

High Level Design

ANALYZING AMAZON SALES DATA

Revision number: 1.2

Last date of revision: 17/10/2022

Aditya Sahoo

Document Version Control

Date	Version	Description	Author
03/10/2022	1.0	Abstract, Introduction, General Description	Aditya Sahoo
12/10/2022	1.1	Design Detail, KPI, Deployment	Aditya Sahoo
17/10/2022	1.2	Final Revision	Aditya Sahoo

Contents

Abstract.....	4
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. KPI	9
4.1 KPIs (Key Performance Indicators).....	10
5. Deployment.....	10

Abstract

Amazon Sales data refers to sales, high performing sellers and several other data points. There are millions of Amazon sellers around the world. Nearly half of them are self-employed and live off their ecommerce/retail businesses (47%), and 22% earn income from their Amazon businesses alone. Amazon sales data Analysis focuses on the process of analysing consumer behaviour, sales, and several other attributes in order to make improved, data-driven decisions. It is key to successfully sustaining their businesses and earning profits and for this purpose, they analyse different metrics like sales, Sales Quantity, Discount rate, Sales over years etc.

By analysing different metrics, you will be able to increase and improve your performance in terms of sales, Items to be sold and discount rates etc. Analysis of the sales data the main factor that contributes to sellers improving their business and increasing their revenue. They can better understand the market trends and customers' buying behaviours and help them cater to what the customers really want. In the world of rising new technology and innovation, E-commerce industry is advancing with the role of Data Science and Analytics.

Data analysis can help them to understand their business in a quiet different manner and helps to improve the quality of the service by identifying the weak areas of the business. This study demonstrates the how different analysis help to make better business decisions and help analyse customer trends and satisfaction, which can lead to new and better products and services. Different analysis performed to get the key insights from this data based on which business decisions will be taken.

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 before 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

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

This Project aims to Analyze Amazon Sales from 2017 to 2019 to know more in-depth information about the items that are in high demand, items that are generating high profits which items should not be sold and how much stock we have to maintain for further Sales etc.

2.2 Tools used

Excel, Power BI, Jupyter Notebook and Python Programming Language.

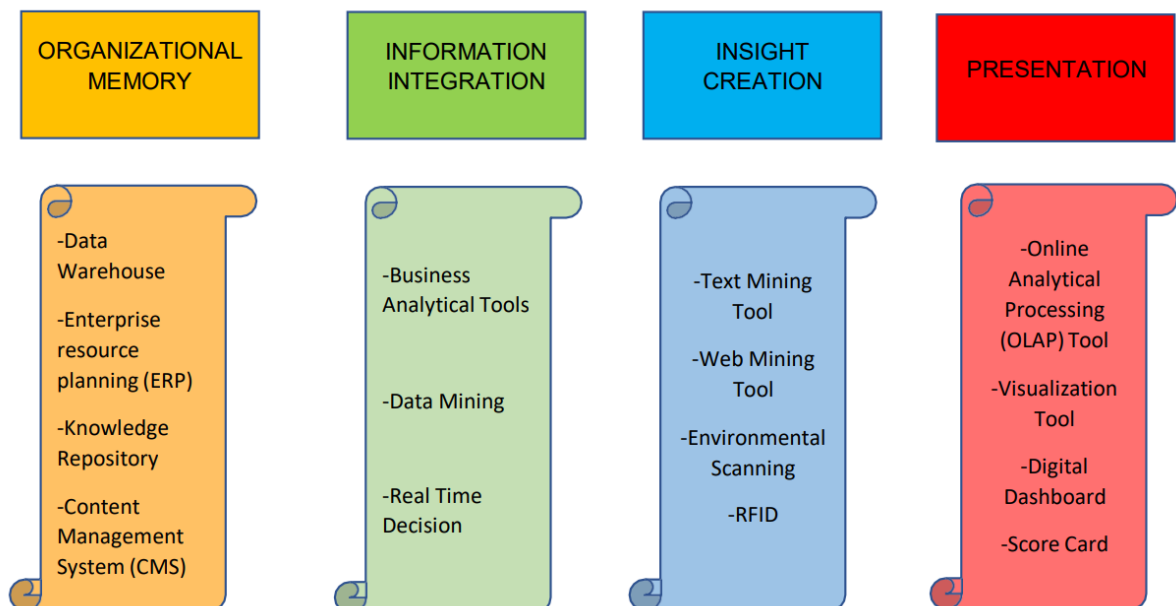


3. Design Details

3.1 Functional Architecture



How BI Works



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-of-date 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).

4. 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.
- 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. KPI

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



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 indicators displaying a summary of Sales Data and its relationships with different metrics.

- Yearly, Quarterly, Monthly Ups and Downs in Sales & Profits.
- Items That Generated Highest Sales, Profit etc.
- Top 5 Items that generated highest Sales and Top 5 Items by Quantity.

5. Deployment

Prioritizing data and analytics couldn't come at a better time. 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, modern analytics platform for your users. With on-premises, cloud, and hosted options, there is a version of Power BI to match your requirements

