Low Level Design

# Analyzing Amazon Sales Data

|  |  |
| --- | --- |
| **Written By** | Chanchal Gopalakrishnan,  Rahul Bhanushali |
| **Document Version** | 0.1 |
| **Last Revised Date** |  |

**DOCUMENT CONTROL**

## Change Record:

|  |  |  |  |
| --- | --- | --- | --- |
| **VERSION** | **DATE** | **AUTHOR** | **COMMENTS** |
| 0.1 | 12- Sep-  2021 | Chanchal Gopalakrishnan | Introduction and architecture defined |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**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 07** | |
|  | **3.1** | **Data Description 07** |
|  | **3.2** | **Data Cleaning 08** |
|  | **3.3** | **Dataset Exporting to Excel 08** |
| **4.** | **Unit** | **test cases 08** |

# Introduction

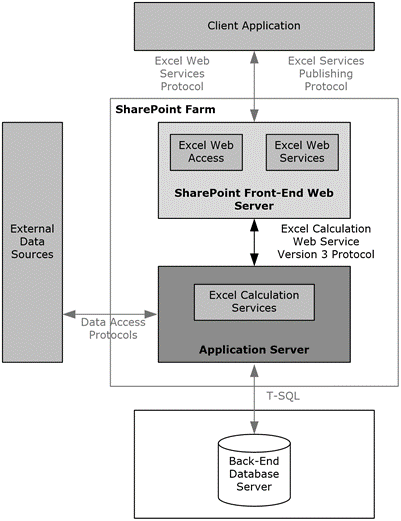
## What is Low-Level design document?

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 dashboard. 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

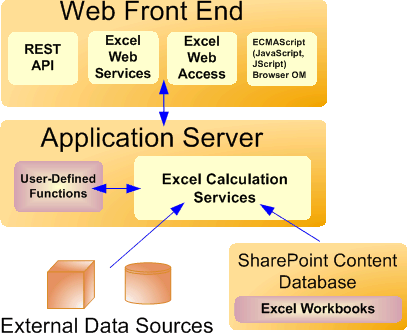


**Excel Server Architecture**

A server technology first introduced in Microsoft Office SharePoint Server 2007 — and its architecture. It also maps out the types of development you can do using Excel Services, discusses unsupported features, and gives an overview of business intelligence in Excel Services.

Excel Services is built on ASP.NET and SharePoint Foundation technologies

The following diagram shows Excel Server’s architecture:



Excel Server is internally managed by the multiple server processes.

**1. Gateway/Load Balancer**

In multiple-server configurations, Excel Services load-balances requests across multiple Excel Calculation Services occurrences in a farm configuration. If your installation includes multiple application servers, Excel Services will balance the load in an attempt to help ensure that no single application server is overloaded by requests.

Administrators can configure the load-balancing behaviour.

1. **Web Front-End Servers and Back-End Application Servers:-**

The Excel Web Access, Excel Web Services, UDFs, JavaScript, the REST service, and Excel Calculation Services components can be divided into components on the Web front-end server and components that live on a back-end application server. The Web front end includes Excel Web Access, JavaScript, the REST service, and Excel Web Services. The Excel Calculation Services component resides on the back-end application server, alongside any UDF assemblies that an administrator has added.

# Architecture Description

## Data Description

The Dataset contains Sales Data of the Amazon retailed during different years by different Sales representatives with discount amount on various categories of Items..etc

* + 1. CustKey: Unique Number of each of the customer buying the products from the Amazon.
    2. DateKey: Date and Month during which the item was bought.
    3. Discount Amount: Discount Amount available on particular item.
    4. Item: Name of the Different Items sold
    5. List Price: The actual price of the item without any discount applied.
    6. Sale Amount: Amount after the discount is applied
    7. Sales Amount Based on List Price: Amount after the discount has been applied on the list price.
    8. Sales Margin Amount: Price after the discount is applied.
    9. Sales Price: The actual amount after the discount at which the item will be sold.
    10. Sales Quantity:Number of items sold.
    11. Sales Rep:Different retailers available for selling the particular items.

## Data Cleaning

In the Cleaning Process, we will convert our original datasets with other necessary attributes format. We will look for duplicate,null and incomplete data within a dataset.

