LOW LEVEL DESIGN (LLD)
Insurance Premium Prediction

Revision Number: 1.1

Last date -17/03/2024

Chirag

Low Level Design (LLD)

Document Control

Date	Version	Description	Author
16/03/2024	1.0	Introduction, Architecture	Chirag
18/03/2024	1.1	Deployment	Chirag
19/03/2024	1.2	Unit Test Case	Chirag

Low Level Design (LLD)

Contents

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 Food Recommendation System. 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
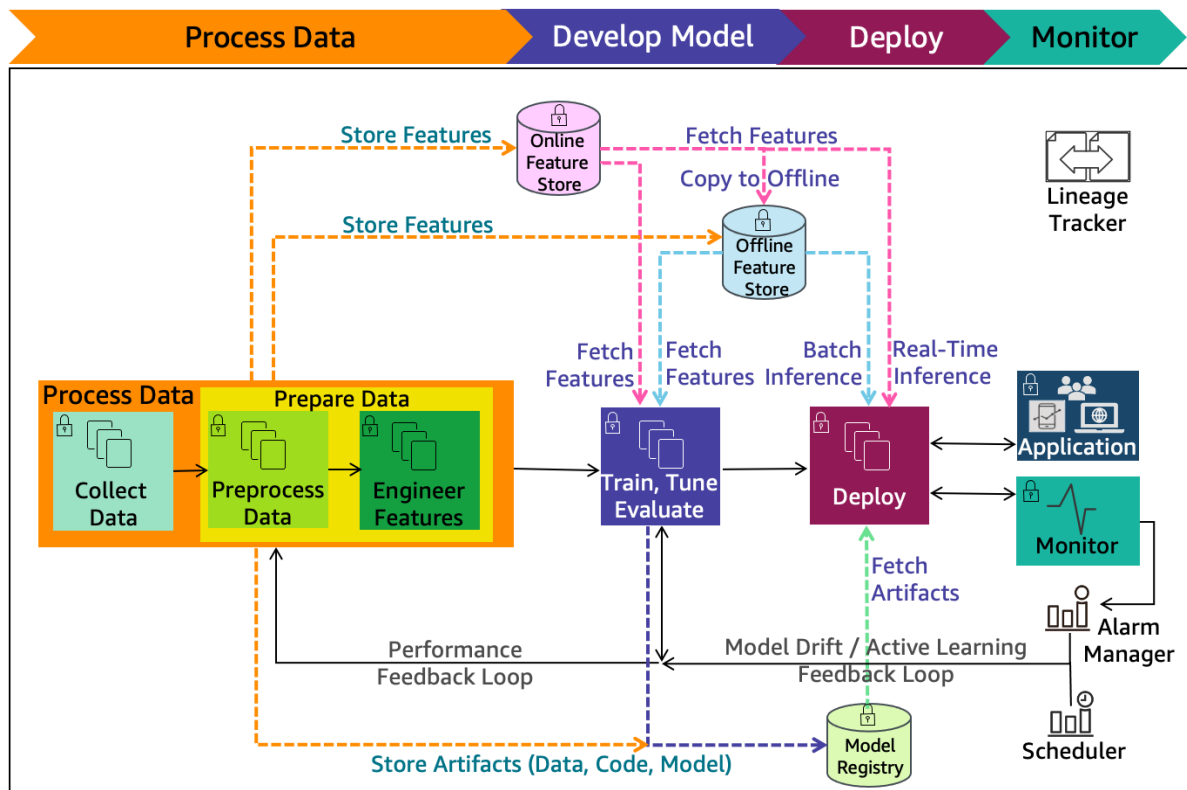
2. Architecture

3. Architecture Description

3.1. Data Description:

Provides an overview of the data utilized in the system, detailing its sources, formats, and key attributes.

3.2. Web Scraping:

Explains the process of extracting data from web sources using specialized tools and techniques, facilitating data acquisition for analysis and processing.

