High Level Design (HLD)

Insurance Premium Prediction

Revision Number: 0.2
Last date of revision: 14/06/2023

Document Version Control

Date Issued	Version	Description	Author
10/06/2023	0.1	Initial HLD	Vijay shinde
12/06/2023	0.2	Design Flow	Vijay shinde
14/06/2023	0.2	Performance Evaluation	Vijay shinde

Contents

Abstract

The objective of our project is to provide individuals with an estimate of their health insurance needs based on their individual health situation. By considering variables such as age, sex, BMI, number of children, smoking habits, and living region, we aim to predict the insurance premium. With this information, customers can make informed decisions when working with any health insurance carrier, considering their plans and perks while keeping the projected cost from our study in mind. This approach allows individuals to focus on the health aspects of an insurance policy, rather than being overwhelmed by ineffective elements.

1. Introduction

Why this High-Level Design Document?

The High-Level Design Document serves as a blueprint or roadmap for the development and implementation of a project or system. It provides an overview of the project's architecture, key components, interactions, and functionalities at a high level

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 purpose of the High-Level Design Document is to:

- Provide a clear understanding of the project's overall structure
- Define the project's scope and boundaries
- Guide the development process
- Describe the user interface being implemented
- Describe the 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

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.

Definition

Term	Description
IPP	Insurance Premium Prediction
Jupyter - Notebook	It is an interactive computational environment where code execution, text, plots, and rich media can be combined seamlessly.
Stream lit	It is a Platform as a Service (PaaS) that empowers developers to build, run, and manage applications in the cloud.

2. General Description

Product Perspective

The Insurance Premium Prediction is a machine learning model designed to assist users in understanding their insurance premium prices based on input data. By leveraging various factors and variables, the model provides insights and estimates regarding the expected cost of insurance premiums. This enables users to make informed decisions and gain a better understanding of the pricing factors involved in their insurance coverage.

Problem statement

The main objective of this model is to predict the insurance premium price based on input data such as BMI, gender, age, and other relevant factors. By utilizing these variables, the model aims to provide accurate estimations of insurance premium costs, allowing individuals to make informed decisions regarding their insurance coverage.

Proposed Solution

To address the problem, we have developed a user interface that allows users to input relevant information for predicting the insurance premium price. The user interface facilitates the collection of necessary data from the user, which is then processed using our trained machine learning model. Once the input is processed, the model generates a predicted value for the insurance premium price. Finally, this predicted value is communicated back to the user, providing them with an estimation of their insurance premium based on the input provided.

Technical Requirements

Regarding technical requirements, there is no need for specialized hardware for the virtualization of the application. Users only require a device with web access and a basic understanding of providing input. On the backend side, a server is necessary to execute all the required packages for processing the input and generating the desired output.

Data Requirements

The data requirements align perfectly with the project's objective, and the dataset is available on Kaggle in the (.zip) file format. To simulate real-time scenarios, we transformed the data into a MongoDB database and exported it into CSV format for further processing.

Tools used

The Python programming language and various frameworks such as NumPy, Pandas, Scikit-learn, Flask, and VS Code were utilized to develop the entire model. These tools and frameworks played a crucial role in implementing the necessary functionalities and components of the model.

- VS Code is utilized as the Integrated Development Environment (IDE).
- Matplotlib, Seaborn, and Plotly are employed for visualizing plots.
- Streamlit is used for deploying the model and developing the front-end.
- Python Flask is utilized for backend development.
- GitHub serves as the version control system.

