

High Level Design (HLD) Amazon Sales and Revenue Analysis

Revision Number: 1.0

Last date of revision: 12/02/2022

Rashmi Dubey



Document Version Control

Date Issued	Version	Description	Author
12 th Feb 2022	1.0	First Version of Complete HLD	Rashmi Dubey



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
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

Various types of Sales and Revenue related trends helps its managers in understanding the performance of their company. State-wise Sales and Revenue trend helps in identification of most important markets and markets which need attention to increase company's performance.

These trends are needed to take datadriven decisions to increase company's sales and revenue.



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
 - o Reliability
 - Maintainability
 - Portability
 - 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

Sales management has gained importance to meet increasing competition and the needfor improved methods of distribution to reduce cost and to increase profits.

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

dashboards to provide insights that can help in making data driven decisions to increase sales and revenue of the company.

2.2 Tools used

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













3 Design Details

3.1 Functional Architecture



Figure 1: Functional Architecture of Business Intelligence

Figure 1: Functional Architecture of Business Intelligence

ORGANIZATIONAL INFORMATION INSIGHT PRESENTATION INTEGRATION **MEMORY** CREATION -Data -Online Warehouse -Business Analytical -Text Mining **Analytical Tools** Tool Processing -Enterprise (OLAP) Tool resource -Web Mining planning (ERP) Tool -Visualization -Data Mining Tool -Knowledge -Environmental Repository -Digital Scanning -Real Time Dashboard -Content -RFID Decision Management -Score Card System (CMS)

How BI 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.
- <u>Use a continuous date filter</u>. 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.
- <u>Use Boolean or numeric filters</u>. Computers process integers and Booleans (t/f) much faster than strings.
- Use <u>parameters</u> and <u>action filters</u>. 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.
 - 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 include filters, calculated groups load only named members of the domain, whereas Tableau's group function loads the entire domain.
- <u>Use Booleans or numeric calculations instead of string calculations</u>. 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 Sales and Revenue Trends.



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 Performance Indicators displaying a summary of Sales and Revenue and their relationship with different metrics.

For Sales:

- Sales by State
- Sales % by City
- Category-wise Product's Sold (in %)
- Monthly trend of Sales

For Revenue:

- Revenue generation per State
- Distribution of Revenue (in %) by Cities
- Product's Revenue(in %) in various Product Categories
- Month-wise Revenue trend



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

Power BI prioritizes choice in flexibility to fit, rather than dictate, your enterprise architecture. Power BI Desktop and Power BI Service leverage your existing technology investments and integrate them into your IT infrastructure to provide a self-service, modernanalytics platform for your users. With on-premises, cloud, and hosted options, there is a version of Power BI to match your requirements.