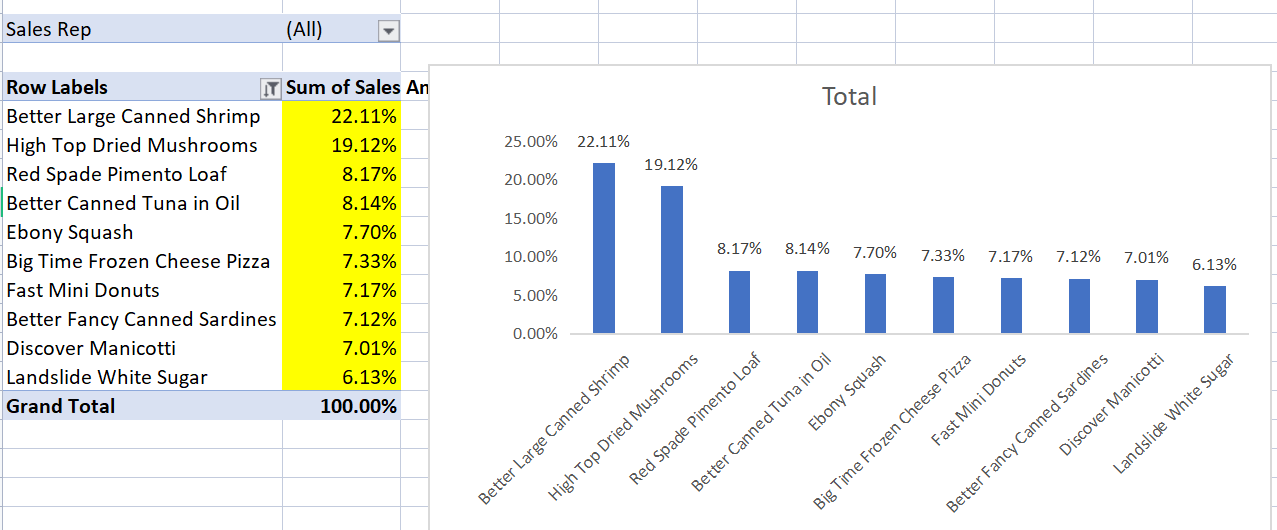
## Data Insertion into Excel

1. Load the dataset into excel.
2. Create sheets and perform various operation like creating pivot according to the requirements

