Low Level Design

Amazon Sales Data Analysis

|  |  |
| --- | --- |
| **Written By** | Team |
| **Document Version** | 0.1 |
| **Last Revised Date** |  |

**DOCUMENT CONTROL**

## Change Record:

|  |  |  |  |
| --- | --- | --- | --- |
| **VERSION** | **DATE** | **AUTHOR** | **COMMENTS** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Reviews:**

|  |  |  |  |
| --- | --- | --- | --- |
| **VERSION** | **DATE** | **REVIEWER** | **COMMENTS** |
|  |  |  |  |

**Approval Status:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **VERSION** | **REVIEW**  **DATE** | **REVIEWED BY** |  | **APPROVED BY** | **COMMENTS** |
|  |  |  |  |  |  |

# Contents

#### Introduction 04

* 1. **What is Low-Level Design Document? 04**

|  |  |  |
| --- | --- | --- |
| **1.2** | | **Scope 04** |
| **2.** | **Architecture 05** | |
| **3.** | **Architecture Description 08** | |
|  | **3.1** | **Data Description 08** |
|  | **3.2** | **Web Scrapping 08** |
|  | **3.3** | **Data Transformation 08** |
|  | **3.4** | **Data insertion into database 08** |
|  | **3.5** | **Connection with SQL server 08** |
|  | **3.5** | **Export Data from database 12** |
|  | **3.6** | **Deployment 12** |
| **4.** | **Unit** | **test cases 15** |

# Introduction

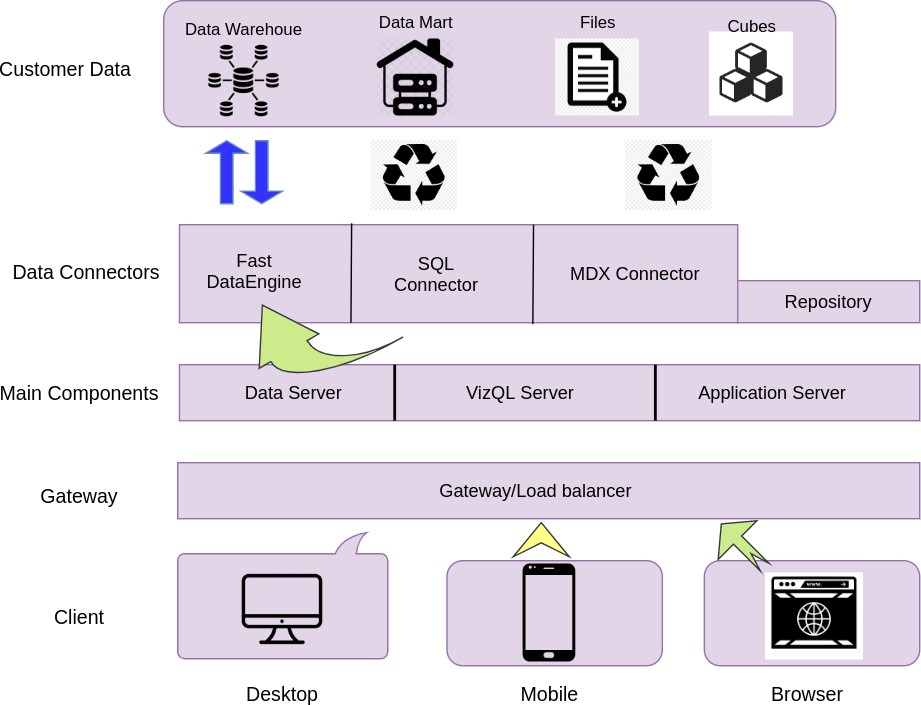
## What is Low-Level design document?

The goal of the LDD or Low-level design document (LLDD) is to give the internal logic design of the actual program code for the Amazon Sales Data Analysis. LDD describes the class diagrams with the methods and relations between classes and programs specs. It describes the modules so that the programmer can directly code the program from the document.

## Scope

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. The process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

# Architecture



**Tableau Server Architecture**

Tableau has a highly scalable, n-tier client-server architecture that serves mobile clients, web clients and desktop-installed software. Tableau Server architecture supports fast and flexible deployments.

