High-Level Design Document (HLD)

ANALYZING AMAZON SALES DATA

Revision Number: 1.2
Last Date of Revision: 21/09/2023

MD TAHSEEN EQUBAL

• Document Version Control

Date Issued	Version	Description	Author
18/10/2023	1.0	Abstract, Introduction, General Description and Deployment	MD TAHSEEN EQUBAL
21/10/23	1.1	Final Revision	MD TAHSEEN EQUBAL

Contents

ABSTRACT

In today's competitive E-commerce industry, organizations strive to gain a competitive edge by efficiently collecting and utilizing customer information and purchasing trends. This high-level design document critically evaluates how Amazon, a service-based organization, leverages Management Information Systems to achieve a competitive advantage through efficient information management and acquisition.

Effective sales management is crucial in today's market for predicting business performance and future growth. Companies with robust sales management processes have demonstrated superior growth by focusing on key products, customer retention, and minimizing losses. With increasing competition, proper sales management is essential for businesses to thrive and compete effectively.

1.INTRODUCTION

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) document is to provide detailed insights into the project and serve as a reference manual for coding. This document aims to detect potential contradictions before coding and describe the high-level architecture of the project. The HLD will:

- Present design aspects in detail.
- Describe the user interface.
- Explain hardware and software interfaces.
- Specify performance requirements.
- Define design features and the project's architecture.
- Enumerate non-functional attributes such as security, reliability, maintainability, portability, reusability, application compatibility, resource utilization, and serviceability.

1.2 Scope

The HLD documentation outlines the system's structure, including database architecture, application architecture, application flow, and technology architecture. The document is designed to be accessible to both technical and non-technical administrators of the system.

2.General Description

2.1. Product Perspective & Problem Statement

This project, "Analyzing Amazon Sales Data," aims to provide in-depth insights into Amazon's sales data from 2017 to 2019. The project seeks to identify high-demand items, profitable products, items to discontinue, and optimal stock levels for future sales.

2.2 Tools Used

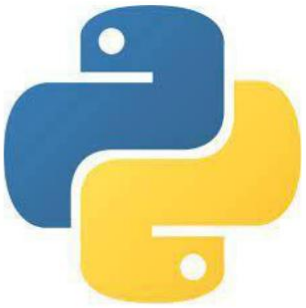
The project utilizes various tools and libraries, including:

- Business Intelligence tools
- MS-Excel
- Tableau
- Python Programming Language

These tools are integral to building the project's framework.