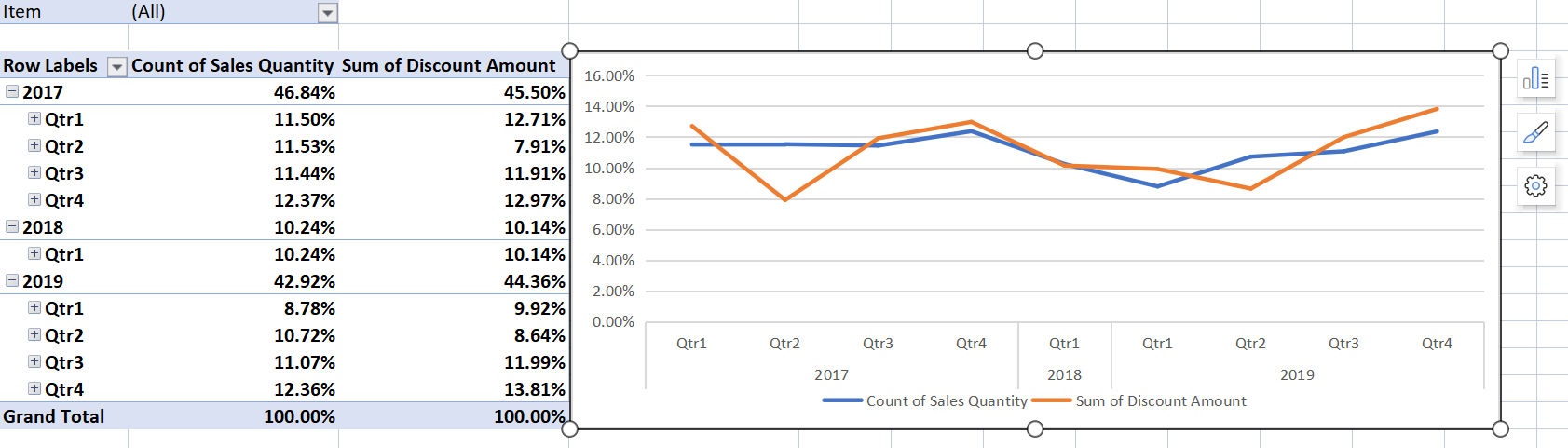
1. **Unit Test Cases**

**Test Case Description & Expected Result:**

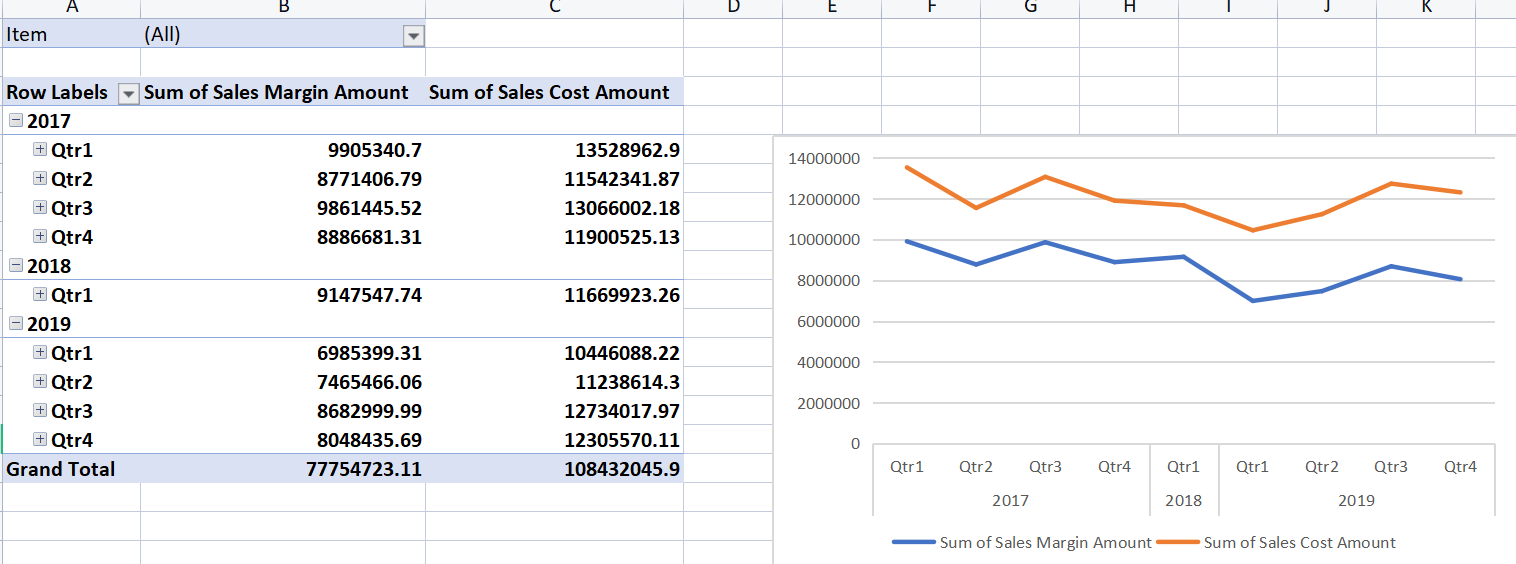
1. **Top 10 Items in Sales:**



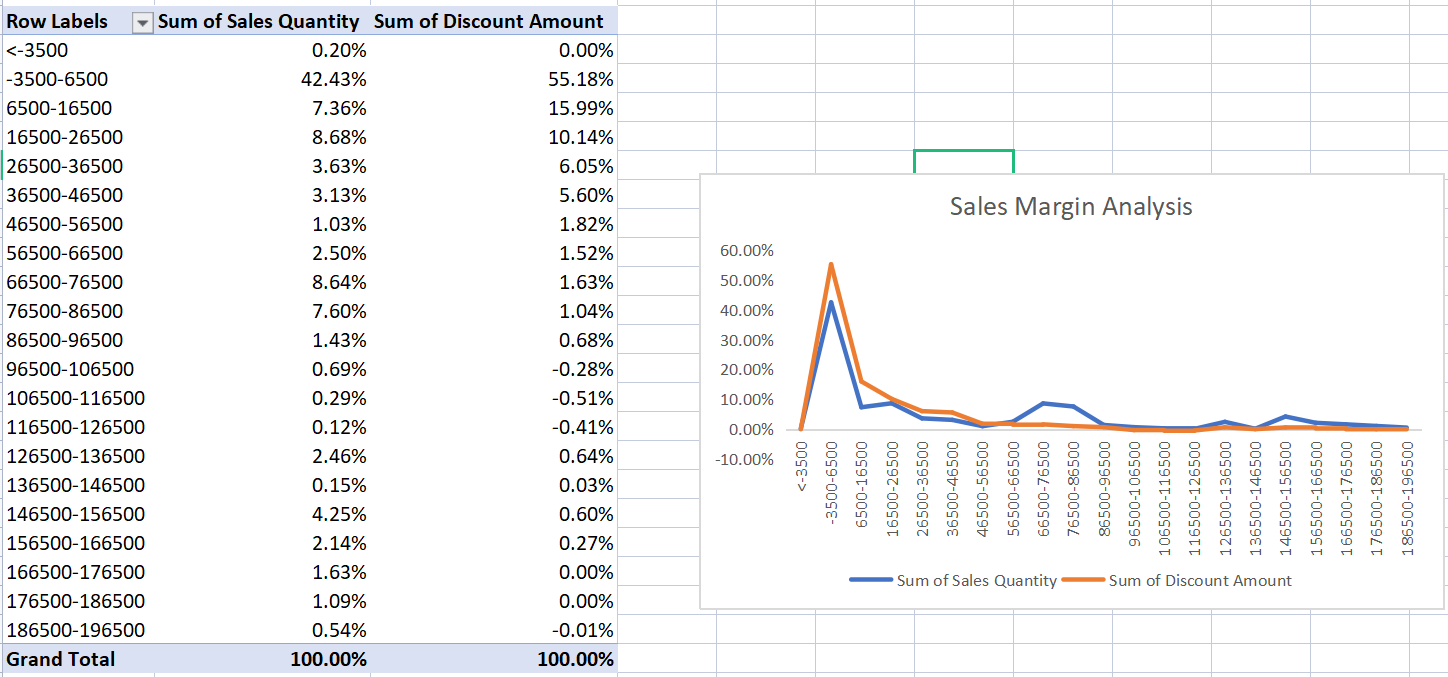
1. **Sales Amount Vs Discount Amount**



1. **Margin Amount Vs Cost Amount**



1. **Sales Margin Analysis**



1. **Number of Sales by Sales Representative**

