Project Documentation: Dashboard for Sales, Inventory, and Amazon Ads

Project Overview

The project aims to build an interactive dashboard to track and analyze **sales performance**, **inventory levels**, and **Amazon ad campaigns**. This dashboard will provide real-time insights to optimize decision-making and business strategies.

Objectives

1. **Sales**: Provide visibility into revenue, units sold, and performance trends.
2. **Inventory**: Monitor stock levels, prevent stockouts, and optimize turnover.
3. **Amazon Ads**: Track ad performance, ROI, and sales impact from campaigns.

Scope

- Integrate data from multiple sources, including ERP systems, Amazon Advertising, and inventory management tools.
- Design a unified dashboard for stakeholders with customizable filters.
- Automate data refresh and ensure real-time insights.

Key Features

1. **Sales Module**:
    - Total sales revenue and units sold.
    - Sales trends over time.
    - Breakdown by product, category, and region.
2. **Inventory Module**:
    - Current stock levels.
    - Reorder alerts and safety stock tracking.
    - Inventory turnover ratio and days of stock remaining.
3. **Amazon Ads Module**:
    - Ad spend, clicks, and impressions.
    - ROI/ACoS (Advertising Cost of Sales).
    - Campaign-wise performance analysis.

Data Requirements

1. **Sales Data**:
    - Source: ERP, CRM, or e-commerce platforms.
    - Fields: Order date, product, category, region, revenue, units sold.
2. **Inventory Data**:
    - Source: Inventory or warehouse management systems.
    - Fields: Product ID, stock level, reorder point, safety stock, turnover rate.
3. **Amazon Ads Data**:
    - Source: Amazon Ads API or reports.
    - Fields: Campaign name, impressions, clicks, ad spend, attributed sales.

Technical Architecture

1. Data Sources:

- ERP systems or e-commerce platforms for sales data.
- Inventory management tools for stock levels.
- Amazon Ads API for ad performance data.

2. ETL Process:

- Extract: Pull data from APIs, databases, or CSV files.
- Transform: Clean and structure data using ETL tools like Power Query or Python scripts.
- Load: Store data in a central database or BI tool.

3. Visualization:

- Tool: Power BI / Tableau / Google Data Studio.
- Components:
    - Sales KPIs: Revenue, units sold, trends.
    - Inventory charts: Stock levels, reorder alerts.
    - Ads performance: ROI, ACoS, and click-through rates.

Dashboard Layout

1. Header

- Filters: Date range, product category, region, and campaign.

- Summary KPIs: Total sales, inventory health, and ad ROI.

2. Sections

Sales

- Line chart: Sales trend over time.
- Bar chart: Sales by product or category.
- Table: Detailed sales breakdown.

Inventory

- Gauge: Current stock vs. reorder point.
- Table: Inventory levels with reorder alerts.
- Bar chart: Stock turnover ratio.

Amazon Ads

- Pie chart: Spend distribution by campaign.
- Bar chart: Ad spend vs. attributed sales.
- Table: Campaign-wise performance.

Key Metrics

1. **Sales Metrics**:
    - Total revenue.
    - Units sold.
    - Growth percentage (MoM, YoY).
2. **Inventory Metrics**:
    - Days of stock remaining.
    - Reorder alerts.
    - Inventory turnover ratio.
3. **Amazon Ads Metrics**:
    - ROI (Return on Investment).
    - ACoS (Advertising Cost of Sales).
    - CTR (Click-through Rate).

Automation and Refresh

- **Frequency**: Data auto-refresh every 1 hour.

- **Method**: Use APIs or scheduled data imports.

Stakeholders

- **End Users**: Sales managers, inventory planners, marketing teams.
- **Technical Team**: Developers for integration and visualization.

Timeline

Phase	Timeline
Requirement Analysis	Week 1
Data Integration Setup	Week 2–3
Dashboard Design	Week 4
Testing and Feedback	Week 5
Deployment	Week 6

Deliverables

- Functional dashboard with interactive filters.
- Automated data pipeline for real-time updates.
- Documentation for maintenance and further enhancements.

Risks and Mitigations

Risk	Mitigation
Data inconsistencies	Implement robust data validation scripts.
API rate limits for Amazon Ads	Use caching or batching for data requests.
Performance with large datasets	Optimize queries and use aggregate data.

Next Steps

- Finalize data sources and access.
- Develop ETL pipelines for integration.
- Build and test dashboard visuals.