High-Level Design (HLD)

Customer-Personality-Analysis

Written By	GURURAJ S
Version	1.0
Date	20/05/2023

Document Change Control Record

Version	Date	Author	Comments

Review

Version	Date	Reviewer	Comments

Approval

Version	Review date	Reviewed By	Approved By	Comments

Abstract

Customer Personality Analysis is a detailed analysis of a company's ideal customers. It helps a business to better understand its customers and makes it easier for them to modify products according to the specific needs, behaviors and concerns of different types of customers.

1. Introduction

1.1 Why these High-Level Design Documents?

The purpose of this High-Level Design (HLD) Documents is to add necessary details to the current project description to represent a suitable for coding. This document is also intended to help detect contradictions before coding. And can be used as a reference manual for how the modules interact at a high level.

The HLD will be:

- Present all of the design aspects and define them in detail.
- Describe the user interface being implemented.
- Describe the needed Python libraries for the coding.
- Describe the performance requirements.
- Include design features and the architecture of the project.
- List and describe the non-functional attributes like:
  - Security
  - Reliability
  - Maintainability
  - Portability
  - Reusability
  - Application Compatibility
  - Resource Utilization
  - Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture(layers), application flow (Navigation), and technology architecture, The HLD uses non-technical and mildly-technical terms which should be understandable to the administrators of the system

1.3 Definition

TERM	DESCRIPTION
DB	Database, the cloud platform where the data will be stored. Can be considered cloud storage.
ML	Machine Learning
API OR APIs	Application Programming Interface can be considered a website link from there we can extract information.

2. General Description

2.1 Product Perspective

Customer Personality Analysis is a detailed analysis of a company's ideal customers. It helps a business to better understand its customers and makes it easier for them to modify products according to the specific needs, behaviors and concerns of different types of customers.

2.2 Problem Statement

Customer Personality Analysis is a detailed analysis of a company's ideal customers. It helps a business to better understand its customers and makes it easier for them to modify products according to the specific needs, behaviors and concerns of different types of customers.
Customer personality analysis helps a business to modify its product based on its target customers from different types of customer segments. For example, instead of spending money to market a new product to every customer in the company's database, a company can analyze which customer segment is most likely to buy the product and then market the product only on that particular segment.

2.3 Proposed Solution

Customer Personality Analysis is a detailed analysis of a company's ideal customers. It helps a business to better understand its customers and makes it easier for them to modify products according to the specific needs, behaviors and concerns of different types of customers.
Customer personality analysis helps a business to modify its product based on its target customers from different types of customer segments. For example, instead of spending money to market a new product to every customer in the company's database, a company can analyze which customer segment is most likely to buy the product and then market the product only on that particular segment.

2.4 Data Requirements

The data is required for the building of the project is already available on the dashboard.

People:

- ID: Customer's unique identifier
- Year_Birth: Customer's birth year
- Education: Customer's education level
- Marital_Status: Customer's marital status
- Income: Customer's yearly household income
- Kidhome: Number of children in customer's household
- Teenhome: Number of teenagers in customer's household
- Dt_Customer: Date of customer's enrollment with the company
- Recency: Number of days since customer's last purchase
- Complain: 1 if the customer complained in the last 2 years, 0 otherwise

Products:

- MntWines: Amount spent on wine in last 2 years
- MntFruits: Amount spent on fruits in last 2 years
- MntMeatProducts: Amount spent on meat in last 2 years
- MntFishProducts: Amount spent on fish in last 2 years
- MntSweetProducts: Amount spent on sweets in last 2 years
- MntGoldProds: Amount spent on gold in last 2 years

Promotion:

- NumDealsPurchases: Number of purchases made with a discount
- AcceptedCmp1: 1 if customer accepted the offer in the 1st campaign, 0 otherwise
- AcceptedCmp2: 1 if customer accepted the offer in the 2nd campaign, 0 otherwise
- AcceptedCmp3: 1 if customer accepted the offer in the 3rd campaign, 0 otherwise
- AcceptedCmp4: 1 if customer accepted the offer in the 4th campaign, 0 otherwise
- AcceptedCmp5: 1 if customer accepted the offer in the 5th campaign, 0 otherwise
- Response: 1 if customer accepted the offer in the last campaign, 0 otherwise

