



Low Level Design Document

Introduction

This Low Level Design (LLD) document outlines the implementation details for **Trendly - E-commerce Sales Trends Visualizer**. The project aims to deliver a Power BI dashboard for an e-commerce company, visualizing sales trends, top products, and customer segments using imported SQL data and DAX calculations.

1. System Components

Component	Description	Key Responsibilities
SQL Data Source	E-commerce transactional database	Store and provide sales, product, customer data
Data Import Layer	Power BI data import (SQL connector)	Extract and load data into Power BI
Data Model Layer	Power BI data model (tables, relationships)	Structure data for analysis
DAX Calculation Layer	DAX measures and calculated columns	Compute KPIs, trends, and segments
Visualization Layer	Power BI dashboard (reports, visuals)	Display trends, top products, segments

2. Class/Interface Overview

Entity/Class	Description	Key Methods/Attributes
DataImporter	Handles SQL data import	<code>connect()</code> , <code>import_tables()</code>
SalesTable	Sales transaction data model	<code>OrderID</code> , <code>ProductID</code> , <code>CustomerID</code> , <code>OrderDate</code> , <code>Amount</code>
ProductTable	Product data model	<code>ProductID</code> , <code>ProductName</code> , <code>Category</code>
CustomerTable	Customer data model	<code>CustomerID</code> , <code>Segment</code> , <code>Region</code>
DAXMeasures	DAX calculations interface	<code>TotalSales</code> , <code>TopProducts</code> , <code>SalesTrend</code> , <code>CustomerSegments</code>

Relationships:

- `SalesTable` references `ProductTable` and `CustomerTable` via foreign keys.
- `DAXMeasures` operates on data models.

3. Data Structure Overview



Table/Model	Fields/Attributes
Sales	OrderID, ProductID, CustomerID, OrderDate, Amount
Product	ProductID, ProductName, Category
Customer	CustomerID, Segment, Region

4. Algorithms/Logic

Data Import Flow:

```
connect_to_sql()
import Sales, Product, Customer tables
define relationships:
    Sales.ProductID -> Product.ProductID
    Sales.CustomerID -> Customer.CustomerID
```

DAX Calculations:

```
TotalSales = SUM(Sales[Amount])
TopProducts = TOPN(10, SUMMARIZE(Product, [ProductName], "Sales", SUM(Sales[Amount])))
SalesTrend = CALCULATE([TotalSales], GROUPBY(Sales[OrderDate]))
CustomerSegments = SUMMARIZE(Customer, [Segment], "Sales", SUM(Sales[Amount]))
```

Visualization Flow:

- Bind DAX measures to visuals (line chart for trends, bar chart for top products, pie chart for segments).

5. Error Handling

Scenario	Handling Approach
SQL connection failure	Show error message, retry option
Data import schema mismatch	Log error, notify user
DAX calculation error	Display fallback value, log details
Missing/empty data	Show "No Data" message in visuals

End of Document