The following diagram shows Tableau Server’s architecture:

**LOW LEVEL DESIGN**

**6**

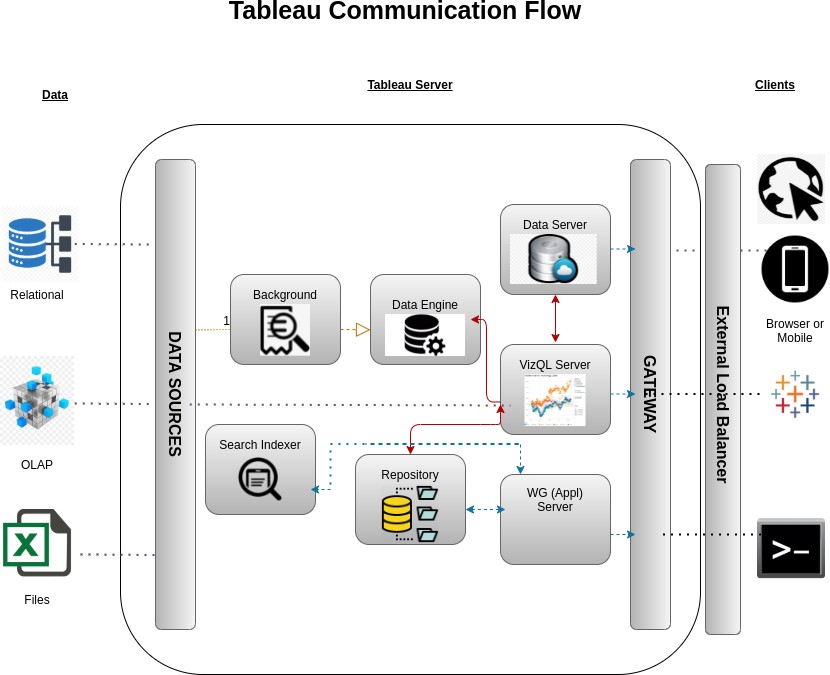


Tableau Server is internally managed by the multiple server processes.

**1. Gateway/Load Balancer**

It acts as an Entry gate to the Tableu Server and also balances the load to the Server if multiple Processes are configured.

**2) Application Server:-**

Application Server processes (wgserver.exe) handle browsing and permissions for the Tableau Server web and mobile interfaces. When a user opens a view in a client device, that user starts a session on Tableau Server. This means that an Application Server thread starts and checks the permissions for that user and that view.

1. **Repository:-**

Tableau Server Repository is a PostgreSQL database that stores server data. This data includes information about Tableau Server users, groups and group assignments, permissions, projects, data sources, and extract metadata and refresh information.

1. **VIZQL Server:-**

Once a view is opened, the client sends a request to the VizQL process (vizqlserver.exe). The VizQL process then sends queries directly to the data source, returning a result set that is rendered as images and presented to the user. Each VizQL Server has its own cache that can be shared across multiple users

1. **Data Engine:-**

It Stores data extracts and answers queries.

1. **Backgrounder:-**

The backgrounder Executes server tasks which includes refreshes scheduled extracts, tasks initiated from tabcmd and manages other background tasks.

1. **Data Server:-**

Data Server Manages connections to Tableau Server data sources

It also maintains metadata from Tableau Desktop, such as calculations, definitions, and groups.

# Architecture Description

## Data Description

The Dataset contains total profit generated in different category of items it different region of the world with respect to some attributes.

* + 1. Region: Different continental region of the world.
    2. Country: It is the list of different countries with respect to the different region.
    3. Item Type: This is the list of different categories of product available in Amazon.
    4. Sales Channel: It is a binary category feature which represents the mode of channel (Offline or Online).
    5. Order priority: This represents the priority of the order placed.
    6. Order date: Date of order placed by the customer.
    7. Order ID: This is a unique ID generated for each order placed by the customer.
    8. Ship Date: Date of order dispatched for the delivery.
    9. Units Sold: Number of items sold with respect to that region.
    10. Unit price: Prices per each unit.
    11. Cost price: Actual price per each unit.
    12. Total Revenue: The total revenue generated for each item category.
    13. Total Cost: The total cost generated for each item category.
    14. Total Profit: The total profit generated for each item category.

## Web Scrapping

Web scraping is a technique to automatically extract content and data from websites using bots. It is also known as web data extraction or web harvesting. Web scrapping is made simple now days, many tools are used for web scrapping. Some of python libraries used for web scrapping are Beautiful Soup, Scrapy, Selenium, etc.

## Data Transformation

In the Transformation Process, we will convert our original datasets with other necessary attributes format. And will merge it with the Scrapped dataset.

## Data Insertion into Database

1. Database Creation and connection - Create a database with name passed. If the database is already created, open the connection to the database.
2. Table creation in the database.
3. Insertion of files in the table

## Make the SQL connection and set up the data source

**Step 1: Configuring Tableau**

Launch Tableau on your workstation and select SQL Server from the connect column on the left. This will open a dialogue box where you need to provide the connection details for SQL Server.

