked:



## 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.

## 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
- 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.



## 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 Key Performance Indicator:

- Wages
- Ages
- Weights
- Nationality
- Acceleration
- Strength
- Vision
- Sprint Speed
- Long shorts
- Finishing
- Overall Rating of Player
- Position Of Player
- Short Power
- Sliding Tackel
- Header Accuracy

## 5 Deployment:

