



Low Level Design Document

Introduction

This Low Level Design (LLD) document outlines the implementation details for **TableauTrend - Tableau Sales Trend Analyzer**. The project aims to deliver a Tableau dashboard for analyzing and visualizing sales trends over time using line and area charts.

1. System Components

Component	Description	Key Responsibilities
Data Source	Sales data input (CSV/DB/Tableau Extract)	Provide raw sales data
Data Preparation	Data cleaning & transformation (Tableau Prep or in-dashboard)	Prepare data for analysis
Tableau Dashboard	Visualization layer	Display sales trends, filters, charts
User Interface	Dashboard controls (filters, date pickers)	User interaction and filtering

2. Class/Interface Overview

Class/Interface	Description	Key Methods/Attributes
DataConnector	Connects to data source	<code>connect()</code> , <code>fetch_data()</code>
DataModel	Represents sales data	<code>date</code> , <code>sales_amount</code> , <code>region</code> , <code>product</code>
TrendChart	Renders line/area charts	<code>render_line_chart()</code> , <code>render_area_chart()</code>
FilterControl	UI filter logic	<code>apply_filter()</code> , <code>reset_filter()</code>

Relationships:

- `DataConnector` feeds data to `DataModel`
 - `TrendChart` consumes `DataModel`
 - `FilterControl` interacts with `TrendChart` and `DataModel`
-

3. Data Structure Overview

Field	Type	Description
<code>date</code>	Date	Date of sale
<code>sales_amount</code>	Float	Sales value
<code>region</code>	String	Sales region



product	String	Product name/category
---------	--------	-----------------------

4. Algorithms/Logic

Data Preparation Flow:

```
connect_to_data_source()
data = fetch_data()
cleaned_data = remove_nulls_and_duplicates(data)
grouped_data = aggregate_by_date_and_dimension(cleaned_data)
```

Visualization Logic:

```
on_filter_change(selected_filters):
    filtered_data = apply_filters(grouped_data, selected_filters)
    render_line_chart(filtered_data)
    render_area_chart(filtered_data)
```

5. Error Handling

Scenario	Handling Approach
Data source unavailable	Show error message, retry option
Invalid/missing data fields	Skip records, log warning
Chart rendering failure	Display fallback message
Filter returns no data	Show "No data available" message

End of Document