A PROJECT REPORT

ON

HEALTH INSURANCE CLAIM PREDICTION USING MACHINE LEARNING

Submitted in partial fulfilment of the requirements for the award of the degree

Bachelor of Technology

in

COMPUTER SCIENCE & ENGINEERING

Submitted by

N PADHMAVATHI	**20G01A0552**
A PAVANI	**20G01A0503**
MACHA SIVAKUMAR	**20G01A0542**
T BHARATH KUMAR	**20G01A0571**

Under the guidance

Mr. S.NATESAN, M. Tech (Ph.D)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SRI VENKATESA PERUMAL COLLEGE OF ENGINEERING &TECHNOLOGY

(AUTONOMOUS)

RVS Nagar, K N Road, Puttur, Chittoor(dist),517582

www.svpcet.org

(2023-2024)

SRI VENKATESA PERUMAL
COLLEGE OF ENGINEERING & TECHNOLOGY
| AUTONOMOUS | ACCREDITED BY NAAC |
RVS Nagar, K.N Road, Puttur, Chittoor dist. AP. I www.svpcet.org

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

****

This is to certify that the project report entitled **“HEALTH INSURANCE CLAIM PREDICTION USING MACHINE LEARNING**” is being submitted by the members of batch no:**CS20A10**

N PADHMAVATHI	**20G01A0552**
A PAVANI	**20G01A0503**
MACHA SIVAKUMAR	**20G01A0542**
T BHARATH KUMAR	**20G01A0571**

In partial fulfillment of the requirements for the degree Bachelor of Technology in Computer Science & Engineering award **from Sri Venkatesa Perumal College of Engineering & Technology, Puttur,** affiliated to Jawaharlal Nehru Technological University Anantapur, Ananthapuram. This is the Bonafede work carried out by them under my guidance and supervision during the academic year 2022-2023.

PROJECT GUIDE **HEAD OF THE DEPARMENT**

Mr.S.NATESAN,M.Tech **Dr.RAMAGANESH,M.Tech,Ph.D**

Submitted for the viva-voce examination held on ……

Internal Examiner **External Examiner**

DECLARATION BY PROJECT GUIDE

I hereby declare that the project report entitled “**HEALTH INSURANCE CLAIM PREDICTION USING MACHINE LEARINIG**” is the bonafide work carried out by the members of Batch no.CS20A10 of Sri Venkatesa Perumal College of Engineering & Technology, Puttur for the award of degree Bachelor of Technology in Computer Science & Engineering during the academic year 2022-2023 is original work and the project has not formed the basis for the award of any degree, diploma, associate fellowship or any other similar title submitted previously.

PROJECT GUIDE

Mr. S.NATESAN ,M.TECH,(PhD)

DECLARATION BY PROJECT MEMBERS

We hereby combinedly declare that the project entitled **“HEALTH INSURANCE CLAIM PREDICTION USING MACHINE LEARNING”** submitted by **Batch no. CS20A10** for the award of our degree in B. Tech Computer Science & Engineering in our original work and the project has not formed the basis for the award of any degree, diploma, associate fellowship or any other similar title submitted previously.

DECLARATION:

N PADHMAVATHI	A PAVANI
(20G01A0552)	(20G01A0503)
MACHA SIVAKUMAR	T BHARATH KUMAR
(20G01A0542)	(20G01A0571)

Place:

Date :

ACKNOWLEDGEMENT

The satisfaction and euphoria accompany the successful completion of task and would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crown all the efforts with success.

We wish to express our deepest sense of gratitude and pay our sincere thanks to our project guide **Mr. S.NATESAN**, **M. Tech,(PhD) ,** Department of CSE, who keen interest in our efforts and provided his valuable guidance throughout our project work.

We also express our sincere gratitude to **Dr. B. RAMA GANESH, M. Tech, Ph. D,** Head of the department of CSE for his great encouragement and valuable support throughout our study.

We owe our gratitude to our principal **Dr.T.SUNIL KUMAR REDDY,M.Tech,Ph.D**, for his kind attention and valuable guidance given to us throughout this course.

We sincerely and whole heartedly thank to our beloved **Sri. RAVURI V. BALAJI,** Vice-Chairman for giving art of infrastructure facilities to us throughout our course study and leading to successful completion of our project.

We are very much thankful to our beloved **Dr. R.VENKATA SWAMY** Chairman of Sri Venkatesa Perumal College of Engineering & Technology, Puttur for his kind attention and valuable guidance to us throughout the course.

We also thankful to all staff members of CSE Department for helping me to complete this project work by giving valuable suggestions.

We would like to thank the member so four families who assisted in the preparation of this report financially.

The last but not least we express our sincere thanks to all our friends who have supported us in the accomplishment of this project.

ABSTARCT

The health care costs constitute a significant fraction of the U.S. economy. Nearly 20% ofthe Gross Domestic Product (GDP) is spent on health care. The health spending in theUS is the highest among all developed nations in absolute numbers as well as apercentage of the economy. In this work, we will develop a medical price prediction system using machine learning algorithms which will aid in steering patients to cost effective providersand thereby curb health spending. The policymakers can also use the tool to better understand which providers are relatively expensive and take punitive actions if necessary.The prediction of the medical price will be done using implementing Random Forest Regression algorithm in machine learning. Additionally, we plan to include the experiments on the same data with other machine learning models such as Random Forest Regression, Decision Tree Regression and Linear Regression and compare results. The findings from these experiments will also be included.

Vision and mission of the college

Vision:

To emerge as a Center of Excellence for Learning and Research in the domains of Engineering, Technology, Computing and Management

Mission:

Mission	Mission statements
M1	To provide congenial academic ambience with state-of-art resources for learning.
M2	Ignite the students to acquire self-reliance in the latest technologies.
M3	Unleash and encourage the innate potential and creativity of students.
M4	Foster enterprising spirit among students work collaboratively.
M5	

Vision and Mission of the Department

Vision:

To contribute for the society through excellence in Computer Science and Engineering with a deep passion for wisdom, culture and values.

Mission:

M1	Mission Statements
M2	Provide congential academic ambidience with necessary new challenges from industry.
M3	Inculcate confidence to face and experience new challenges from industry.
	Ingnite the students to acquire self-reliance on latest technologies.

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PS01	PS02
CO1	✓												✓	✓
CO2		✓											✓	✓
CO3			✓										✓	✓
CO4				✓									✓	✓
CO5					✓								✓	✓
CO6						✓							✓	✓
CO7							✓						✓	✓
CO8								✓					✓	✓
CO9									✓				✓	✓
CO10										✓			✓	✓
CO11											✓		✓	✓
CO12												✓	✓	✓

TABLE OF CONTENT

Chapter Name. **Page No.**

LIST OF FIGURES

LIST OF TABLES

CHAPTER-01

INTRODUCTION

CHAPTER-01

INTRODUCTION

1.1 Introduction

The capacity to simplify datasets is a fundamental advantage of employing machine learning (ML) in the insurance sector. Machine learning (ML) is effective with organized, semi-structured, and unstructured datasets. Machine learning has several applications in insurance, ranging from perceived risk-taking and premium loss to expenditure management,regression, processes, and fraud detection. Learning is classified into three types: supervisedlearning, unsupervised learning, and reinforcement learning. Following decades of supervisedlearning, the majority of insurers estimate risk and accomplish desired results by combining known parameters. If variables change, the technique detects them and seeks to alter them inaccordance with the purpose. Reinforcement learning is mostly based on ANN (Artificial Neural Network), which may adjust the target/goal dynamically depending on the purpose.

We live on a planet full of threats and uncertainty. Including People, households, durable, properties are exposed to different risks and the risk levels can vary. These risks range from risk of health diseases to death if not get protection, and loss in property or assets. But risks cannot usually be avoided. Therefore, health insurance is a policy that covers or minimizes the expenses of losses caused by a variety of hazards. Insurance is financial protection against any type of risk. Financial challenges can be avoided if a person has a healthinsurance policy at the time of medical treatment. Apart from this, many people are not so wise while choosing the health insurance plans. So, people can be fooled easily about the amount of the insurance and may unnecessarily buy some expensive health insurance. A lossmay occur when a client purchases a plan which may be claimed a less amount than their eligibility.

Various parameters or factors play an important role in estimating the insurance charges and each of these is important. If any factor is omitted or changed when the amountsare computed then, the overall policy cost changes. It is therefore very critical to carry out these tasks with high accuracy. So, the possibility of human mistakes is high so insurance agents also use different tools to calculate the insurance premium. And thus , ML is beneficialhere. ML may generalize the effort or method to formulate the policy. The model is trained on insurance data from the past. The model can then accurately predict insurance policy costs by using the necessary elements. This decreases human effort and resources and improves thecompany's profitability. Thus, the accuracy can be improved with ML. It helps in predicting the amount that can be claimed by an individual based on their basic information asparameters provided such as BMI, Number of children, smoking habit etc.

People's healthcare cost forecasting is now a valuable tool for improving healthcare accountability. The healthcare sector produces a very large amount of data related to patients,diseases, and diagnosis, but since it has not been analyzed properly, it does not provide the significance which it holds along with the patient healthcare cost.

1.2 Existing System

Now-a-days, most of the people are taking health insurance for their future purpose, but most of them don't know how to calculate the insurance amount. Because various factorsinfluence the claim charges of insurance, in which insurance claim charge calculation becomes a very difficult task for the user to do. So, insurers generally use traditionally methodto estimate their claims by grouping similar claims and exposures together and analyzing historical loss development patterns across the different groups. This method does not accurately predict individual claim behaviour.

