HIGH LEVEL DESIGN (HLD)

Insurance Premium Prediction

**ARKADEV CHAKRABARTI**

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
| **06.11.2022** | V1.0 | Initial HLD- V1.0 | ARKADEV CHAKRABARTI |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[Document Version Control 1](#_Toc10259)

[Abstract 3](#_Toc10260)

[1.0 Introduction 4](#_Toc10261)

[1.1 Why this High-Level Design Document? 4](#_Toc10262)

[1.2 Scope 4](#_Toc10263)

[1.3 Definitions 5](#_Toc10264)

[2.0 General Description 6](#_Toc10265)

[2.1 Product Perspective 6](#_Toc10266)

[2.2 Problem Statement 6](#_Toc10267)

[2.3 Proposed Solution 6](#_Toc10268)

[2.4 Further Improvements 7](#_Toc10269)

[2.5 Technical Requirements 7](#_Toc10270)

[2.6 Data Requirements 7](#_Toc10271)

[2.7 Tools Used 8](#_Toc10272)

[2.8 Constraints 9](#_Toc10273)

[2.9 Assumptions 9](#_Toc10274)

[3.0 Design Details 10](#_Toc10275)

[3.1 Process Flow 10](#_Toc10276)

[3.2 Event Log 10](#_Toc10277)

[4.0 Performance 11](#_Toc10278)

[4.1 Reusability 11](#_Toc10279)

[4.2 Application Compatibility 11](#_Toc10280)

[4.3 Deployment 11](#_Toc10281)

[5.0 Conclusion 13](#_Toc10283)

# Abstract

Insurance premium prediction the goal of this project is to give people an estimate of how much they need based on their individual health situation. The dataset is regression problem. Six regression models naming Linear Regression, RandomForestRegression, AdaBoost Regressor, KNN have been used to compare and contrast the performance of these algorithms. Later accuracies of all these models were compared. The RandomForestRegression gives very good accuracy that’s why we choose the RandomForestRegression for this project.

# 1.0 Introduction

## 1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add necessary details to current project description to represent a suitable model for coding. This document is used as a reference manual for how the model interact at a high-level.

### The HLD will

* Presents all design aspects and define them in detail.
* Describe the user interface being implemented.
* Describe the hardware and software interfaces.
* Describe the performance requirements.
* Include design feature and the architecture of the project.

## 1.2 Scope

The HLD document presents the structure of the system, such as the database architecture, application architecture, and technology architecture. The HLD uses non-technical to middle-technical terms which should be understandable to the administrators of the system.

## 1.3 Definitions

|  |  |  |
| --- | --- | --- |
| **Term** |  | **Description** |

Collection of all the information

Database

Integrated Development Environment

IDE

Application Programming Interface

API

Key Performance Indicator

KPI

Visual Studio Code

VS Code

Exploratory Data Analysis

EDA

KNearest Neighbors

KNN

DVC Data Version control

# 2.0 General Description

## 2.1 Product Perspective

The Insurance Premium Prediction is a machine learning based predictive model which will help us to predict the expenses of the health insurance.

## 2.2 Problem Statement

To develop an API interface to predict the expenses of insurance using people individual health data:

* BMI value affects the expenses of insurance.
* Smoking affects a lot to expenses of insurance.
* To create API interface to predict expenses of the premium insurance.

## 2.3 Proposed Solution

The solution proposed here is an estimating expense of insurance based on people health data and this can be implemented to perform above mention use cases. In first case, analyzing how BMI value affect the people health as well as expenses of the insurance. In the second case, we analyze that smoking affects expenses a lot. And in the last use case, we will be making an interface to predict the expenses.

## 2.4 Further Improvements

## 2.5 Technical Requirements

The solution can be a cloud-based or application hosted on an internal server or even be hosted on a local machine. For accessing this application below are the minimum requirements:

* Good internet connection.
* Web Browser.

For training model, the system requirements are as follows:

* +4 GB RAM preferred
* Operation System: Windows, Linux, Mac
* Visual Studio Code / Jupyter notebook

## 2.6 Data Requirements

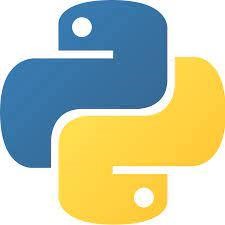
Data requirements completely depends on out problem statement.

* Comma separated values (CSV) file.
* Input file feature/field names and its sequence should be followed as per decided.

## 2.7 Tools Used

Python programming language and frameworks such as NumPy,

Pandas, Scikit-learn, Matplotlib, Seaborn, and Flask are used to build the whole model.



* Pandas is an open-source Python package that is widely used for data analysis and machine learning tasks.
* NumPy is most commonly used package for scientific computing in Python.
* Seaborn and Matplotlib is an open-source data visualization library used to create interactive and quality charts/graphs.
* Scikit-learn is used for a machine learning.
* Flask is used to build API.
* VS Code is used as IDE (Integrated Development Environment)
* GitHub is used as version control system.
* Front end development is done using HTML/CSS.
* Heroku is used for deployment of the model.

## 2.9 Assumptions

The main objective of the project is to develop an API to predict the expense of insurance. Machine learning based Regression model is used for predicting above mentioned cases on the input data.

# 3.0 Design Details

## 3.1 Process Flow

Start

Data

Collection

EDA

Data

Cleaning

Feature

Engineering

Model

Building

Model

Testing

Flask Setup

Deployment

## 3.2 Event Log

The system should log every event so that the user will know what process is running internally.

**Initial Step-By-Step Description:**

* The system identifies at what step logging required.
* The system should be able to log each and every system flow.
* Developer can choose logging method. You can choose database logging.

System should not hang out even after using so many loggings.

# 4.0 Performance

## 4.1 Reusability

The entire solution will be done in modular fashion and will be API oriented. So, in the case of the scaling the application, the components are completely reusable.

## 4.2 Application Compatibility

The interaction with the application is done through the designed user interface, which the end user can access through any web browser.

## 4.3 Deployment



# 

# 5.0 Conclusion

This system shows us that which mushroom is edible or not. After analyzing we got to know that if the population is several and spore print color is white or chocolate the mushroom is not edible. Accuracy, which plays a key role in prediction-based system. From the results we could see that most of the machine learning model works absolutely fine but we are use random forest model for this problem in terms to the accuracy. Our predictions help user to know whether he should choose that mushroom or not, or we can say it tells whether the mushroom is edible or not.