**High-Level Design (HLD)**

**Analyzing World’s Best Wine Reviews**

|  |  |
| --- | --- |
| Written By | Akanchha Bagla |
| Version | 1.0 |
| Date | 05/30/2022 |

**Document Change Control Record**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comments** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Review

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Reviewer** | **Comments** |
|  |  |  |  |

**Approval**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Review date** | **Reviewed By** | **Approved By** | **Comments** |
|  |  |  |  |  |

**Abstract**

This project is about analyzing the wine produced around the world. The main goal here is to find a meaningful relationship between different attributes.

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**

The Store Sales Prediction is an ML infused Web Application that is capable of predicting future product demand by analysing past records. It will output a number representing measure of product sales.

**2.2 Problem Statement**

Store sales are very important in jurisdictions with varied population density levels, market demand and outlet types. Some of these factors are substantially responsible for understanding and analysing product demand. Conventional ways include but are not limited to lengthy qualitative survey designs and generic business intuition. Utilization of such behavioural and psychological survey may lead to potential bias and may not be statistically sound. This may, in turn, lead to miscalculated business decisions and may harm revenue streams of the store. This may also lead to demand and supply disequilibrium which has the potential of hurting profits substantially along with incorrect market basket analysis.

**2.3 Proposed Solution**

Our approach is to balance both business intuition and evidence based approach to predict item sales. We will perform initial data exploration to find the important relationships between different features and aligning it with business acumen and understanding. Gradient Boosting Regression has been utilized further to predict the future sales demand. A dedicate web application has been created for the sole purpose of providing insights pertaining to store sales in future with input features getting processed at the back end in order to predict the same. The machine learning algorithm mentioned has been hyperparameter tuned in order to find the best parameters responsible for predicting store sales with minimal error.

**2.4 Data Requirements**

The data is required for the building of the project is already available on the dashboard. The Store Sales Prediction data is recorded many product descriptions along with past sales quantity. For building the ml model we will use the dataset that is given. The data is consisted of 8523 rows and various information about products like product id, product category, store id, store location, etc.

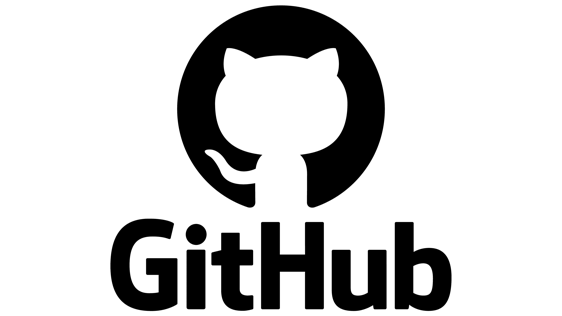
**2.5 Tools Used**

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.









A picture containing icon

Description automatically generated

Logo

Description automatically generated

**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.

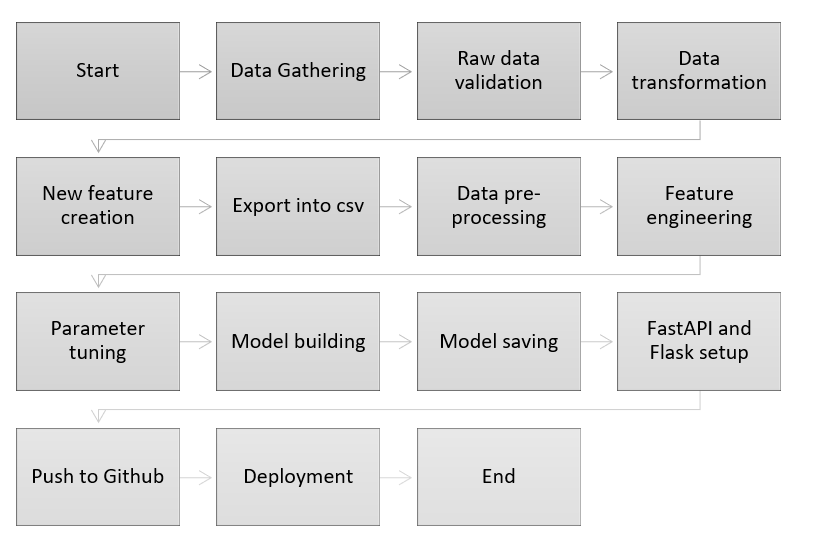
**2.7 Assumptions**

The main objective is to implement a system that will produce approximate future demand for a product in stores.

**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 Stores Sales Price Prediction is solely dependent on evidence-based machine-learning algorithms. We have trained various ML algorithms (via PyCaret) to find the best performing model i.e. Gradient Boosting Regression 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 Heroku cloud platforms for hosting our application. The cloud platform will run the system and it will give the flexibility to use our application globally.



**6. Conclusion**

The Store Sales Prediction is for assisting business owners and other stakeholders to predict product demand in the future. It will help them grow the business profits and it will help the optimize complex supply chain. An evidence based store sales prediction application can be of greater use to minimize the bullwhip effect. The idea is to analyse the past data and use a machine learning induced application to predict the target value/product sales in the future.

**7. Reference**

Google image for collection the logos and images.

Sketch diagram for drawing the diagrams.