Manual systems for health insurance claim prediction involve human experts, such as physicians and insurance claims adjusters, reviewing and analyzing data to make predictionsabout the likelihood of a claim being approved or denied. One disadvantage of manual systems is that they can be time-consuming and expensive. It can take a significant amount oftime for experts to review and analyse data, and the cost of employing these experts can be high. Additionally, there is a risk of human error, which can lead to inaccurate predictions and potentially costly mistakes. Another disadvantage is that manual systems may not be scalable. As the volume of claims increases, it becomes more difficult for experts to review each claim individually, and it may be necessary to automate some aspects of the process. Manual systems may also be biased, as experts may have different opinions or approaches toanalysing data. This can result in inconsistencies in the prediction process and potentially unfair treatment of claimants. Finally, manual systems may not be able to effectively handle complex or unstructured data. For example, if a claim involves multiple diagnoses ortreatment plans, it may be difficult for experts to accurately predict the outcome. Overall, while manual systems can provide some level of accuracy in predicting health insurance claims, they have several disadvantages compared to automated systems. These include higher costs, potential for bias and inconsistency, and limitations in handling large amounts of complex data.

There are several existing systems for prediction for health insurance like Statistical models such as linear regression and logistic regression, are relatively simple and easy to interpret. However, they may not capture complex patterns in the data or handle large amountsof data efficiently.

1.3 Disadvantages of Existing System

- It is a time-consuming process.
- Traditional methods won't provide accurate results.
- Very difficult to understand the inner calculations for an user.
- Manually, it is difficult to calculate the claim charges accurately.
- Also, there is a risk of human error, which can lead to inaccurate predictions and potentially costly mistakes.
- Existing statistical models may not handle large amounts of data or capture patternsin the data efficiently.
- Manual systems include higher costs, potential for bias and inconsistency and limitations in handling large amounts of complex data.
- Manual systems may not be able to handle complex or unstructured data effectively.
- Manual systems may not be scalable.
- Because of the payment errors made by the insurance companies, processing the claims can lead to different losses including financial loss and administrative costs etc.

1.4 Proposed System

For an insurer, it is very difficult to calculate the health insurance value to claim accurately. The task of the proposed system is to calculate the medical insurance accurately that can be claimed by an individual based on their basic information as parameters providedsuch as age, BMI, Number of children, smoking habit etc. So, the proposed system InsuranceClaim Prediction can be implemented by using machine learning models. In this system, insurance data based on some features are trained and tested over Random Forest Regressor (RFR) and predict the charges based on features. The whole industry runs on the concept of risk or financial loss reduction. So, this proposed system provides accurate predictions by using historical data gives a probability to decrease financial loss for the company and users.

A computational intelligence approach is applied to predict health insurance. By using machine learning models for the insurance prediction, the human mistakes can be avoided. Also, manual efforts can be reduced and the medical insurance claim prediction can be done in a short time with accurate results. With the prediction of high certainty, problems such as accountability could be solved, enabling control over all parties. It could also be used for otherapplications such as risk assessment, or for the application of new policies by governments toimprove public health.

It avoids the payment errors so it can result in reducing the losses like financial loss,administrative costs, etc.It can be helpful to solve accountability problems**.**

1.5 Advantages of Proposed System

- It reduces manual efforts.
- It saves the time by completing the task in a very short time.
- It can be able to handle complex data effectively.
- It can avoid human mistakes and minimize individual efforts.

1.6 Literature Survey

In 2018, Muhammad rFauzan during this paper, the truth of XGBoost is applied to predict statements. Compare the output with the performance of XGBoost, a group of techniques e.g., AdaBoost, Random Forest, Neural Network. XGBoost offers higher Gini structured accuracy. Mistreatment publically accessible urban center Seguro to Kaggle datasets. The dataset includes vast quantities of NaN values however this paper manages missing values by medium and median replacement. However, these simple, unprincipled strategies have additionally proved to be biased. They, therefore, target exploring the cubic centimeter methods that are extremely applicable for the issues of many missing values, suchas XGboost.

Belhadji, E., G. Dionne and F. Tarkhani: The aim of this article is to develop a modelto aid insurance companies in their decision-making and to ensure that they are better equipped to fight fraud. This tool is based on the systematic use of fraud indicators. We firstpropose a procedure to isolate the indicators which are most significant in predicting the probability that a claim may be fraudulent. We applied the procedure to data collected in theDionne–Belhadji study (1996). The model allowed us to observe that 23 of the 54 indicatorsused were significant in predicting the probability of fraud. Our study also discusses the model's accuracy and detection

G. Kowshalya, M. Nandhini. in 2018 classifiers are developed during this study to predict and estimate dishonorable claims and a proportion of premiums for the varied customers based mostly upon their personal and monetary data. For classification, the algorithms Random Forest, J48, and Naïve Bayes are chosen. The findings show that RandomForest exceeds the remaining techniques betting on the artificial dataset. This paper thus doesn't cowl claim forecasts, however rather focuses on false claims. The on top of previousworks failed to contemplate each foreseen the value or claim severity, they solely create a classification for the issues of claims (whether or not a claim was filed for that policyholder)during this study An article by Nidhi Bhardwaj and Rishabh Anand used individuals' health data to forecast their insurance premiums. To assess and evaluate the performance of variousalgorithms, regression was utilised. The dataset was used to train the models, and the results of that training were utilised to make predictions. The model was then tested and verified bycomparing the anticipated quantity to the actual data. The accuracy of these models was latercompared. Multiple linear regression and gradient boosting algorithms outperformed linear regression and decision trees, according to the findings. Gradient boosting was suitable in thisscenario since it required far less computing time to attain the same performance measure, although its performance was equivalent to that of multiple

CHAPTER-02

ANALYSIS

CHAPTER-02

ANALYSIS

2.1 Introduction

Data mining (DM) and machine learning (ML) techniques are widely used for insurance cost prediction and medical fraud detection. Hierarchical Decision Trees and other ML models are used for predictive analytics of healthcare costs. They also suggested that machine learning tools and techniques are critical in the healthcare sector and that they are exclusively used in the diagnosis and prediction of medical insurance costs. Similarly, the underwriting process and medical investigations necessary by the insurance firm to profile the applicants' risks can be difficult and costly.

Statistics has been in use in the insurance industry from the onset of the industry. Thereis a whole discipline for the use of statistics in the insurance industry known as Actuarial Science. With the massive increase in data processed by the industry, Predictive Analytics iscoming into the limelight. It encompasses data mining, predictive modelling, and machine learning techniques like classification, regression, clustering and outlier detection to make accurate and fast predictions about unforeseen events in the future using the current data.

Claim Analysis is an important aspect of Predictive Analytics in the insurance industryas approximately 80% of the premium revenue generated is spent on claims by the insurancecompanies. Hence, it is essential to do a thorough analysis of claims to improve cash flow. By analysing the insurance data, relations between various factors (variables) are observed and a function is derived to model predictions. These predictions can be used for making decisions. Apart from the structured data used by the companies, there is a huge scope of unstructured data that provides vital information.

2.2 System Requirements Specification:

2.2.1 User Requirements

User requirements refer to the specific needs and expectations of users or stakeholdersfor a product, system, or service. These requirements are typically gathered and documentedduring the early stages of a project and serve as the foundation for designing and developinga solution that meets the needs of the intended users. User requirements are critical for guidingthe development process and ensuring that the resulting solution aligns with the users' needs and expectations.

The user requirements of this project are as follows:

- PC, Mac or laptop with x86-64 (64-bit) compatible processors.
- GHz or better processor is recommended.
- Internet connection required for managing the application.
- Microsoft Windows specific requirements:
- Microsoft Windows 7 / 8 / 10 / 11/ 12.

2.2.2 Software Requirements

- **Python IDE**

It includes different software platforms like PyCharm, Anaconda, VisualStudio, Python IDLE, PyDev, Thonny, Spyder, etc for the development of the application. Here Visual Studio was used. Visual Studio is an Integrated DevelopmentEnvironment (IDE) that is used to develop GUI (Graphical User Interface), console, Web applications.

- **Jupyter Notebook**

Jupyter notebooks basically provides an interactive computational environment for developing Python based Data Science applications. They are formerly known as ipython notebooks. Jupyter notebooks can illustrate the analysis process step by step by arranging the stuff like code, images, text, output etc. in a step-by-step manner.

- **Spreadsheet**

An Excel spreadsheet is a computer application that is designed to add, display, analyse, organise, and manipulate data arranged in rows and columns. It is the most popular application for accounting, analytics, data presentation, etc. Here, excel spreadsheet was used to store the dataset which is required to train the machine learning model.

2.2.3 Hardware Components

- Processor – i5 or above
- Memory – 2GB RAM

Software Requirements	**Hardware Requirements**
➢ Operating System - windows 7 or ab	➢ Processor - i3 or above
➢ Backend - Python	➢ RAM - 4GB
➢ Frontend - HTML, CSS	➢ CPU - 1.6GHz
➢ Environment - Jupyter Notebook	➢ Hard Disk – 40GB
➢ Spreadsheet - Excel	

