High-Level Design (HLD)

High-Level Design (HLD)

Insurance Premium Prediction

Document Version Control

| Date Issued | Version | Description | Author |
| --- | --- | --- | --- |
| 29 July 2022 | 1 | Initial HLD-v1.0 | Mohamed Naji Aboo |
|  |  |  |  |

[**Abstract**](#_4w2r7h180fzh) **4**

[**1 Introduction**](#_jx5q4t7bnzf9) **4**

[1.1 Why this High-Level Design Document?](#_f4huffr566vc) 4

[1.2 Scope](#_b7w3p6uevc8) 5

[1.3 Definitions](#_sf0ec1qz5bjr) 5

[**2 General Description**](#_qx9anglz7g07) **6**

[2.1 Product Perspective](#_rgar8pi6wp4c) 6

[2.2 Problem statement](#_u8jfkj436gdk) 6

[2.3 PROPOSED SOLUTION](#_4ni6kpcgm0y6) 6

[2.4 Data Requirements](#_qi1lc8ka5n3w) 6

[**3 Design Details**](#_9xw5vfvttlrg) **8**

[3.1 Process Flow & Data training & Evaluation](#_nsz0i1yr22wo) 8

[3.2 Error Handling](#_81m9zmwngexx) 9

[**4. Conclusion**](#_1t2x6habopod) **9**

# 

# Abstract

Purpose of the application is to predict the insurance premium. Insurce premium will vary depending on various features. This application will help the common person to understand the insurance premium.

# 1 Introduction

## 1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

* Present all of the 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 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 to mildly-technical terms which should be **understandable to the administrators of the system.**

## 1.3 Definitions

| **Term** | **Description** |
| --- | --- |
| IDE | Integrated Development Environment |
| Heroku | Cloud Platform |
| CICD | Continues Integration and Deployment |
| github | Source repository |

# 2 General Description

## 2.1 Product Perspective

Insurance Premium prediction is a machine learning-based model which will help us to predict the cost of an insurance premium

## 2.2 Problem statement

**To create an** Al solution for insurance premium prediction

## 2.3 PROPOSED SOLUTION

The solution proposed here is to machine learning regression model that can predict the insurance premium.

## 2.4 Data Requirements

We shall download the data from Kaggle, which has the following fields

| Feature Name | Datatype |
| --- | --- |
| age | float |
| sex | category |
| bmi | float |
| children | int |
| smoker | bool |
| region | category |
| charges | float |

Sex category have following values

* male
* female

Region category has following values

* southwest
* southeast
* northwest
* northeast

**Smoker has two values**

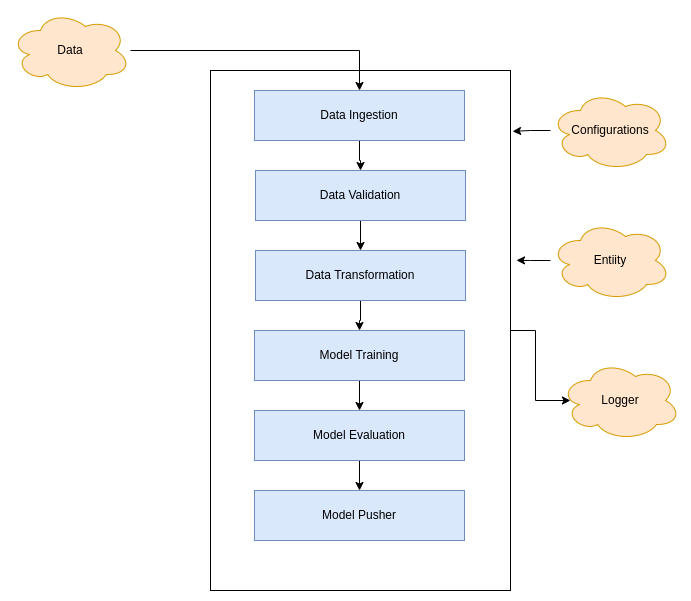
* True
* False

**2.7 Tools used**

* Python
* Numpy
* Sklearn
* Flask
* Git
* Heroku
* Visual Studio code as IDE

# 3 Design Details

## 3.1 Process Flow & Data training & Evaluation



* Data Ingestion
  + Here we will download the dataset from the repo
  + We will split it to test and train data frame
* Data Validation
  + Will do the validation
  + We shall ensure that the features are same
  + We shall ensure the datatypes are matching
  + We shall ensure categories are matching
* Data Transformation
  + Here will do the transformation
  + Empty values shall be replaced with the median for integer/float datatypes and for the category it shall be replaced with most frequent
* Model Training
  + In model training, We shall run the model based on the configuration
  + We shall be adding multiple regression model configuration
  + Based on the best accuracy, we shall be selecting the model
* Model Evaluation
  + Model shall be evaluated for train and test data
  + New model shall be compared with the last model and based on the performance we shall be selecting the model
* Model Pusher
  + Once the new model has higher accuracy, it shall be pushed to the location

## 3.2 Error Handling

All the errors are logged to the log file

# 4. **Conclusion**

The ML model will help us to calculate the insurance premium cost