

Low level design

Customer Segmentation and Clustering



# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Comments |
| 0.1 | 19-11-2024 |  | Introduction and Architecture defined |
| 0.2 | 19-11-2024 |  | Architecture & Architecture Description appended and updated |
| 0.3 | 26-11-2024 |  | Unit Test Cases defined and appended |

# 1 Introduction

## 1.1 What is Low-Level design document?

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code for Customer Segmentation and Clustering. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document.

## 1.2 Scope

Low-level design (LLD) is a component-level design process that follows a step-by step refinement process. This 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

# 3 Architecture Description

## 3.1 Data Description

The **Online\_retail.csv** dataset is an ideal choice as it is concise and free of unnecessary data. With over 1,00,000 entries, it provides sufficient information to enable the model to gain a comprehensive understanding and deliver accurate insights.

## 3.2 Web Scrapping

To achieve optimal results, it is essential to gather and analyse as many diverse datasets as possible. This helps the model create more accurate and well-defined clusters, enhancing its ability to deliver precise insights.

## 3.3 Export Data from Database

The data in a stored database is exported as a CSV file to be used for Data Pre-processing and Model Training.

## 3.4 Data Preprocessing

Data preprocessing involves checking for **NULL values**, identifying **unique values** in columns, and addressing any **faulty or inconsistent data** to ensure the dataset's quality and reliability for analysis.

## 3.5 Data Clustering

The **K-Means algorithm** will be utilized to create clusters in the pre-processed data. The optimal number of clusters is determined using the **elbow plot** method. Clustering aims to group data points based on their similarities, enabling the implementation of various algorithms to analyse these clusters. The K-Means model is trained on the pre-processed data and subsequently saved for future use in predictions, ensuring consistency and scalability in analysing similar datasets.

## 3.6 Model Building

After creating the clusters, the best model will be identified for each cluster. Algorithms will be applied to each cluster using the optimal parameters determined from the **Elbow Graph**, which provides the ideal number of clusters for the K-Means algorithm. Additionally, the **Silhouette Score** will be used to evaluate the quality of the clustering results. By combining these techniques, we aim to achieve the most accurate and effective clustering outcomes.

## 3.7 Data from User

The model will take CSV file from the user.

## 3.8 Data Validation

The model is designed to address all issues related to validation, ensuring reliable and accurate performance throughout the analysis process.

## 3.9 Data Clustering

The data clustering process is performed using the **Elbow Method** to determine the optimal number of clusters and the **Silhouette Score** to evaluate the quality of the clustering.

## 3.10 Deployment

The model will be uploaded to Heroku.