2.3 Flowchart

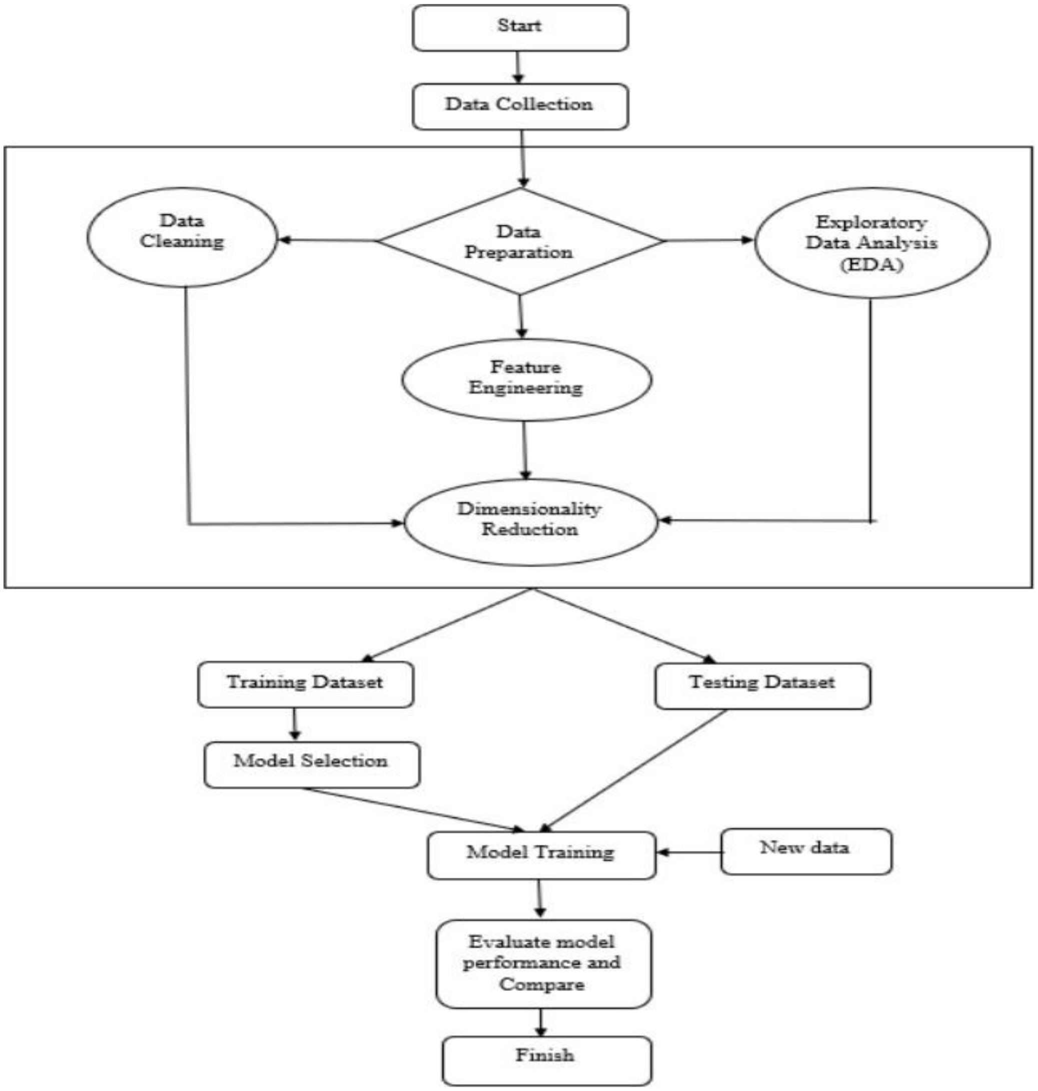


Fig 2.3: ML framework for working methodology

2.3.1. Data Collection

There are various methods and sources for collecting data. One of the most common and effective ways is to search for and share data. Data can be searched on the internet or through other available resources. Another approach is Data Augmentation, which involves augmenting existing data with external data to increase dataset diversity rather than collectingnew data. This technique is frequently used in deep learning to train large artificial neural networks (ANN). Crowdsourcing and generating synthetic datasets are additional methods for data collection. In this project, the datasets used in the analysis were obtained from Kaggle.com.

2.3.2 Data Preparation

Data Preparation is a crucial step in Machine Learning that involves transforming the data into a suitable format for use by an ML algorithm. It involves various processes such asdata cleaning, exploratory data analysis (EDA), normalization, and dimensionality reduction,which can significantly impact the performance of the model. Proper data preparation ensuresthat the data is free from errors, inconsistencies, and outliers, making it more reliable and accurate for analysis. EDA helps to gain insights into the data and identify patterns, relationships, and trends that can inform feature engineering and selection. Normalization and dimensionality reduction techniques such as scaling, feature extraction, and feature selectionhelp to reduce the complexity of the data, making it more efficient for processing by the MLalgorithm.

2.3.3 Data Cleaning

Data cleaning is an essential step in data pre-processing that involves detecting and handling inaccurate, false, incomplete, corrupt, or irrelevant data records. It is crucial in ensuring the quality and usability of the dataset for machine learning algorithms. One of the commonly used approaches in data cleaning is variable-by-variable cleaning, where erroneousvalues are identified and removed based on certain criteria such as permissible value range, variance, and standard deviation. In cases where there are missing feature values, the featureis either eliminated if there are many missing values or imputed with a dummy value. Mean substitution is the most common approach for treating missing values.

2.3.4 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a crucial step in the data analysis process that helps in understanding the data before applying any machine learning model to it. EDA involves the use of different graphical and statistical methods to analyse and summarize data, identify patterns and relationships between different features, and detect any anomalies or outliers in the dataset. EDA is usually done by visualizing data with the help of different graphs, charts, and other visualization tools, which allows data scientists to identify the different characteristics of the data, including hidden relationships among various features.

2.3.5 Feature Engineering

Feature engineering is a critical step in the data preparation process before moving onto model training and evaluation. It involves creating new features based on the insights gained from EDA and domain knowledge to enhance the model's performance. Feature engineering is often a challenging and time-consuming process that requires creativity and expertise. The new features are typically generated by applying various calculations and operations to the existing features to extract more meaningful information.

The new features might be a ratio, or a mathematical transformation or any statistical or scientific formula to generate a more significant feature. Feature engineering can be done both manually by statisticians and by using feature encoding techniques in the case of categorical variables. Feature engineering has proved to be greatly beneficial for random forests, neural networks, and gradient boosting machines. Encoding is important as machinelearning itself is based on mathematical models and algorithms, so most of the algorithms cannot classify between categorical and continuous values. Encoding follows two methodologies: nominal and ordinal. Nominal Encoding is performed where the order of thedata is not of much importance and vice-versa.

Apart from encoding, there are other techniques for feature engineering such as normalization. Normalization is used for scaling all the values in a dataset in a fixed range between 0 and 1. The formula used for normalization is

$$[Xnorm = (X - Xmin) / (Xmax - Xmin)]$$

2.3.6 Dimensionality Reduction

Dimensionality Reduction refers to the process of reducing the number of features in a datasets while preserving the important information. Feature selection and feature extractionare the two approaches for dimensionality reduction. In this analysis, only feature selection isused to reduce the dimensionality of the feature set. Feature extraction is more suitable for data used for pattern recognition or image processing, where meaningful inferences cannot beobtained just by looking at the data.

2.3.7 Feature Selection

Considering all features for modelling can potentially decrease the predictability of the model. Therefore, it is preferred to select the features that contribute more to the target variable. There are several methods for feature selection, such as manual methods like univariate selection, where each feature is evaluated to determine its importance. For univariate analysis, statistical methods such as variance and Pearson correlation are used. However, univariate analysis is more reliable for linear data, and it is difficult to perform on large datasets. In such cases, multivariate analysis can be performed. The three methods of multivariate analysis are filter, wrapper, and embedded. The feature selection methods are Chi-Square test, Recursive Feature Elimination (RFE), and Tree-based feature selection.

Chi-Square test:It is a type of statistical filter method that is used to evaluate the correlation between different features using their frequency distribution. In this method, feature selection is basedon the intrinsic properties of the features and is independent of any ML algorithm.

Recursive feature elimination (RFE):

It is a type of wrapper method used for feature selection. The term “wrapper” is usedbecause this method wraps up a classifier in a feature selection algorithm. In RFE, features are recursively removed from the dataset based on an external estimator used which is the classifier. The classifier assigns weights to a feature based on its performance. It is a greedy algorithm that seeks to generate the best performing subset.

2.4 Tree-based feature selection:

It is a type of embedded method in which there is an inbuilt method for feature importance which generates a set of features along with their importance.

2.4.1 Training Dataset and Testing Dataset

Once the data preparation phase is complete and the dataset has been divided into training dataset and testing dataset. Roughly 80% of the total data has been allocated for training purposes, while the remaining 20% is reserved for testing. The objective of using thetraining dataset is to build a regression model that can accurately predict medical insurance costs for a given year. On the other hand, the test dataset is utilized to assess the performanceand efficacy of the developed model.

2.4.2 Model Selection

Once the data preparation phase is complete and the dataset has been divided into training and testing sets, it is essential to select appropriate models for the training process. As the problem at hand involves classification, the selection of suitable classifiers is necessary. It is worth noting that both the datasets used in this analysis belong to the binary classification category.

2.4.3 Model Training

After the selection of appropriate models, the training dataset is utilized to train the models, initially considering all available features. However, subsequently, feature selection techniques are employed to determine which features are most relevant. The classification algorithms are then executed solely on the selected features.

2.4.4 Evaluate model performance and compare

Ultimately, a comparative analysis is conducted to evaluate the performance of the various models, in combination with the feature selection techniques, to determine the most effective combination for both datasets. This allows for the identification of the best model and feature selection technique for each dataset.

2.5 Algorithm

Regression is a statistical analysis technique that helps to understand the relationship between independent variables or features and a dependent variable or outcome. It is commonly used as a method for predictive modeling in machine learning, where an algorithmis employed to predict continuous outcomes. There are several techniques available for performing statistical predictions, but here Random Forest Classifier was used

2.5.1 Random Forest Regression

Random Forest is a popular machine learning algorithm that belongs to the supervisedlearning technique. It can be used for both Classification and Regression problems in ML. Itis based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random foresttakes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

The below diagram explains the working of the Random Forest algorithm:

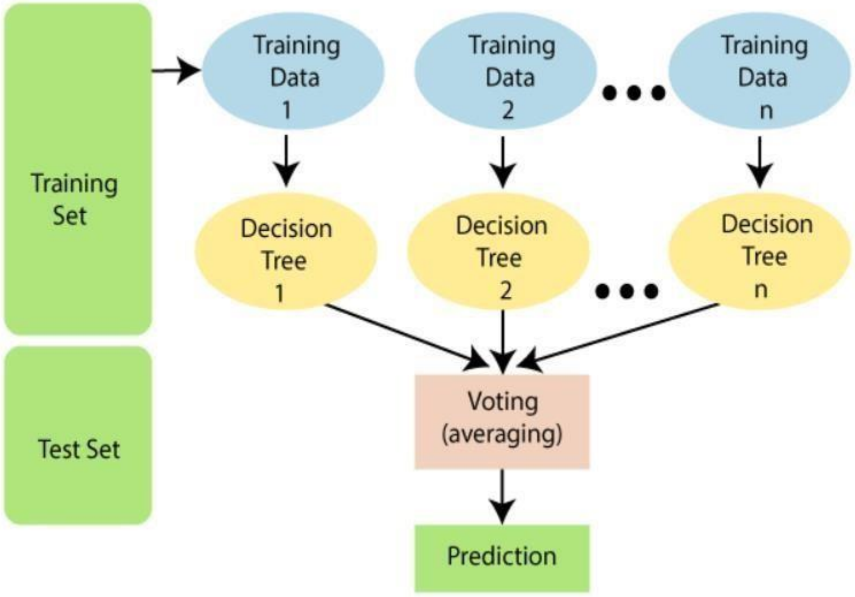


Fig 2.5.1: ML framework for working methodology

2.5.1.2How does Random Forest algorithm work?