EXCEL

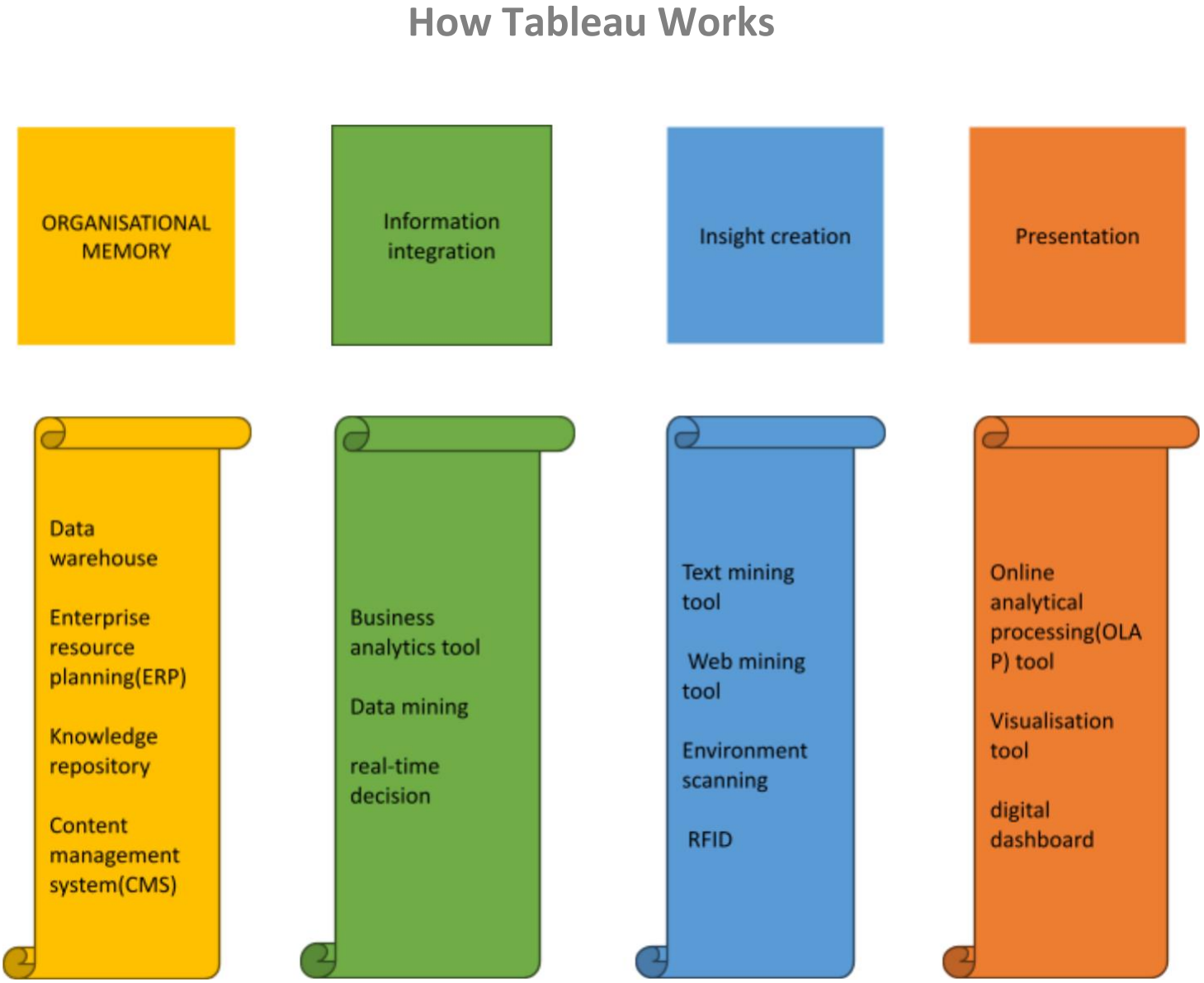
PYTHON



3.Design Details

3.1 Functional Architecture

Figure 1 : Getting Data from score , using Excel to clean and format data and then using Tableau to create dashboards for decision making



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

1. 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.

1. 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).

1. 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.

4.KPI

Dashboards will be implemented to display and indicate certain KPIs

Key Performance Indicators (KPIs) will be implemented to monitor and display various metrics related to sales data, including yearly, quarterly, and monthly trends, top-selling items, and forecasting.

4.1 KPIs (Key Performance Indicators)

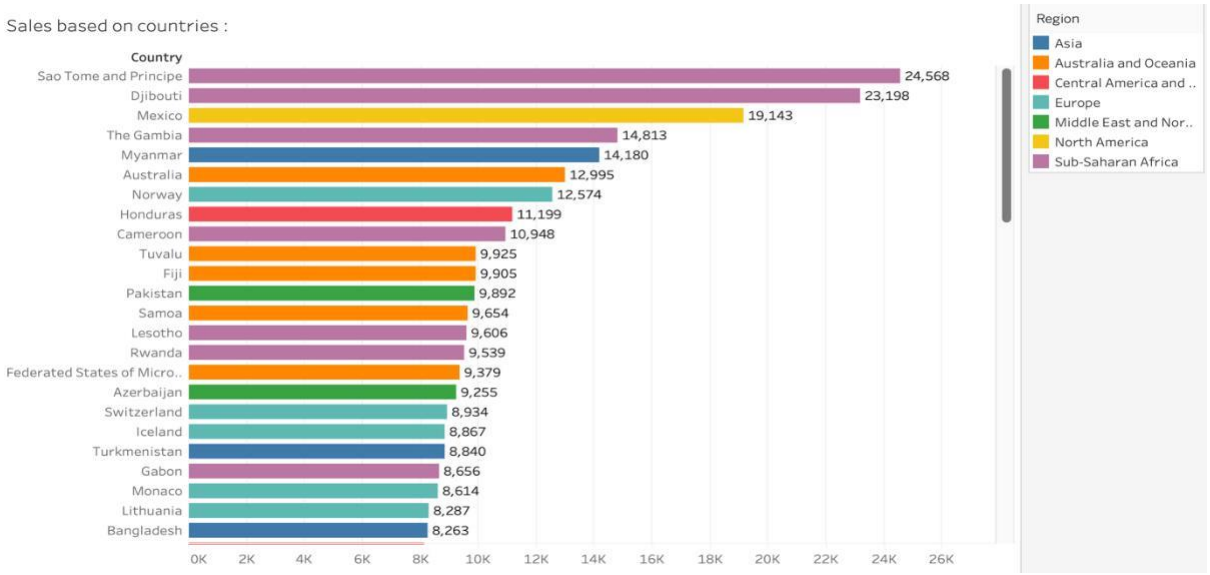
Key indicators displaying a summary of Sales
Data and its relationships with different metrics.

