



Low Level Design Document

Introduction

This Low Level Design (LLD) document outlines the implementation details for **Retail360 - Store Performance Visualizer**. The project aims to deliver a Tableau dashboard for comparing sales, foot traffic, and conversion rates across multiple retail stores, featuring heatmaps and trend lines for actionable insights.

1. System Components

Component	Description	Key Responsibilities
Data Source	Raw data storage (CSV/DB/Tableau Extract)	Store sales, foot traffic, conversion data
Data Preparation	ETL/cleaning scripts or Tableau Prep	Clean, aggregate, and format data
Tableau Dashboard	Visualization layer	Display heatmaps, trend lines, KPIs
User Interface	Tableau dashboard interface	User interaction, filtering, drill-down

2. Class/Interface Overview

Class/Interface	Description	Key Methods/Attributes
DataModel	Represents store performance data	<code>store_id</code> , <code>date</code> , <code>sales</code> , <code>foot_traffic</code> , <code>conversion_rate</code>
DataLoader	Loads and preprocesses data	<code>load_data()</code> , <code>clean_data()</code> , <code>aggregate_data()</code>
DashboardController	Manages dashboard logic	<code>update_filters()</code> , <code>refresh_views()</code>

Relationships:

- `DataLoader` populates `DataModel`
- `DashboardController` interacts with Tableau visualizations using processed data

3. Data Structure Overview

Field	Type	Description
<code>store_id</code>	String	Unique store identifier
<code>date</code>	Date	Date of record
<code>sales</code>	Float	Total sales amount
<code>foot_traffic</code>	Integer	Number of visitors



conversion_rate	Float	Sales/Foot Traffic ratio
-----------------	-------	--------------------------

4. Algorithms/Logic

Data Preparation Flow:

```
For each record in raw data:
  Validate required fields (store_id, date, sales, foot_traffic)
  Calculate conversion_rate = sales / foot_traffic if foot_traffic > 0
  Aggregate data by store and date
Export cleaned data to Tableau Data Source
```

Dashboard Logic:

- Filter data by date/store as per user input
- Generate heatmaps (store vs. metric)
- Plot trend lines (metric over time)
- Display KPIs (totals, averages)

5. Error Handling

Scenario	Handling Approach
Missing/invalid data fields	Skip record, log error
Division by zero (conversion)	Set conversion_rate to 0, log warning
Data source connection failure	Show error message in dashboard
Tableau extract refresh failure	Notify admin, retry on next schedule

End of Document