Random Forest works in two-phase first is to create the random forest by combiningN decision tree, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the newdata points to the category that wins the majority votes.

2.5.2 Applications of Random Forest

There are mainly four sectors where Random Forest mostly used:

A. Banking: Banking sector mostly uses this algorithm for the identification of loan risk.

B. Medicine: With the help of this algorithm, disease trends and risks of the disease can beidentified.

C.Land Use: We can identify the areas of similar land use by this algorithm.

D. Marketing: Marketing trends can be identified using this algorithm.

2.5.3 Advantages of Random Forest

- Random Forest is capable of performing both Classification and Regression tasks.
- It is capable of handling large datasets with high dimensionality.
- It enhances the accuracy of the model and prevents the overfitting issue

CHAPTER-03

SYSTEM IMPLEMENTATION

CHAPTER-03

SYSTEM IMPLEMENTATION

3.1 Introduction

The implementation of health insurance claim prediction using machine learning has the potential to revolutionize the healthcare industry. By leveraging the power of machine learning algorithms. Health care providers and insurance companies can predict he likely hood of a claim being approved or denied with a high degree of accuracy. This allows them to make more informed decisions, improve patient care, and reduce costs.

After the machine learning model has been trained, it is important to evaluate its performance on a separate portion of the historical data using metrics such as mean squared error or R-squared. This allows healthcare providers and insurance companies to determine the accuracy of the mode land make any necessary adjustments.

Finally, the trained machine learning model can be deployed as a predictive system for insurance claims. This allows healthcare providers and insurance companies to input patient data and receive apredicted out come for the claim in real - time. Overall, the implementation of health insurance claim prediction using machine lear ning has the potential to transform the healthcare industry by improving patient care.

In order to build a health insurance claim prediction system using machine learning, a design that incorporates both frontend and backend components is necessary. For the frontend design, we have used HTML, CSS to create a user-friendly interface that allows healthcare providers and insurance companies to input patient data and receive a predicted outcome for the claim. The frontend design should be intuitive and easy to navigate, with clear instructions and feedback for the user. The design should also be responsive, allowing it to be used on a variety of devices, including desktop computers, laptops, etc.

3.2 Block Diagram:

A block diagram is a visual representation of a system or process that shows the major components or stages of the system and how they are connected or interact with each other

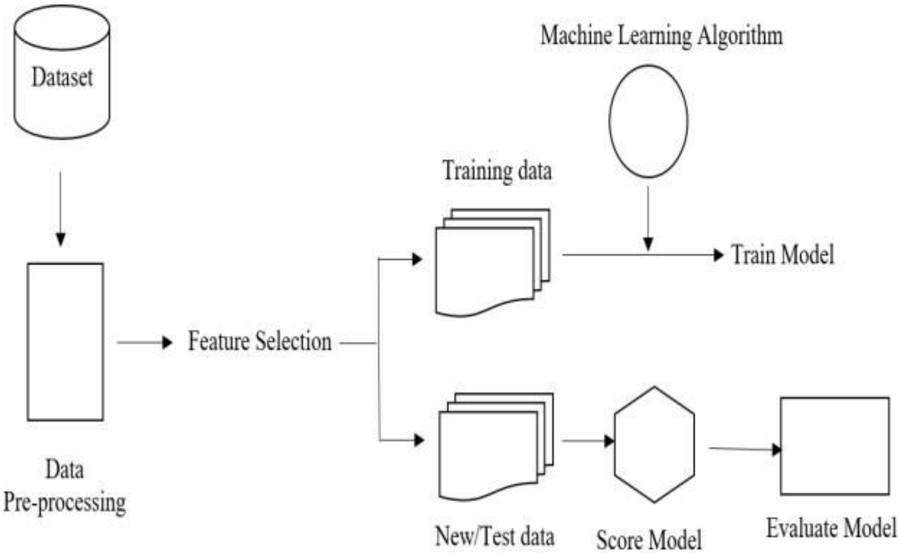


Fig 3.1: *Block Diagram*

3.3 DFD/UML Diagrams:

3.3.1 DFD Diagram

DFD is the abbreviation for Data Flow Diagram. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present. Specific operations depending on the type of data can be explained by a flowchart. It is a graphical representation of how the data flows through a system. It is useful for analyzing existing as well as proposed system. It provides an overview of what data is system processes, what transformations are performed, what data are stored, what results are produced, etc. Data Flow Diagram can be represented in several ways. The DFD belongs to structured-analysis 18odelling tools. Data Flow diagrams are very popular because they help us to visualize the major steps and data involved in software-system processes. DFD uses hierarchy to maintain transparency thus multilevel DFD's can be created.

3.3.2 Level DFD: level DFD goes one step deeper into parts of 1-level DFD. It can be used to plan or record the specific/necessary detail about the system's functioning

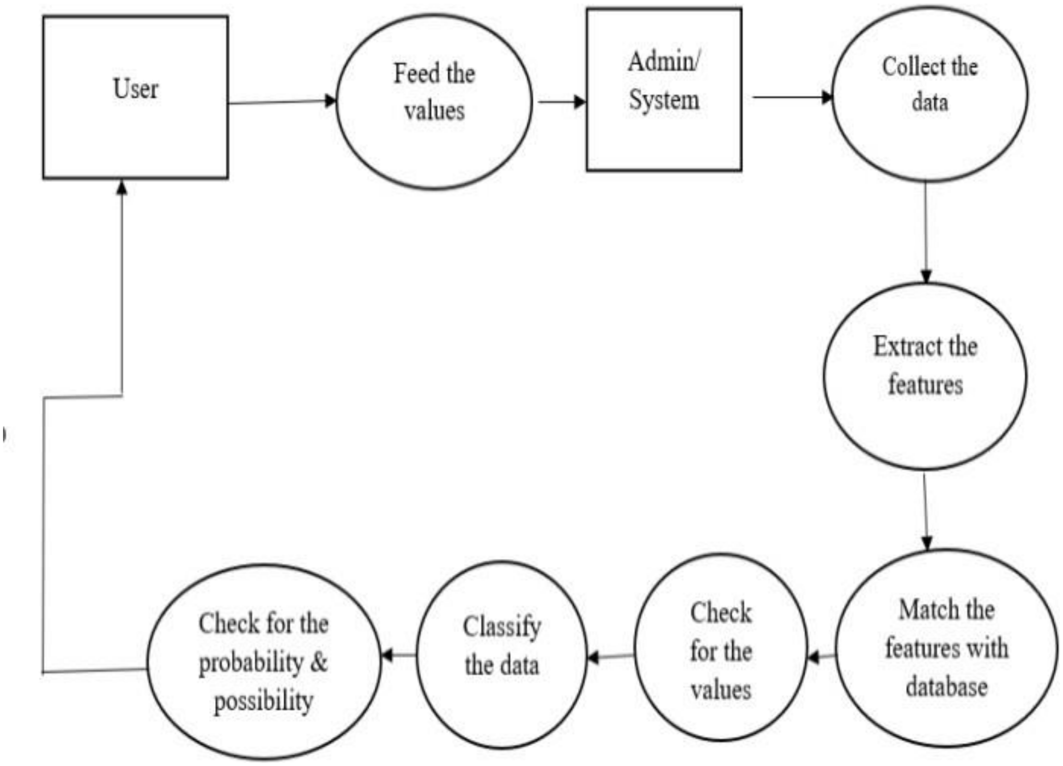


Fig 3.3.2 Level DFD

3.4 UM L Diagrams

Unified Modeling Language (UML) is a general-purpose modelling language. The main aim of UML is to define a standard way to visualize the way a system has been designed. It is quite similar to blueprints used in other fields of engineering. UML is not a programming language; it is rather a visual language. We use UML diagrams to portray the behaviour and structure of a system. UML helps software engineers, businessmen and system architects with modelling, design and analysis. The Object Management Group (OMG) adopted Unified Modelling Language as a standard in 1997. Its been managed by OMG ever since. International Organization for Standardization (ISO) published UML as an approved standard in 2005. UML has been revised over the years and is reviewed periodically.

3.4.1 Use case diagram:

Use case diagrams are a set of use cases, actors, and their relationships. They represent the use case view of a system. A use case represents a particular functionality of a system. Hence, use case diagram is used to describe the relationships among the functionalities and their internal/external controllers. These controllers are known as actors.

