

# High Level Design (HLD)

## Analyzing Google Apps Store

Revision Number: 1.0  
Last date of revision: 22/07/2023

Preeti Jain

## Document Version Control

[illegible]

## Contents

Document	Version
Control.....	2
Abstract.....	
3	1 Introduction
.....	5
1.1 Why this High-Level Design Document?.....	5
1.2 Scope .....	5
2	General Description
.....	6 2.1 Product
Perspective & Problem Statement .....	6
Technology is the increasing need nowadays and used everywhere. One of the features	
Technology is android. Which we all use in our daily life. Android is a mobile operating	
system based on a modified version of the Linux kernel and other open source software,	
designed primarily for touchscreen mobile devices such as smartphones and tablets.	
.....	6
2.2 Tools used.....	6
3	Design
Details.....	7 3.1
Functional Architecture .....	7 3.2
Optimization .....	8 4
KPIs.....	9
4.1 KPIs (Key Performance Indicators) .....	9 5
Deployment.....	9

## Abstract

This project aims to analyze data from the Google Play Store to gain insights into app performance and user behavior. By extracting and analyzing metrics such as installs, active users, retention rates, revenue, and user reviews, the project aims to understand the popularity, engagement, and monetization potential of apps. The analysis will be conducted using Power BI, utilizing visualizations and interactive reports to present the findings. The project's objective is to provide app developers and stakeholders with actionable insights to enhance app performance, optimize user experience, and drive business growth in the competitive Google Play Store ecosystem.

## 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.

## 6 High Level Design (HLD)

## 2 General Description

### 2.1 Product Perspective & Problem Statement

Technology is the increasing need nowadays and used everywhere. One of the features Technology is android. Which we all use in our daily life. Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, designed primarily for touchscreen mobile devices such as smartphones and tablets.

Do ETL : Extract-Transform-Load the dataset and find for me some information from this large data. This is a form of data mining.

What all information can be achieved by mining this data, would be brainstormed by the interns

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

The objective of the project is to perform data visualization techniques to understand the insight of the data. This project aims to apply various Business Intelligence tools such as Tableau or Power BI to get a visual understanding of the data.

### 2.2 Tools used

Business Intelligence tools and libraries such as Numpy, Pandas, Power BI are used to build the whole framework.