Place:

- NumWebPurchases: Number of purchases made through the company's website
- NumCatalogPurchases: Number of purchases made using a catalogue
- NumStorePurchases: Number of purchases made directly in stores
- NumWebVisitsMonth: Number of visits to company's website in the last month

2.5 Tools Used

As far as programming language is concerned, we have utilized python, which provides some of the most powerful libraries and packages in order to predict store sales. These libraries include but are not limited to Pandas, NumPy ,PyCaret, Sweetviz and Pandas-Profiling. Flask web framework along with Spyder has been leveraged for creating a dedicated application and for all modular coding. The team has utilized GitHub for storing all files in order to accomplish easy access and efficient extraction.

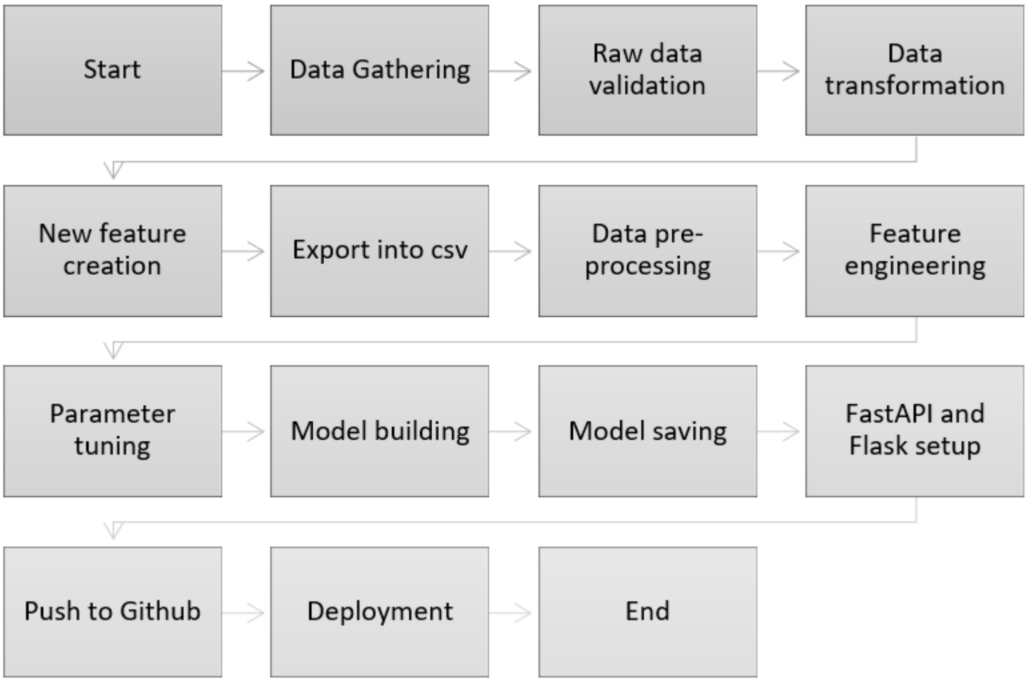
2.6 Constraints

The System should be user-friendly, the user should get all proper messages while using the web app. He/she also should get a proper error message if he/she has done something wrong on the web-app page. All the errors and results should be delivered in the easiest possible way and all the buttons are going to insert on the webpage should be labelled properly, so the user did not get confused to use the system.