Constraints

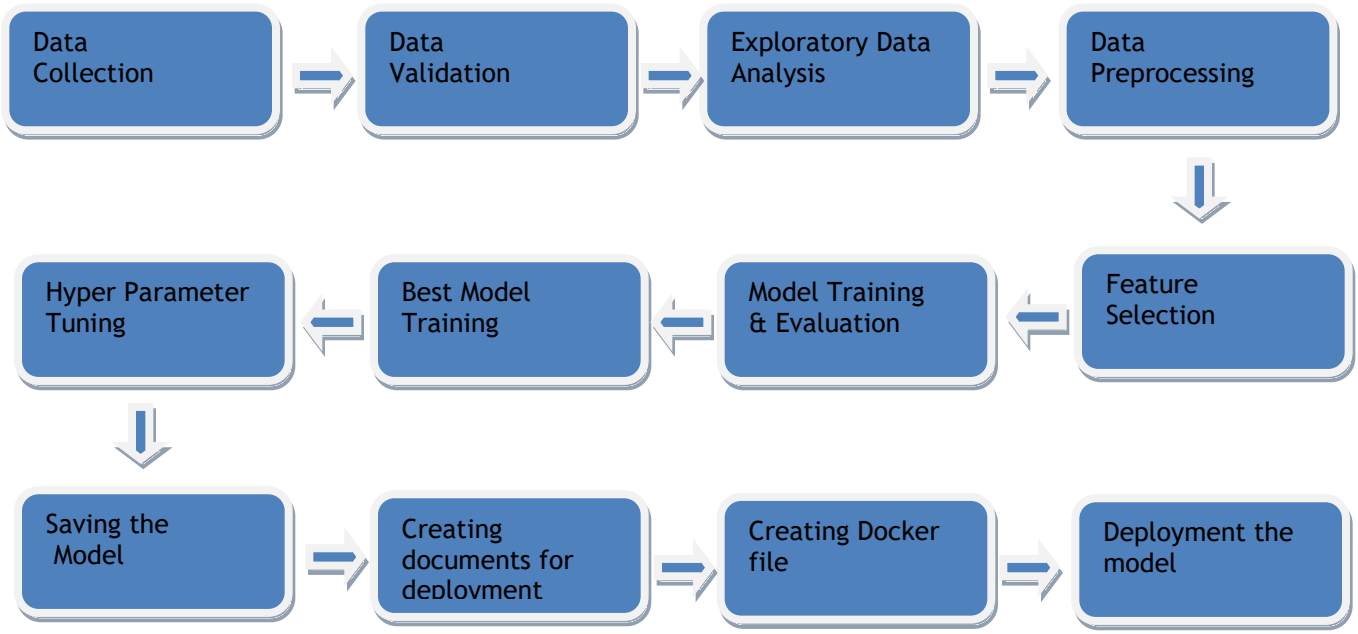
The system should be designed to be user-friendly and as automated as possible, ensuring that users are not required to have any knowledge of the underlying workings.

Assumptions

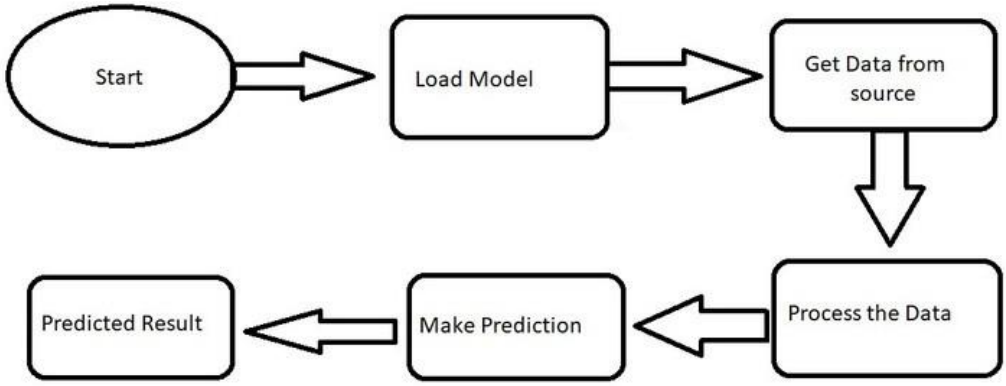
The primary objective of the project is to implement the previously mentioned use cases (as stated in section 2.2 Problem Statement) for new datasets obtained from a source. A Machine Learning-based model is employed to detect the aforementioned use cases based on the input data. It is assumed that all aspects of this project possess the capability to seamlessly integrate and function as expected by the designer.

3. Design Details

Process Flow



Deployment Process



Logging

In logging, whenever an error or exception occurs, the event is logged into the system log file along with the corresponding reason and timestamp. This logging mechanism assists developers in debugging system bugs and rectifying errors effectively..

Error Handling

If any errors are encountered, an explanation will be provided to clarify the cause of the issue. In this context, an error is defined as any occurrence that deviates from the normal and intended usage. The aim is to provide users with clear insights into what went wrong and help them understand the nature of the error.

4. Performance Evaluation

The Insurance Premium Prediction project, based on machine learning, utilizes input data such as age, BMI, sex, and more to predict insurance premiums accurately.

Reusability

The code and components utilized in the project should be designed with reusability in mind, ensuring that they can be reused without any issue

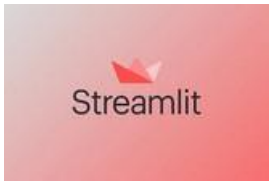
Application Compatibility

The various components of this project will utilize Python as the interface between them. Each component will have its designated task to perform, and Python will facilitate the seamless transfer of information between these components..

Resource Utilization

During the execution of any task, it is expected that all available processing power will be utilized until the completion of that specific function. This ensures efficient and optimal utilization of computational resources while maximizing the speed and performance of the task.

Deployment



5. Conclusion

The Insurance Premium Prediction system is designed to provide price predictions, offering customers valuable insights based on a trained knowledge base and a set of predefined rules. Users can leverage this system to estimate the approximate value of their insurance premiums, enabling them to make informed decisions regarding their insurance coverage.