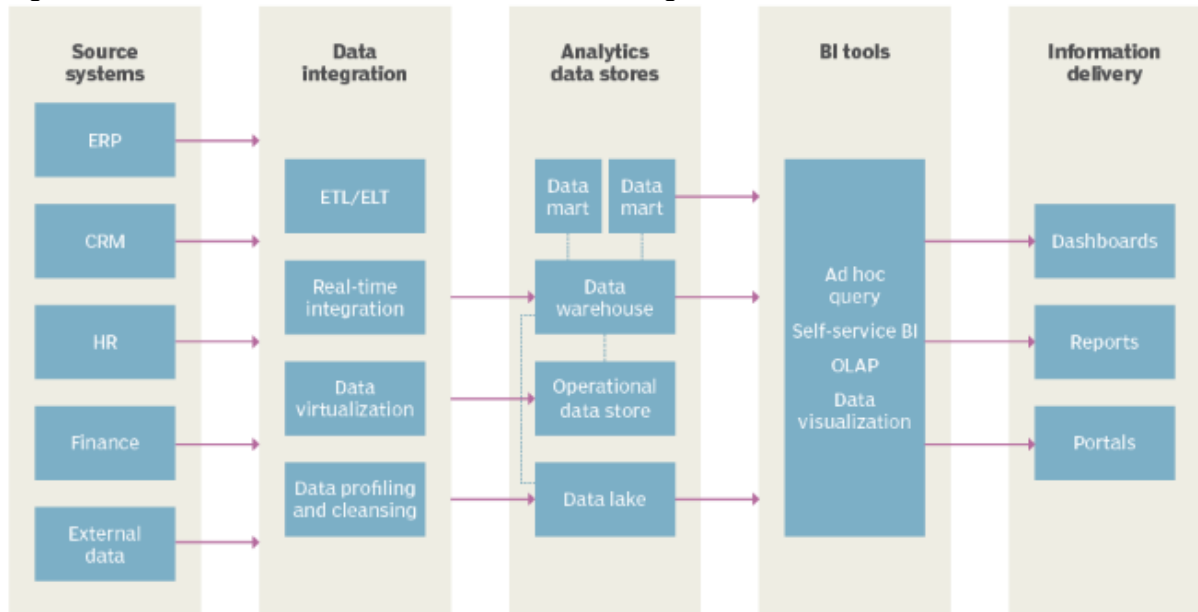


## 7 High Level Design (HLD)

### 3 Design Details

#### 3.1 Functional Architecture

Figure 1: Functional Architecture of Business Intelligence



### How BI Really Works



### 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.
- LODs - Look at the number of unique dimension members in the calculation.
    - 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.

## 9 High Level Design (HLD)

- 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 KPIs

When it comes to analyzing data from the Google Play Store, the High-Level Design (HLD) should focus on key performance indicators (KPIs) that provide insights into various aspects of app performance, user engagement, and revenue generation. Here are some KPIs you can consider for Google Play Store data analysis:

### 4.1 KPIs (Key Performance Indicators) -

Key indicators displaying a summary of the Housing Price and its relationship with different metrics

1. **Total Installs:** The number of times your app has been installed. This metric helps measure the overall reach and popularity of your app.
2. **Active Installs:** The number of currently active installations. It provides an idea of the size of your active user base.
3. **User Reviews and Ratings:** The average rating and the number of user reviews for your app. They provide feedback on user satisfaction and can indicate areas for improvement.
4. **Acquisition Channels:** The sources or channels through which users discover and install your app. It helps assess the effectiveness of marketing and user acquisition efforts.
5. **Conversion Rate:** The percentage of users who perform a desired action, such as making a purchase or subscribing to a service, after interacting with your app. It helps measure the effectiveness of your app in driving conversions.

## 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 PowerBi at scale, as well as organizing, orchestrating, and unifying disparate sources of data for business users and experts alike to author and consume content.

## 10 High Level Design (HLD)

1. **Define the data requirements:** Determine the specific data you want to analyze from the Google Play Store. This could include metrics like installs, ratings, reviews, revenue, etc.



2. Extract data from the Google Play Store: Identify the available data sources or APIs that provide access to the desired data. Google Play Developer API is a commonly used API for retrieving app data from the Play Store.
3. Transform and load the data: Extract the data from the Google Play Store API and transform it into a format suitable for analysis in Power BI. This may involve data cleaning, merging, and formatting as per your analysis requirements.
4. Create a data model: Design a data model in Power BI that represents the relationships between different data tables. This will allow you to perform analysis and create visualizations efficiently.
5. Develop visualizations and reports: Use Power BI's visualizations and reporting capabilities to create meaningful insights from the Google Play Store data. Choose the appropriate visualizations such as bar charts, line charts, tables, etc., to represent the KPIs and metrics you identified in the HLD.
6. Apply filters and slicers: Implement filters and slicers in your Power BI reports to enable interactive analysis. This allows users to dynamically explore the data based on various dimensions and metrics.
7. Implement data refresh: Set up a data refresh schedule in Power BI to ensure your reports are always up to date with the latest data from the Google Play Store. This may involve automating the data extraction process or manually refreshing the data.
8. Publish and share the reports: Once you have created the reports and dashboards, publish them to the Power BI service or Power BI Report Server. You can then share the reports with relevant stakeholders or embed them in other platforms for wider access.
9. Monitor and iterate: Continuously monitor the performance of your Power BI reports and gather feedback from users. Use this feedback to make improvements and iterate on your reports to meet evolving analysis requirements.