1. Yearly, Quarterly, Monthly Ups and Downs in Sales & Profits.
2. Items That Generated Highest Sales, Profit etc.
3. Top 5 Items that generated highest Sales and Top 5 Items by Quantity.
4. Bottom 5 Items that generated Lowest Sales and Bottom 5 Items by Quantity.
5. Forecasting.

5.Deployment

The project emphasizes the importance of data and analytics in business decision-making. It leverages Power BI for scalable, self-service analytics, integrating existing technology investments into the IT infrastructure. The deployment approach prioritizes flexibility to fit various enterprise architectures, offering on-premises, cloud, and hosted options.

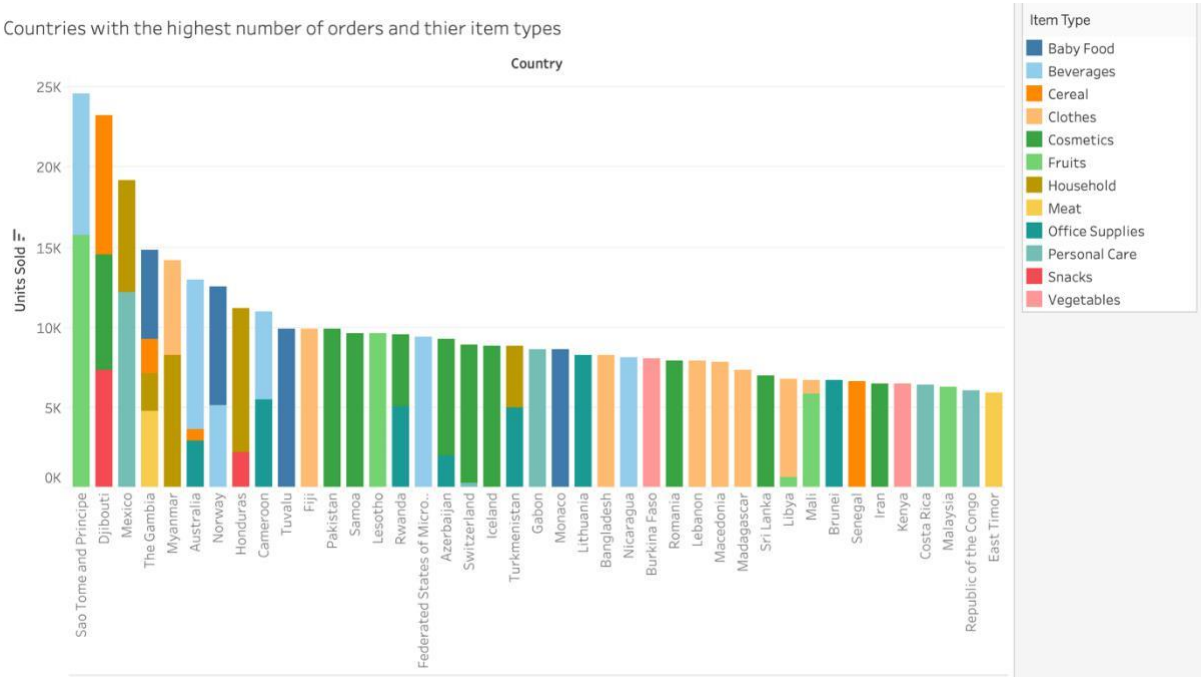
This High-Level Design Document outlines the project's objectives, scope, and technical details. It provides a foundation for coding, ensuring that the project's architecture and requirements are well-defined. As the project progresses, this document will serve as a valuable reference for development and evaluation.



Sales Channel

Total Profit and Total cost

Countries with the highest number of orders and thier item types



This dashboard provides actionable insights for improving distribution methods. It helps you identify profitable regions, optimize order modes, assess cost efficiency, and make data-driven decisions based on historical trends. By using these insights, you can reduce distribution costs and increase profits, ultimately enhancing your overall business strategy with a primary focus on optimizing distribution methods, reducing operational costs, and boosting overall profitability. This project involved leveraging data analytics tools and techniques, such as Excel, Python, and Tableau, to gain valuable insights into the company's sales operations.