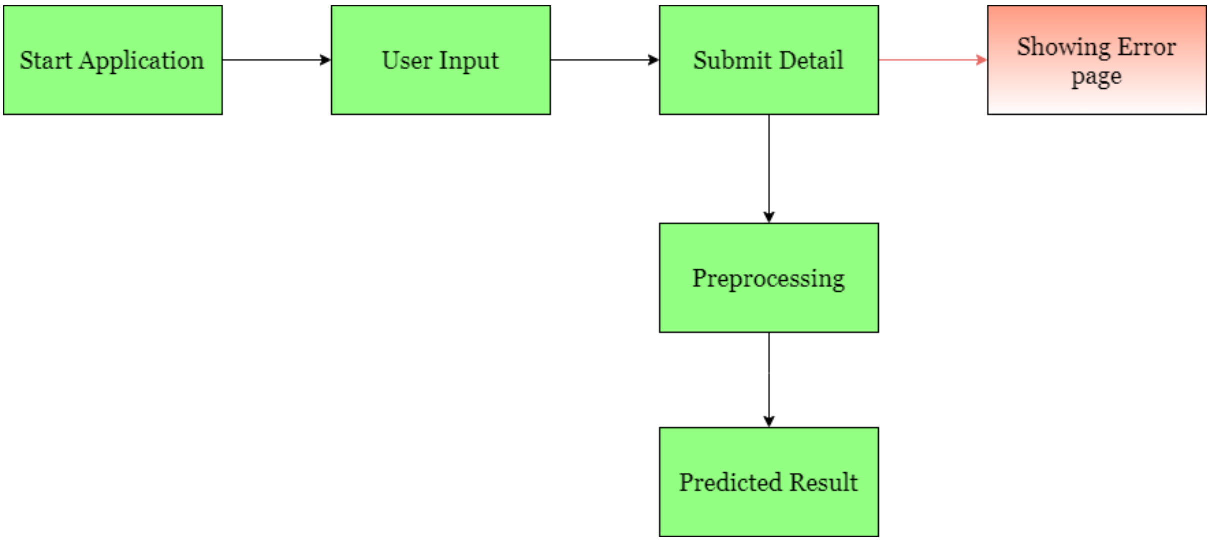
3. Design Details

3.1 Process Flow

The following process flow has been used for this project. This process is based on modular coding i.e., use of OOPS concepts to build the entire end to end project.



3.2 Deployment Process



3.3 Error Handling

If any error occurs in inputting data (in web application), the resulting error message should be concise and understandable by users. This will enable the user to spot data entry errors and rerun the process with necessary amendments. And we have to log every error for our application and have to manage the same. Necessary log files have been created to keep a record of all data entered by the user each time the application is used for predicting item sales.

4. Performance

The Customer-Personality-Analysisis solely dependent on evidence-based machine-learning algorithms. We have trained various ML algorithms (sk_learn) to find the best performing model i.e. decision tree model for clustering in order to predict the target. Our system performance will be based on the data we are going to feed the algorithms. Model performance will involve our finalized model, the web application and the deployment server collectively.

4.1 Reusability

The code and the modules created at the time of building the project is required to maintain all coding guidelines and full project code is written in a modular fashion. Our system incorporates the flexibility to work properly from any location. And it should handle any improper input value from the user by providing a meaningful error message so the user can correct his/her mistake and enter valid input to get the desired results.

4.2 Application Compatibility

The different libraries in Python programming language, CSS, and HTML have been used to build the system. Flask has been used for making the web APIs and HTML/CSS has been leveraged to make the web application. All the components of the application are supposed to work properly and it is required to produce a result without any major impediments in place.

4.3 Resource Utilization

Our application should utilize the given resources efficiently and effectively. It should use a optimal amount of internet to work and call the APIs on the web page. Our system has been designed to use less computational in order to make the application faster. Our application will be deployed cloud platform and it should utilize the resource given on the cloud and work properly.

5. Deployment

For the deployment process, we will be using Render.com cloud platforms for hosting our application. The cloud platform will run the system and it will give the flexibility to use our application globally.

https://cpa-mxyo.onrender.com

6. Conclusion

- Most of the customers are university graduates.
- Most of the customers are living with partners.
- Those living alone have spent more than those living with partners.
- Most of the customers have only one child.
- Those having no children have spent more.
- Middle Age Adults, aged between 40 and 60, are famous age group category.
- Middle Age Adults are spending on average, more than the other age groups.
- Most of the customers are earning between 25000 and 85000.
- Those who are earning more are also spending more.

7. Reference

Google image for collection the logos and images.

Sketch diagram for drawing the diagrams.