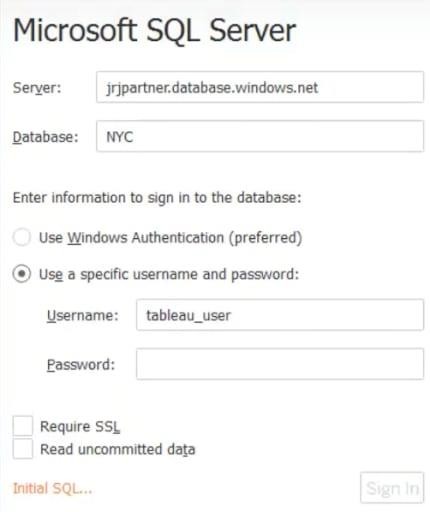
To connect with tableau, you will need to provide information about the server which hosts your database. If you want to connect to a contained database, you can also specify the name of the database.

To connect with a port other than the default port, you need to specify the port and server as follows:

<server\_name><port\_number>

Example query: my\_server 8051

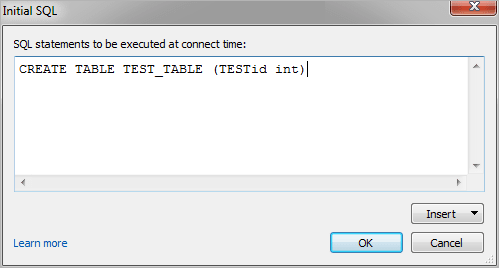
There are two ways in which you can sign-in to the server, either by using Windows authentication or by using the username and password. Using the username and password becomes a must if you’re working with a password-protected server in a non-Kerberos environment.



Click on Sign in to establish a connection. This will enable a connection without SSL. To establish an SSL enabled connection, click the Require SSL checkbox before you sign in.

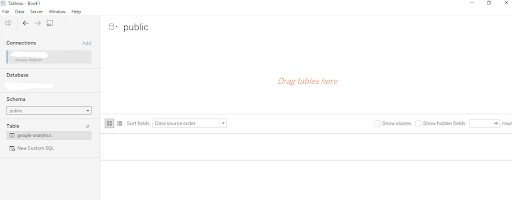
SQL Server provides an option to let the user queries access the modified rows even before they have been committed. This option is called Read Uncommitted data. It saves time by preventing complex queries such as extract refreshes from locking the database and causing a delay. If this option is unchecked, Tableau makes use of default isolation levels.

If you want to run a specific SQL command every-time a new connection is established, you can use the Initial SQL option. This will open a dialogue box, where you can specify your desired SQL query.



#### Step 2: Configuring Data Source

The data source page loads up after configuring the Tableau connector and successfully signing in. This is how the page looks like:



Select the data source name option and give a unique name to the database you are using. It’s considered a good practice to have a unique name as it makes it much easier for users to identify the database from which data is being fetched.

To select the desired schema, you can use the schema drop-down list from the column on the left. You can also perform a text-based search to find the desired option. Now similarly find and select the desired table and drag it onto the canvas.

This is how you can connect SQL Server with Tableau. Now click on the sheets tab to begin the analysis.

We loaded our amazon sales data soon after the configuring the SQL with tableau . We performed some quarries with our data to get some insights. And created some tables by grouping the country and items in order to find the percentage profit in each item category.

## Export Data from Database

Data Export from Database - The data in a stored database is exported as a CSV file to be used for Data Pre-processing.

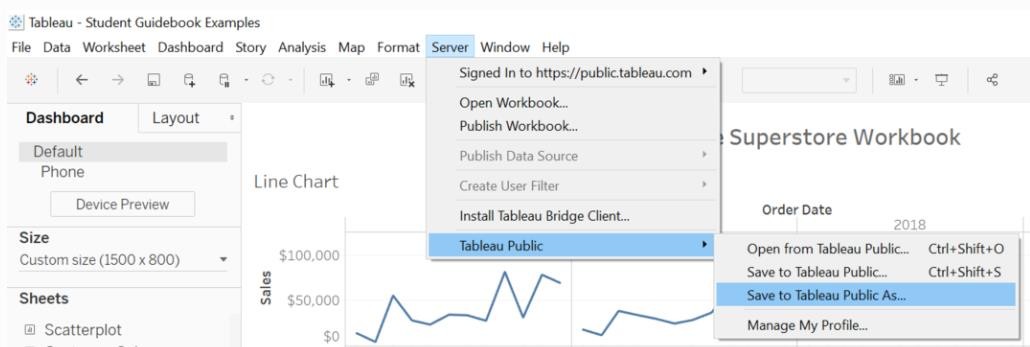
## Deployment.

