Project:

Insurance Premium Predictor

By-Mr.Aniket Dumbre

Prepared By	Approved By
Mr.Aniket Dumbre	Mr.Sunny Savita

Prepared By	Approved By
Mr.Aniket Dumbre	Mr.Sunny Savita

1. Abstract:

The Healthcare industry is of the prime sector in the insurance industry. There are many insurance companies present in markets which offer multiple health policies depending upon customer requirement. Suggested work's goal is to anticipate a person's insurance costs using ML based predictor system with personal information and medical information, regardless of whether or not they have any health problems. Multi-Linear, SVM, KNN, Decision Tree, Random Forest, Gradient Boosting, XG Boost Regression were some of the regression models used in this study. After comparing the accuracies, it was determined that Random Forest was the most accurate of all the methods, with an accuracy of 88 percent. Finally, the best model deployed on cloud where client can estimates the insurance premium.

2. Introduction

2.1 Why this High – Level Design Document?

The purpose of this High-Level Design Document is to add the necessary details to the current project description to represent a suitable model for coding. This document is also intended to help to 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 details.
- 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

2.2 Scope

The HLD documentation presents the structure of the system, such as database 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.

2.3 Definitions

Terms	Descriptions
Database	Collections of all information monitored by the system
IDE	Integrated Development Environment
AWS	Amazon Web Services
KPI	Key Performance Indicator
VS Code	Visual Studio Code
EDA	Exploratory Data Analysis
KNN	KNearest Neighbors

3. General Description

3.1 Product Perspective

The insurance premium predictor is a ML based model which will help us to predict the insurance premium based on individual age, gender, smoking habits, no. of children's, bmi, region , sex etc.

3.2 Problem Statement

To develop an API interface to predict the premium of insurance using people individual health data and analyzing the following features:

1. Age
2. Sex
3. Smoking habits
4. no. of children's
5. bmi
6. region

3.3 Dataset Description

Dataset have 3 numerical features (age, bmi, children etc), 3 categorical features (sex, smoker, and region) and target feature 'expenses' as numerical feature.

3.4 Attribute Information

1. age – age of individuals in years
2. sex – Male/Female
3. bmi – body mass index
4. no of children – no. of children person have
5. smoker – whether person smokes or not
6. region – 4 region
7. expenses – expenses of person on health insurance

3.5 Proposed solution

To create an end to end cloud application with CICD pipelines using docker, github so that there is always best model deployed in system.

3.6 Further Improvements

The accuracy of model can be further improved by increasing data and hyper parameter tuning. The project code is written in modular fashion so that whenever new data will come the model will undergo training and select best model depending upon accuracy for prediction.

3.7 Technical Requirements

The solution is cloud-based. For accessing this application below are the minimum requirements:

- Good internet connection.
- Desktop/Laptop
- 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

3.8 Data Requirements

- We need balanced data in .csv format
- Input file feature/field names and its sequence should be followed as per decided.

3.9 Tools used

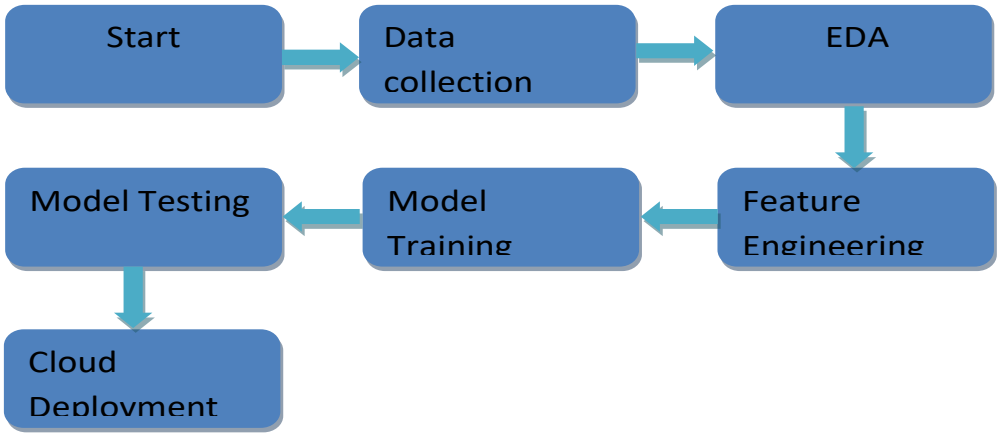
- **Python 3.8** – Programming language
- **NumPy** – NumPy is most commonly used package for scientific computing in Python.
- **Jupyter Notebook** – tool to do EDA
- **Scikit learn** – Machine learning library
- **Pandas** – Pandas is an open-source Python package that is widely used for data analysis and machine learning tasks.
- **Matplotlib, seaborn** – Data visualization
- **VS code** – IDE
- **Docker and apache airflow**- For continuous integration and deployment.
- **AWS** - Cloud deployment of application
- **MongoDB Atlas** – to retrieve, insert, delete and update the database
- **GitHub** – Source Code Version Control System
- **Flask** – Create and deploy webapp.

3.10 Constraints-nil

3.11 Assumptions-
The main objective of this project is to predict insurance premium based on input from user. We have used apache airflow for model monitoring, AWS cloud and Flask app for prediction so it should be accessible from every system which is connected to the internet.

4. Design Details

4.1 Application Process Flow



4.2 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.

4.3 Error Handling

If error encountered it should be displayed to user what went wrong.

4.4 Performance

The application is going to give user an estimation of premium, so it should be as accurate as possible otherwise it will lead to lose the business. Also model retraining is very important to improve performance.

4.5 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.

4.6 Application Compatibility

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

4.7 Deployment

The best model is deployed in AWS EC2 instance using docker and github actions.

5. 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. Accuracy is key metrics which plays a key role in prediction-based system. From the results we could see that Random Forest 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

6. Document Version Control

Date Issued	Version	Description	Author
14/11/2022	00	Initial Release-Draft created	Aniket Dumbre
18/11/2022	01	Modification in architecture	Aniket Dumbre