Low Level Design

(LLD)

Insurance Premium Prediction

Written by	Jayanto Kumar Bhakat
Document Version	0.1
Last Revised Date	15 Jan 2024

Document Control

Change Record:

Version	Date	Author	Comments
0.1	15/01/2023	Jayanto Kumar Bhakat	Initial HLD – V1.0

Reviews:

Version	Date	Reviewer	Comments

Approval Status:

Version	Review Date	Reviewed By	Approved By	Comments

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

START ------> DATA GATHERING -----> EDA ----> DATA CLEANING-----> FEATURE ENGINEERING -----> MODEL CREATION -----> MODEL TESTING ------> FLASK SETUP -----> DEPLOYMENT

3 ARCHITECTURE DESCRIPTION

3.1 Data Description

The primary source of data for this project from Kaggle. The dataset is comprised of 1338 records with six attributes. The data is in a structured format and stored in a CSV file.

3.2 Exploratory Data Analysis

Exploring the data by visualizing the distribution of values in some columns of a dataset and the relationships between expenses and another column. Visualizing, the distribution of Age and BMI (body mass index). Also, check the region-wise have any differences in the expenses.

3.3 Data Pre-processing

If data is not suited to take place directly for the regression. Then, cleaning of dataset becomes important for using the data under various regression algorithms.

3.4 Model Building

After data pre-processing is completed, we will split the dataset into training sets and validation sets. Then we will use a training set for building the best model. The model will be trained on several algorithms. We will calculate RMSE and r2 scores for each model and select the model with the best score.

3.5 Data Validation

Here Data Validation will be done on the test set.

3.6 Deployment

We will be deploying the model to the Azure platform.

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 user can edit all the input fields	1. Application is accessible	User should be able to edit all the input fields
Verify whether user has options to filter the inputs fields	1. Application is accessible	User should filter the options of input fields
Verify whether user gets submit button to submit the inputs	1. Application is accessible	User should get submit button to submit the inputs
Verify whether user can see the output after submitting the inputs	1. Application is accessible	User should get outputs after submitting the inputs