

HIGH LEVEL DESIGN (HLD)

HIGH LEVEL DESIGN (HLD)

Insurance Premium PredictionHIGH LEVEL DESIGN (HLD)

HIGH LEVEL DESIGN (HLD)

# Insurance Premium Prediction

NIKITA MAKARAND PANDE

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
| **0.1.10.2021** | V1.0 | Initial HLD- V1.0 | NIKITA |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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

We use data analysis to predict the insurance premium for individual customers. We have three regression models – a Linear Regression, Gradient Boosting Regression, and Random forest is a machine-learning algorithm that has been use to compare and contrast the performance of various regression models.

The dataset was use as training data for a model, and that model helped to generate predictions. The predicted amount was then compare with the actual data in order to test and verify the accuracy of the model. Later, accuracies for all these models were assess and it was find that Gradient Boosting (GB) and Random Forest algorithms performed better than the remaining models.

In this situation, gradient boosting is the most accurate model because it achieves a score that is comparable to other models.

# 1.0 Introduction

## 1.1 Why this High-Level Design Document?

The main purpose of this document is to provide additional details about the model that will be use in coding the project. This document can serve as a reference for how the model behaves 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 provides a detailed overview of the system's architecture, including its database and application architectures. Additionally, it uses non-technical terms which should be easily understood by administrators who manage the system.

## 1.3 Definitions

|  |  |
| --- | --- |
| Term | Description |
| Database | Collection of all the information |
| IDE | Integrated Development Environment |
| API | Application Programming Interface |
| KPI | Key Performance Indicator |
| VS Code | Visual Studio Code |
| EDA | Exploratory Data Analysis |
|  |  |

# 2.0 General Description

## **2.1 Product Perspective**

The insurance premium estimation process is a machine learning-based prediction model that will help us to predict the cost of personal health insurance.

## **2.2 Problem Statement**

To develop an API interface that can predict the premium of insurance using people's individual health data, to analyze the following:

* The premium for people with a BMI value that is above or below a certain threshold will be affect.
* Smoking affects the premiums that companies are willing to pay for insurance policies designed to prevent smoking-related illnesses.
* To create an API interface that can predict the premium for a product

## **2.3 Proposed Solution**

The solution proposed here is an insurance premium based on people's health data. This can be implement to perform various use cases, such as analyzing how a person's BMI value affects their health and the cost of the insurance, warning people if they are smoking and causing them to pay higher premiums, or making an interface that predicts what a particular premium would be for someone in specific circumstances.

**2.5 Technical Requirements**

To access the solution, you will need either a cloud-based or an application-hosted service. The requirements for accessing this software are minimal:

* 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**

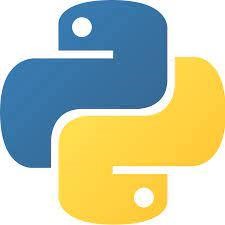
Data requirements completely depends on out problem statement.

* 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, Plotly, Flask 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.
* Plotly 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.

## **2.8 Constraints**

The app can provide insurance prices based on the user's specific information, such as their BMI and sex, as well as whether they are a smoker or not.

## **2.9 Assumptions**

The main objective of the project is to develop a platform that can predict the premium people will receive based on their health information. Machine learning-based regression models are used for this purpose, as input data.

# 3.0 Design Details

## **3.1 Process Flow**

End

Model Building

Visualization

Start

Deployment

Model Testing

Data Cleaning

Export data from csv

Flask Setup

Prediction

Data Transformation

Data Preprocessing

## **3.2 Event Log**

The system should keep track of every event so that the user will be aware of what processes are running inside it.

An overview of the initial step-by-step process for completing the task.

* The system identifies which step in the logging process is required.
* The system should be able to log each and every system flow.
* Database logging is an option for developers. They can choose between different methods.

The system should not keep logging in after using so many logins.

# 4.0 Performance

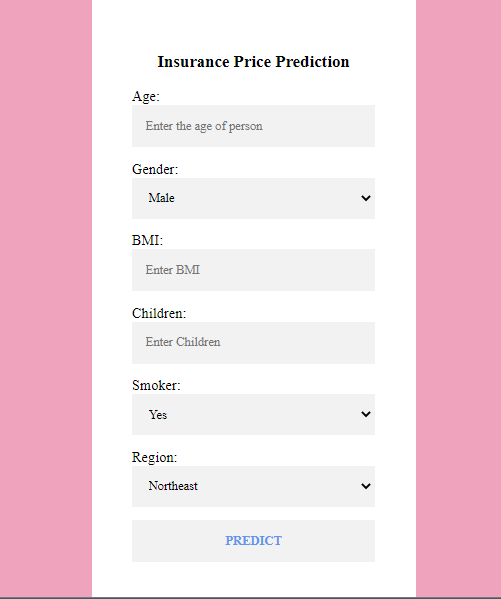
## **4.1 Reusability**

The entire solution will be implement in modular fashion, making it easy to scale up the application. Additionally, the components will be accessible through an API, so that modifications can easily be made if necessary.

**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 the different techniques that are used in order to estimate the how much amount of premium required on the basis of individual health situation. After analyzing it shows how a smoker and non-smokers affecting the amount of estimate. Also, significant difference between male and female expenses. Accuracy, which plays a key role in prediction-based system. From the results we could see that Gradient Boosting turned out to be best working model for this problem in terms of the accuracy. Our predictions help user to know how much amount premium they need on the basis of their current health situation.