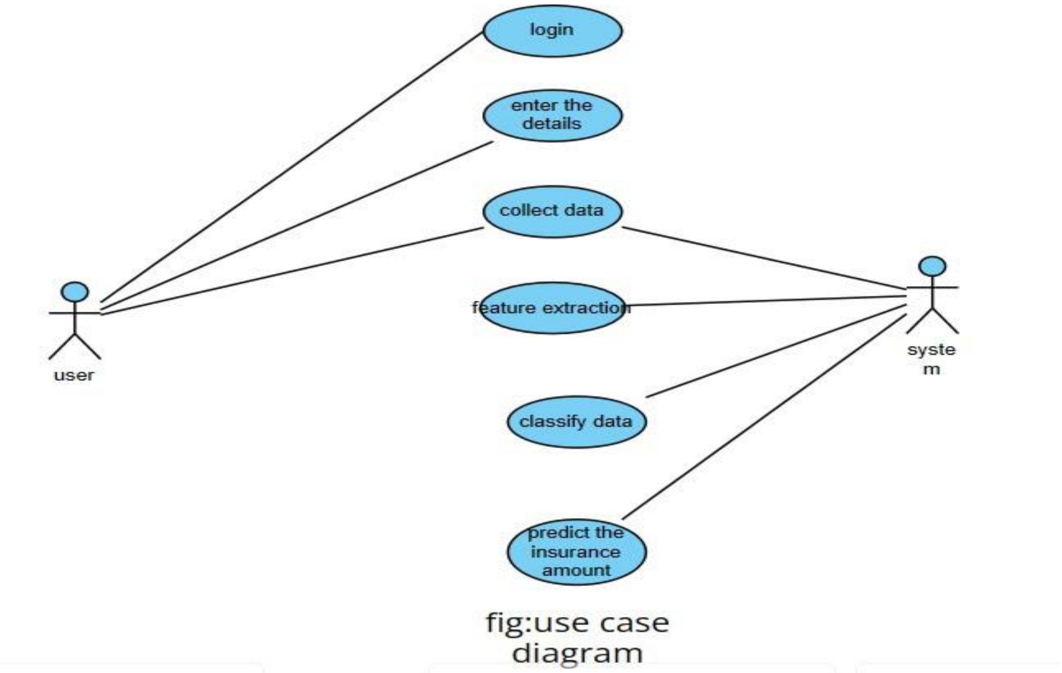


Fig 3.4.1 : Use Case diagram

3.4.2 Class Diagram:

Class diagrams are the most common diagrams used in UML. Class diagram consists of classes, interfaces, associations, and collaboration. Class diagrams basically represent the object-oriented view of a system, which is static in nature. Active class is used in a class diagram to represent the concurrency of the system. Class diagram represents the object orientation of a system. Hence, it is generally used for development purpose. This is the most widely used diagram at the time of system construction.

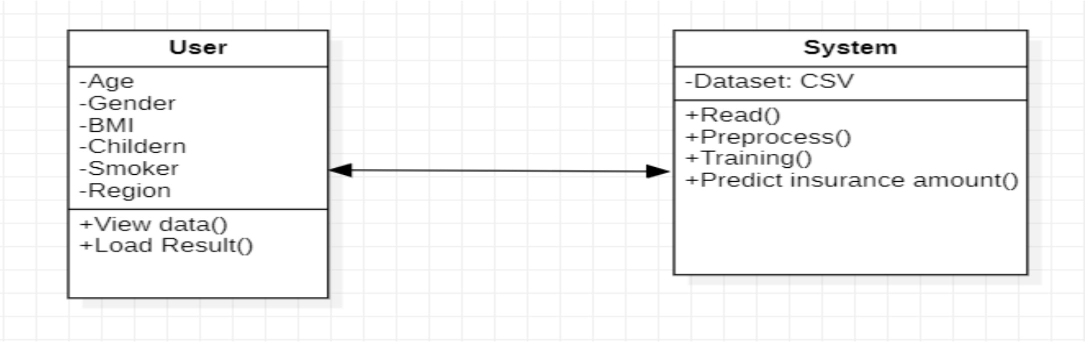


Fig 3.4.2 : Class Diagram

3.4.3 Sequence Diagram:

A sequence diagram is an interaction diagram. From the name, it is clear that the diagram deals with some sequences, which are the sequence of messages flowing from one object to another. Interaction among the components of a system is very important from implementation and execution perspective. Sequence diagram is used to visualize the sequence of calls in a system to perform a specific functionality.

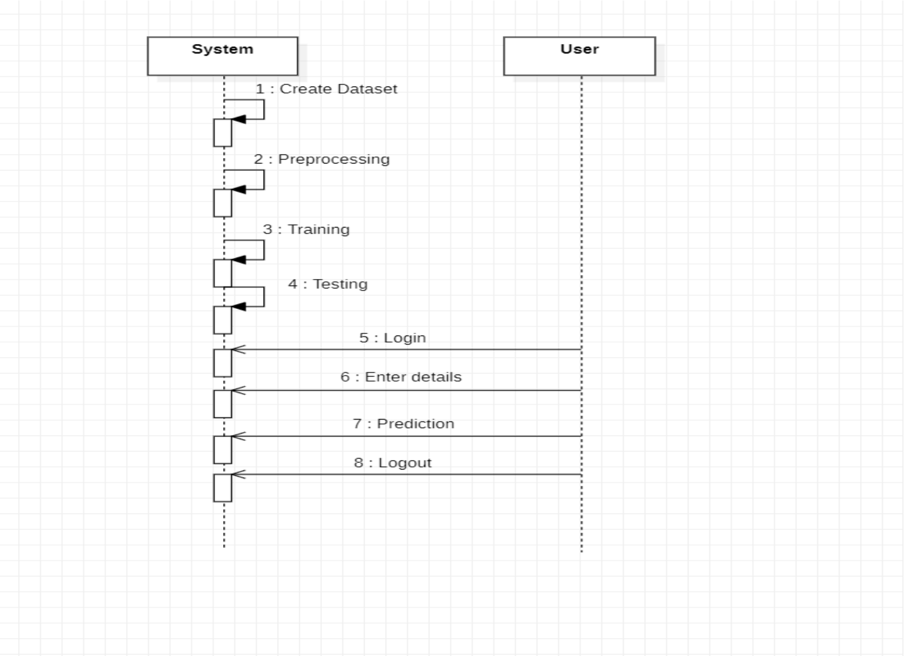


Fig 3.4.3 : Sequence diagram

3.4.4 Activity Diagram:

Activity diagram describes the flow of control in a system. It consists of activities and links. The flow can be sequential, concurrent, or branched. Activities are nothing but the functions of a system. Numbers of activity diagrams are prepared to capture the entire flow in a system. Activity diagrams are used to visualize the flow of controls in a system. This is prepared to have an idea of how the system will work when executed.

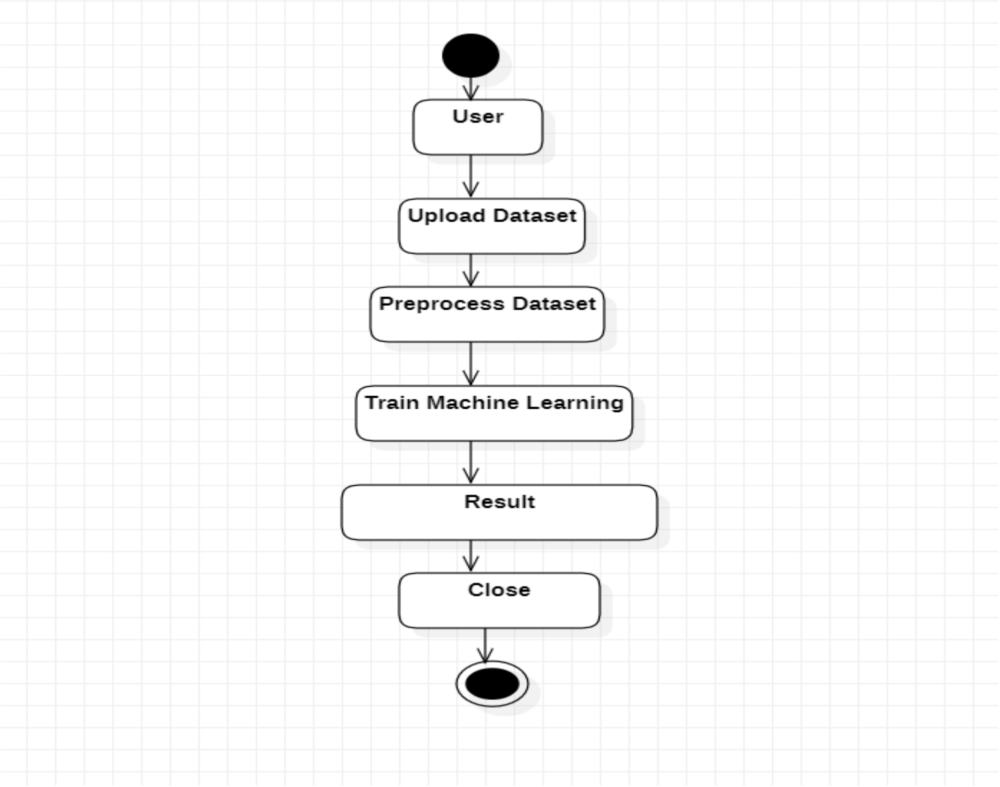


Fig 3.4.4 : Activity diagram

3.4.5 component diagram

Component diagrams represent a set of components and their relationships. These components consist of classes, interfaces, or collaborations. Component diagrams represent the implementation view of a system. During the design phase, software artifacts (classes, interfaces, etc.) of a system are arranged in different groups depending upon their relationship. Now, these groups are known as components. Finally, it can be said component diagrams are used to visualize the implementation.

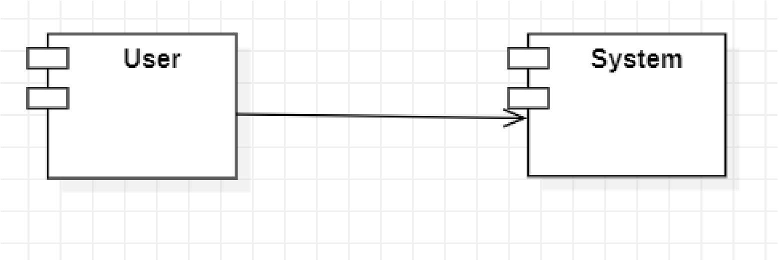


Fig3.4.5 : Component diagram

3.4.6 Deployment diagram:

Deployment diagrams are a set of nodes and their relationships. These nodes are physical entities where the components are deployed. Deployment diagrams are used for visualizing the deployment view of a system. This is generally used by the deployment team.