3.4. Data Insertion into Database:

Defines procedures for securely inserting processed data into the database, ensuring data integrity and reliability.

3.5. Export Data from Database:

Explains methods for extracting data from the database for analysis or reporting purposes, maintaining data consistency.

3.6. Data Pre-processing:

Details preprocessing techniques like scaling or encoding applied to the data to prepare it for modeling.

3.7. Data Clustering:

Describes algorithms used to group similar data points together, aiding in data analysis and pattern recognition.

3.10. Model Building:

Outlines the process of constructing predictive models using machine learning algorithms, including validation and refinement.

3.11. Data from User:

Specifies how user-provided data is collected and integrated into the system for further analysis and processing.

3.12. Data Validation:

This section outlines the protocols and criteria used to validate incoming data, ensuring its accuracy and adherence to predefined standards. It covers techniques for detecting and handling data anomalies to maintain data integrity.

3.13. User Data Inserting into Database:

Here, the process of inserting user-provided data into the database is described, including validation steps to verify data accuracy and reliability. It also addresses error handling mechanisms to manage any issues encountered during data insertion.

3.14. Data Clustering:

This section provides additional insights into the data clustering process, discussing any optimizations or enhancements implemented to improve clustering accuracy and efficiency. It may include discussions on algorithmic improvements or parameter tuning.

3.15. Model Call for Specific Cluster:

Describes how specific models are selected and invoked based on the clustering results to provide personalized recommendations. It may include

details on model selection criteria and the decision-making process for recommending relevant models.

3.16. Recipe Recommendation & Saving Output in Database:

Outlines the process of recommending recipes to users based on their data and saving the recommendation output in the database for future reference. It may include discussions on recommendation algorithms and mechanisms for storing recommendation data securely.

3.17. Deployment:

Specifies the steps and procedures for deploying the system in a production environment, ensuring scalability and reliability. It covers deployment strategies, testing protocols, and monitoring mechanisms to ensure smooth system operation in a live environment.

4. Unit Test Cases:

Test Case Description	Pre-Requisite	Expected Result
Verify whether the Application URL is accessible to the user	1. Application URL should be defined	Application URL should be accessible to the user
Verify whether the Application loads completely for the user when the URL is accessed	1. Application URL is accessible 2. Application is deployed	The Application should load completely for the user when the URL is accessed
Verify whether the User is able to sign up in the application	1. Application is accessible	The User should be able to sign up in the application
Verify whether user is able to successfully login to the application	1. Application is accessible 2. User is signed up to the application	User should be able to successfully login to the application
Verify whether user is able to see input fields on logging in	1. Application is accessible 2. User is signed up to the application 3. User is logged in to the application	User should be able to see input fields on logging in
Verify whether user is able to edit all input fields	1. Application is accessible 2. User is signed up to the application 3. User is logged in to the application	User should be able to edit all input fields

Verify whether user gets Submit button to submit the inputs	1. Application is accessible 2. User is signed up to the application 3. User is logged in to the application	User should get Submit button to submit the inputs
Verify whether user is presented with recommended results on clicking submit	1. Application is accessible 2. User is signed up to the application 3. User is logged in to the application	User should be presented with recommended results on clicking submit
Verify whether the recommended results are in accordance to the selections user made	1. Application is accessible 2. User is signed up to the application 3. User is logged in to the application	The recommended results should be in accordance to the selections user made
Verify whether user has options to filter the recommended results as well	3. User is logged in to the application	User should have options to filter the recommended results as well
Verify whether KPIs modify as per the user inputs for the user's health	1. Application is accessible 2. User is signed up to the application 3. User is logged in to the application	KPIs should modify as per the user inputs for the user's health
Verify whether the KPIs indicate details of the suggested recipe	1. Application is accessible 2. User is signed up to the application 3. User is logged in to the application	The KPIs should indicate details of the suggested recipe