Once you’ve completed your dashboard, follow these steps:**- Server, Tableau Public, Save to Tableau Public As**

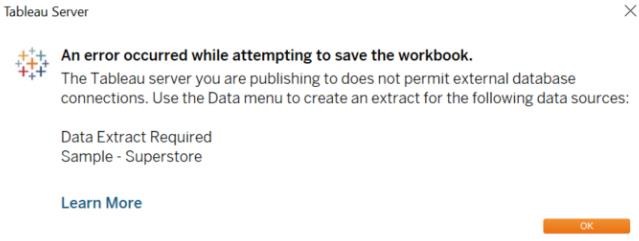
You may be prompted to log into your Tableau Public profile first if this is your first time publishing.

### LOW LEVEL DESIGN

**12**



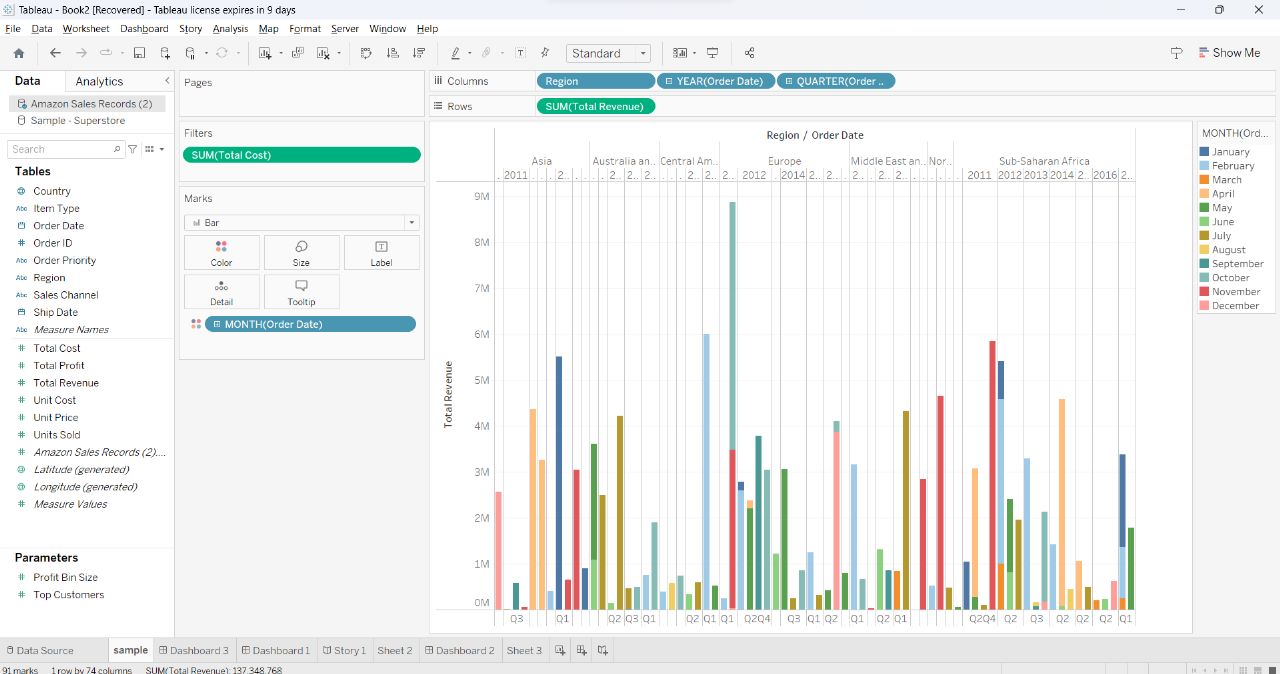
Next, fill out the title you want your viz to have and click “save”.



This message means that your connection to the Sample-Superstore data set is a live connection. Tableau Public cannot host live connections, so you’ll need to convert your connection to an extract (like a frozen screenshot of your data).

Here in the below screenshot, we can see that out workbook has been published to tableau public.

 **13 LOW LEVEL DESIGN**

****

# Unit Test Cases

|  |  |
| --- | --- |
| **TEST CASE DESCRIPTION** | **EXPECTED RESULTS** |
| Seasonal Discounts. | Providing discounts to specific products in desired country during the right seasons which increases the sale rate there by increases the profit. |
| Discounts for product with maximum profit. | Finding the products which has maximum profits in different countries and providing discounts which increases the profit. |
| Finding regional sales trend. | By analyzing the sales trend in the different country by grouping the products helps us to know which products are not meeting the expectation which helps us to change the plans accordingly. |
| Increasing the specific items during the right seasons. | By doing so we can increase the sales rate of that products in the expected season. |
| Inflation | Purchasing the Items when there is a low in Demand and sell it in the Required Area |