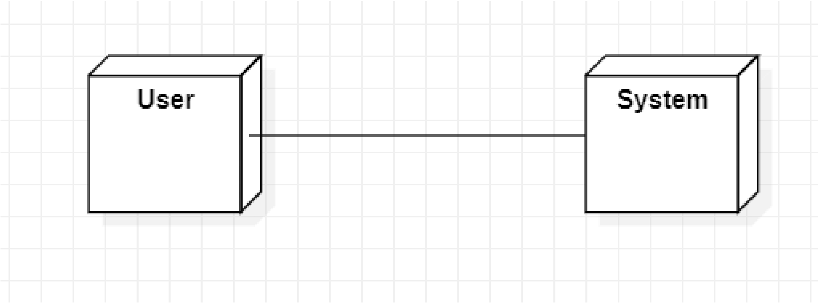


Fig 3.4.6 : Deployment diagram

3.5 Module design & Organization

In a health insurance claim prediction system using machine learning, there are several web modules that may be used to facilitate the prediction process. These modules include:

3.5.1 User interface

This module is responsible for providing a user-friendly interface that allows healthcare providers and insurance companies to input patient data and receive a predicted outcome for the claim. The user interface may be developed using HTML, CSS and should be responsive and intuitive.

Mainly there are two modules. They are

- System module
- User module

3.5.1.1 System module:

The process which will be done under system module is as follows:

- ➢ The process get started.
- ➢ The first step is taking the dataset and pre-processing the dataset.
- ➢ Then splitting the dataset into two datasets i.e., training dataset which is 80% andtesting dataset which is 20%.
- ➢ Next train the machine learning model by using the training dataset and test the model by using testing dataset.
- ➢ Then evaluate the model and observe the accuracy and performance of the model. The process got completed.

3.5.1.2 User module:

- ➢ The process get started.
- ➢ The user has to login if already account exists otherwise the user has to register.
- ➢ Then the user has to enter the details as parameters for prediction.
- ➢ Then the system will gives the medical insurance amount that the person can claim Up to.
- ➢ The process got ended.

3.5.2 Data preprocessing:

This module is responsible for cleaning and pre processing the data, including removing missing values, converting categorical data to numerical data, and scaling the data. This module may be developed using Python and data processing libraries such as Pandas and NumPy.

3.5.3 Feature engineering:

This module is responsible for identifying the most relevant features for predicting insurance claim outcomes. This could include features such as patient age, type of treatment, and the presence of commodities. This module may also be developed using Python and data processing libraries.

3.5.4 Machine learning model:

This module is responsible for selecting and training the machine learning algorithm for predicting insurance claim outcomes. This module may be developed using Python and machine learning libraries such as Scikit-learn and TensorFlow.

3.5.5 Model evaluation:

This module is responsible for evaluating the performance of the machine learning model on a separate portion of the historical data, using metrics such as mean squared error or R-squared. This module may also be developed using Python and data processing and machine learning libraries.

3.5.6 Deployment:

This module is responsible for deploying the trained machine learning model as a predictive system for insurance claims. This module may be developed using Python and web frameworks such as Django or Flask. Overall, these web modules work together to create a health insurance claim prediction system that is accurate, efficient, and user-friendly.

CHAPTER-04

SYSTEM DEPLOYMENT AND TESTING

CHAPTER-04

SYSTEM DEPLOYMENT AND TESTING

4.1 TESTING:

Software testing can be stated as the process of verifying and validating whether a software or application is bug-free, meets the technical requirements as guided by its design and development, and meets the user requirements effectively and efficiently by handling all the exceptional and boundary cases.

The process of software testing aims not only at finding faults in the existing software but alsoat finding measures to improve the software in terms of efficiency, accuracy, and usability. It mainly aims at measuring the specification, functionality, and performance of a software program or application.

Software testing is divided into two types:

1. **Verification:** it refers to the set of tasks that ensure that the software correctly implements a specific function.
2. **Validation:** it refers to a different set of tasks that ensure that the software that has been built is traceable to customer requirements.

Verification: "Are we building the product right?"

Validation: "Are we building the right product?" Software Testing can be broadly classified into two types:

4.1.1 Manual testing

Manual testing includes testing software manually, i.e., without using any automation tool or any script. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behavior or bug. There are different stages for manual testing suchas unit testing, integration testing, system testing, and user acceptance testing.

Testers use test plans, test cases, or test scenarios to test software to ensure the completeness of testing. Manual testing also includes exploratory testing, as testers explore the software to identifyerrors in it.

4.1.2 Automation testing

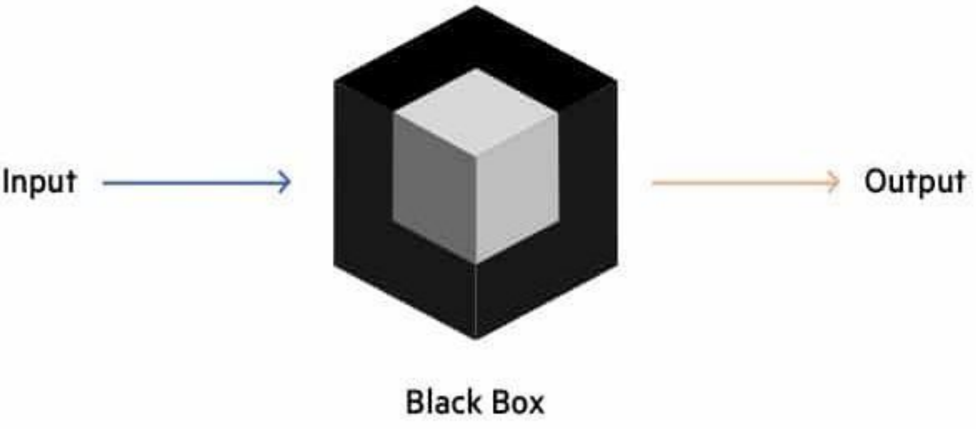
Automation testing, which is also known as Test Automation, is when the tester writes scripts and uses another software to test the product. This process involves the automation of a manual process. Automation Testing is used to re-run the test scenarios quickly and repeatedly, that were performed manually in manual testing. Apart from regression testing, automation testing is also used to test the application from a load, performance, and stress point of view. It increases the test coverage, improves accuracy, and saves time and money when compared to manual testing.

What are the different types of software testing?

Software testing techniques can be majorly classified into two categories:

4.1.2.1 Blackbox Testing

The technique of testing in which the tester doesn't have access to the software's source code and is conducted at the software interface without any concern with the internal logical structure of the software is known as black-box testing.



4.1.2.2 Whitebox Testing

The technique of testing in which the tester is aware of the internal workings of the product, has access to its source code, and is conducted by making sure that all internal operations are performed according to the specifications is known as white box testing.

Black Box Testing	White Box Testing
Internal workings of an application are not required	Knowledge of the internal workings is a must.
End users, testers, and developers.	Normally done by testers and developers.
This can only be done by a trial and error method.	Data domains and internal boundaries can be Better tested.
Also known as closed box/data-driven testing.	Also known as clear box/structural testing.

What are the different types of testing?

Software level testing can be majorly classified into 4 levels:

1. Unit Testing

A level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed.

2. Integration Testing:

A level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units.

4.2 Top Down Integration Testing

Top-Down Integration testing which is also known as Incremental integration testing.In this Top-Down approach, the higher-level modules are tested first after higher level modules the lower-level modules are tested. Then these modules undergo for integration accordingly. Here the higher-level modules refer to main module and lower-level modules refers to submodules. This approach uses Stubs which are mainly used to simulate the submodule, if the invoked submodule is not developed this Stub works as a momentary replacement.

4.3 Bottom-up Integration Testing

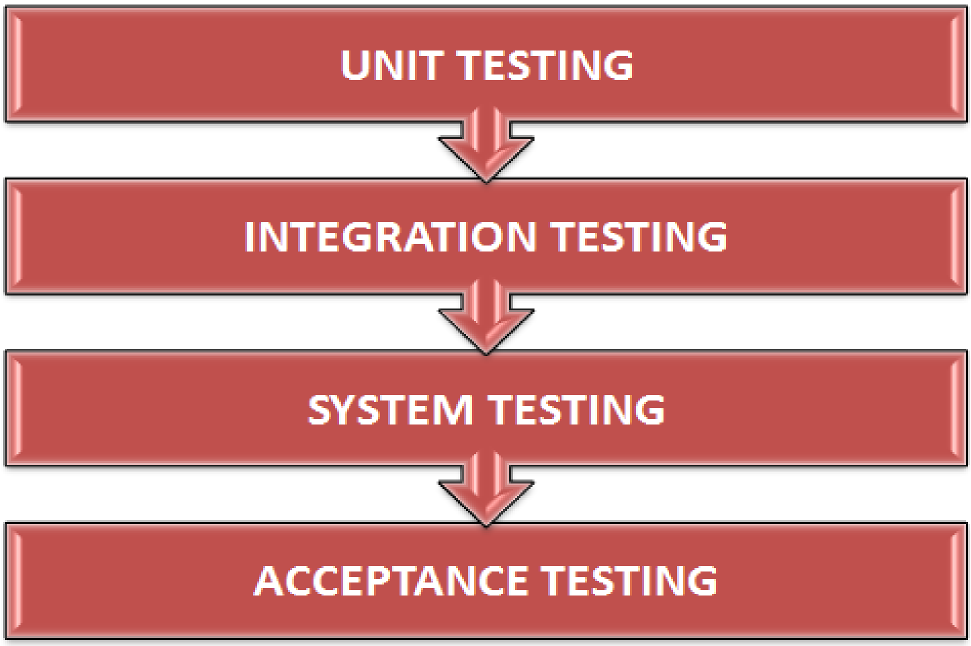
Bottom-Up Integration testing is another approach of Integration testing. In this Bottom-Up approach, the lower-level modules are tested first after lower-level modules the higher-level modules are tested. Then these modules undergo for integration accordingly. Here the lower-level modules refer to submodules and higher-level modules refers to main modules. This approach uses test drivers which are mainly used to initiate and pass the required data to the sub modules means from higher level module to lower-level module if required.3

4.3.1 System Testing:

A level of the software testing process where a complete, integrated system/software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

4.3.2 Acceptance Testing:

A level of the software testing process where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery



4.4 Source code:

```
import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

from matplotlib import style

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LinearRegression

from sklearn import metrics

from sklearn.metrics import mean_squared_error

from sklearn.metrics import mean_absolute_error

from sklearn.metrics import r2_score

from sklearn.ensemble import RandomForestRegressor

from sklearn.tree import DecisionTreeRegressor

from sklearn.model_selection import KFold

import pickle

df = pd.read_csv("insurance.csv")

df.head()

df.shape

df.columns

df.describe

df.info()
```

```
plt.figure(figsize=(5,5))

style.use('ggplot')

sns.countplot(x='sex', data=df)

plt.title('Gender Distribution')

plt.show()

plt.figure(figsize=(5,5))

sns.countplot(x='smoker', data=df)

plt.title('Smoker')

plt.show()

plt.figure(figsize=(5,5))

sns.countplot(x='region', data=df)

plt.title('Region')

plt.show()

plt.figure(figsize=(5,5))

sns.barplot(x='sex', y='charges',hue='smoker', data=df)

plt.title('Charges for smokers')

df[['age','bmi','children','charges','sex','smoker','region']].hist(bins=30, figsize=(10,10),
  color='blue')

plt.show()

df.head()

df['sex'] = df['sex'].apply({'male':0, 'female':1}.get)

df['smoker'] = df['smoker'].apply({'yes':1, 'no':0}.get)

df['region'] = df['region'].apply({'southwest':1, 'southeast':2, 'northwest':3,
  'northeast':4}.get)

plt.figure(figsize=(10,7))
```

```
sns.heatmap(df.corr(), annot = True)
plt.show()
df.plot(kind="box",subplots=True,sharex=False,sharey=False,figsize=(20,10),color='deeppink')
X = df.drop(['charges'], axis=1)
y = df.charges
linreg = LinearRegression()
print("train score:",linreg.score(x_train,y_train))
print("test score:",linreg.score(x_test,y_test))
print('MAE= ',metrics.mean_absolute_error(y_test,y_pred))
print('MSE= ',metrics.mean_squared_error(y_test,y_pred))
print(f"r2 score: {r2_score(y_test,y_pred)}")
print('Adjusted R2 value= ',1 - (1 - (linreg.score(x_test,y_test))) * ((756 - 1)/(756-10-1)))
print('RMSE (train)= ',np.sqrt(mean_squared_error(y_train,x_pred)))
print('RMSE (test)= ',np.sqrt(mean_squared_error(y_test,y_pred)))
data = {'age':50,'sex':0, 'bmi':25, 'children':2, 'smoker':1, 'region':2}
index = [0]
cust_df = pd.DataFrame(data, index)
cust_df
cost_pred = linreg.predict(cust_df)
print("The medical insurance cost of the new customer is: ", cost_pred)
# Fitting Random Forest Regression to the dataset
regressor = RandomForestRegressor(n_estimators = 100)
regressor.fit(x_train,y_train)
```

```
x_pred=regressor.predict(x_train)

y_pred=regressor.predict(x_test)

print("train score:",regressor.score(x_train,y_train))

print("test score:",regressor.score(x_test,y_test))

print('MAE= ',metrics.mean_absolute_error(y_test,y_pred))

print('MSE= ',metrics.mean_squared_error(y_test,y_pred))

print(f"r2 score: {r2_score(y_test,y_pred)}")

print('Adjusted R2 value= ',1 - (1 - (regressor.score(x_test,y_test))) * ((756 - 1)/(756-
  10-1)))

print('RMSE (train)= ',np.sqrt(mean_squared_error(y_train,x_pred)))

print('RMSE (test)= ',np.sqrt(mean_squared_error(y_test,y_pred)))

data = {'age':50,'sex':0, 'bmi':25, 'children':2, 'smoker':1, 'region':2}

index = [0]

cust_df = pd.DataFrame(data, index)

cust_df

cost_pred = regressor.predict(cust_df)

print("The medical insurance cost of the new customer is: ", cost_pred[0])

# Fitting Decision Tree Regression to the dataset

regressordt = DecisionTreeRegressor()

regressordt.fit(x_train,y_train)

x_pred=regressordt.predict(x_train)

y_pred=regressordt.predict(x_test)

print("train score:",regressordt.score(x_train,y_train))

print("test score:",regressordt.score(x_test,y_test))
```

```
  print('MAE= ',metrics.mean_absolute_error(y_test,y_pred))

  print('MSE= ',metrics.mean_squared_error(y_test,y_pred))

  print(f"r2 score: {r2_score(y_test,y_pred)}")

  print('Adjusted R2 value= ',1 - (1 - (regressordt.score(x_test,y_test))) * ((756 - 1)/(756-
    10-1)))

  print('RMSE (train)= ',np.sqrt(mean_squared_error(y_train,x_pred)))

  print('RMSE (test)= ',np.sqrt(mean_squared_error(y_test,y_pred)))

  kf=KFold(n_splits=7)

  kf

  for train_index,test_index

  kf.split(['age','sex','bmi','children','smoker','region','charges']):

  print(train_index,test_index)

  def get_score(model,x_train,x_test,y_train,y_test):

  model.fit(x_train,y_train)

return model.score(x_test,y_test)

get_score(LinearRegression(),x_train,x_test,y_train,y_test)

get_score(RandomForestRegressor(n_estimators = 100),x_train,x_test,y_train,y_test)

get_score(DecisionTreeRegressor(),x_train,x_test,y_train,y_test)

from sklearn.linear_model import LinearRegression

regressor = LinearRegression()

regressor.fit(x_train,y_train)

pickle.dump(regressor, open('model.pkl','wb'))

model=pickle.load(open('model.pkl','rb'))
```

App=Flask.py

```
from flask import Flask, request, render_template import pickle

(_name_, template_folder='template') model = pickle.load(open("model.pkl", "rb"))

@app.route('/') def home():

return render_template('index.html')

@app.route('/health') def health():

return render_template('health.html')

@app.route('/predict', methods = ["POST", "GET"]) def predict():

age = int(request.form['age']) gender = int(request.form['gender']) bmi =
float(request.form['bmi'])

children = int(request.form['children']) smoke = int(request.form['smoke']) region =
int(request.form['region'])

result = model.predict([[age, gender, bmi, children, smoke, region]]) return
render_template('submit.html', result="$ {:.2f}".format(result[0]))

if _name_ == '_main_': app.run(debug=True)
```

INDEX.html

```
<!DOCTYPE html>

<html lang="en" xmlns:https="http://www.w3.org/1999/xhtml">

<head>

    <meta charset="UTF-8">

    <title>Life Line</title>

    <link href='https://fonts.googleapis.com/css?family=Josefin Sans' rel='stylesheet'>

    <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
```

```
</head>

<body>

<nav>

    <label class="logo">Bringing The Future of Health Care</label>

    <ul class="menu">

        <li><a href="#">About</a></li>

        <li><a href="#">Services</a></li>

        <li><a href="#">Contact</a></li>

    </ul>

</nav>

<section class="about">

    <div class="main">

        <div class="about-text">

            <h1>About Us</h1>


            <p>Health insurance cost prediction system. This website helps you to predict health insurance cost of the user with the help of the attributes given by the user. To know your health insurance cost click on get started.</p>

            <button type="button" onclick="window.location.href='{{ url_for('health') }}';">Get Started</button>

        </div>

        <img src="https://images.moneycontrol.com/static-mcnews/2021/11/Health-insurance-770x433.jpg?impolicy=website&width=770&height=431">

    </div>

</section>
```

```
</body>

</html>

<style>

   * {

       padding: 0;

       margin: 0;

       font-family: 'roboto', sans-serif;

       box-sizing: border-box;

   }

 body {

       background-image:                url('https://www.statnews.com/wp-
content/uploads/2021/04/AdobeStock_340574222-1-1600x900.jpeg');

       background-repeat:  no-repeat;

       background-attachment: fixed;

       background-size: cover;

   }

nav {

       font-family: 'roboto', sans-serif;

       height: 70px;

       width: 100%;

       display: flex;

       justify-content: space-between;

       align-items: center;
```

```
        background-color: rgba(255, 255, 255, 0.9);

        padding: 0 20px;

        box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);

    }

    label.logo {

        color: #03254c;

        font-size: 35px;

        line-height: 80px;

        font-weight: bold;

    }

.menu {

        list-style: none;

        display: flex;

}

 .menu li {

        margin-right: 20px;

    }

.menu a {

        text-decoration: none;

        color: #03254c;

        font-weight: bold;

        font-size: 16px;

        transition: color 0.3s;

    }
 .menu a:hover {
        color: #67032f;
    }
.main {
        display: flex;
        justify-content: flex-start;
    }
.about {
        padding: 20px; /* Add padding as needed */
    }
 .about-text {
        max-width: 50%; /* Adjust as needed */
        padding: 40px; /* Add padding to move the button down */
        background-color: rgba(255, 255, 255, 0.8); /* Add background color to improve
readability */
    }
.about-text h1 {
        text-align: left; /* Adjust alignment as needed */
    }
 .about-text h5 {
        text-align: left; /* Adjust alignment as needed */
    }
.about-text p {
        text-align: left; /* Adjust alignment as needed */
```

```
    }

    button {

        background: #67032f;

        color: white;

        border: 2px solid transparent;

        font-weight: bold;

        padding: 13px 30px;

        border-radius: 30px;

        transition: .4s;

    }

  button:hover {

        background: #fff;

        color: #67032f;

        border: 2px solid #67032f;

        cursor: pointer;

    }

  /* Rest of your CSS remains unchanged */

</style>
```

Health.html

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Ensure Complete Peace Of Mind</title>
    <link href="healthstyle.css" rel="stylesheet" type="text/css">

    <style>

        body {

            background-image:
url('https://i.pinimg.com/736x/23/e8/20/23e820216a84f32bc11077c20c0e3f0e.jpg');

            background-repeat: no-repeat;

            background-attachment: fixed;

            background-size: cover;

            margin: 0;

            padding: 0;

        }

* {

            box-sizing: border-box;

            margin: 0;

            padding: 0;

        }

 nav {

            font-family: 'Bradley Hand, cursive';

            height: 70px;

            width: 100%;

            display: flex;

            justify-content: space-between;

            align-items: center;

            background-color: rgba(255, 255, 255, 0.9);
            padding: 10px 20px;

            box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);

        }

  label.logo {

            color: #03254c;

            font-size: 35px;

            line-height: 80px;

            font-weight: bold;

        }

 .heading {

            text-align: center;

            padding-top: 100px;

            padding-right: 30px;

            color: #67032f;

        }

 button {

            background: #67032f;

            color: white;

            text-decoration: none;

            border: 2px solid transparent;

            font-weight: bold;

            padding: 13px 30px;

            border-radius:  30px;

            transition: .4s;
        }
   button:hover {
            background: transparent;
            border: 2px solid #67032f;
            cursor: pointer;
        }
   form {
            padding-left: 20px;
            padding-right: 20px; /* Added right padding */
            padding-top: 20px;
            font-size: 20px;
            max-width: 400px;
            margin: 0 auto;
        }
    .input {
            height: 30px;
            width: 100%;
            font-size: 15px;
            margin-bottom: 10px;
        }
    .step-box {
            max-width: 200px;
            margin: 0 auto;
            background-color: rgba(255, 255, 255, 0.8);
            padding: 15px;

            border-radius: 10px;

            margin-bottom: 15px;

        }

  .parent {

            display: flex;

            justify-content: space-between; /* Added to align left and right */

            flex-wrap: wrap; /* Added for responsiveness */

        }

   .child,

        .child2 {

            flex: 0 0 48%; /* Adjusted width */

            padding: 1rem;

            margin-bottom: 10px;

        }

        .bt {

            text-align: center;

            padding-top: 20px;

        }

    </style>

  </head>

  <body>

    <nav>

        <label class="logo">Ensure Complete Peace Of Mind</label>
  </nav>
  <div class="heading">
    <h1>PREDICTING HEALTH INSURANCE !</h1>
  </div>
  <form action='/predict' method="POST">
    <div class="parent">
      <div class="child">
        <label>Age :</label><br><br>
        <input type="number" name="age" placeholder="Age > 18" min="18"
max="64" required="required" class="input"><br>
      </div>
      <div class="child">
        <label>BMI :</label><br><br>
        <input type="number" name="bmi" placeholder="BMI"
required="required" class="input"><br>
      </div>
      <div class="child">
        <label>Gender :</label><br><br>
        <input type="number" name="gender" placeholder="0-Male / 1-Female"
min="0" max="1" required="required"
          class="input"><br>
      </div>
      <div class="child2">
        <label>Children :</label><br><br>
```

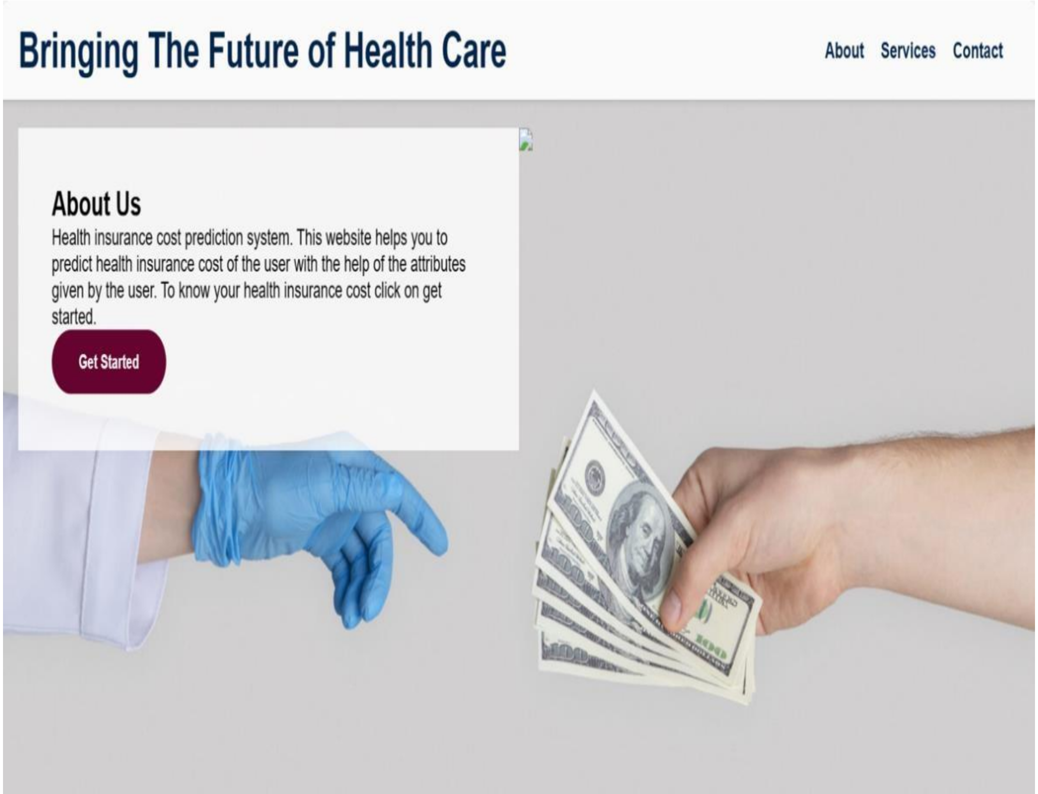
```
            <input type="number" name="children" placeholder="0 for None" min="0"
max="5" required="required" class="input"><br>
        </div>
        <div class="child2">
            <label>Do you smoke ?</label><br><br>
            <input type="number" name="smoke" placeholder="1-Yes / 0-No"
max="1" min="0" required="required"
                class="input"><br>
        </div>
        <div class="child2">
            <label>Region :</label><br><br>
            <input type="number" name="region" placeholder="1-SW / 2-SE / 3-NW /
4-NE" max="4" min="1"
                required="required" class="input"><br>
        </div>
      </div>
      <div class="bt">
        <button type="submit">Predict</button>
      </div>
   </form>
</body>
</html>
```

Submit.html

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>
  <meta charset="UTF-8">
  <title>FINAL PREDICTION</title>
  <link href="https://fonts.googleapis.com/css?family=Josefin+Sans"
rel="stylesheet">
  <style>
    body {
      background-image: url('https://www.statnews.com/wp-
content/uploads/2021/04/AdobeStock_340574222-1-1600x900.jpeg');
      background-repeat: no-repeat;
      background-attachment: fixed;
      background-size: cover;
      margin: 0;
      padding: 0;
      font-family: 'Josefin Sans','crimson text';
    }
    nav {
      background-color:#fff;
      box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
      padding: 10px 20px;
      position: fixed;
      width: 100%;
      z-index: 1000;
    }
    label.logo {
            color: #990000;

            font-size: 24px;

            font-weight: bold;

        }

        h2 {

            color: #fff;

            font-size: 28px;

            font-weight: bold;

            padding-top: 120px;

            text-align: center;

        }

        }

    </style>

</head>

<body>

    <nav>

        <label class="logo">Medi Care</label>

    </nav>

    <h2>Predicted Health Insurance Cost Is</h2>

    <h1>{{result}}</h1>

</body>

</html>
```

4.5 Output Screens:



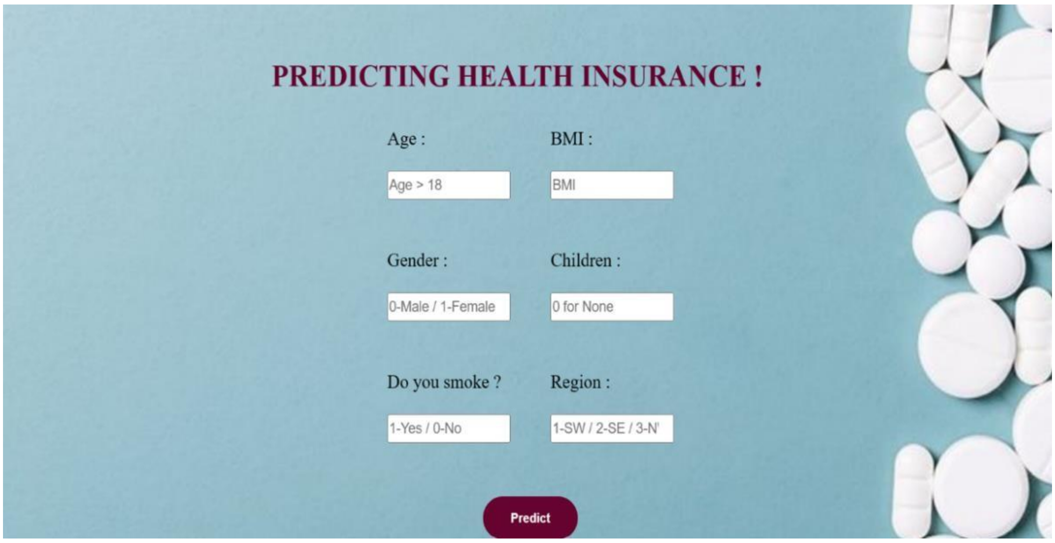
Bringing The Future of Health Care

About Us

Health insurance cost prediction system. This website helps you to predict health insurance cost of the user with the help of the attributes given by the user. To know your health insurance cost click on get started.

Figure:4.5.1 Home Screen

The above page shows the Home screen of web page



PREDICTING HEALTH INSURANCE !

Age : Age > 18

BMI : BMI

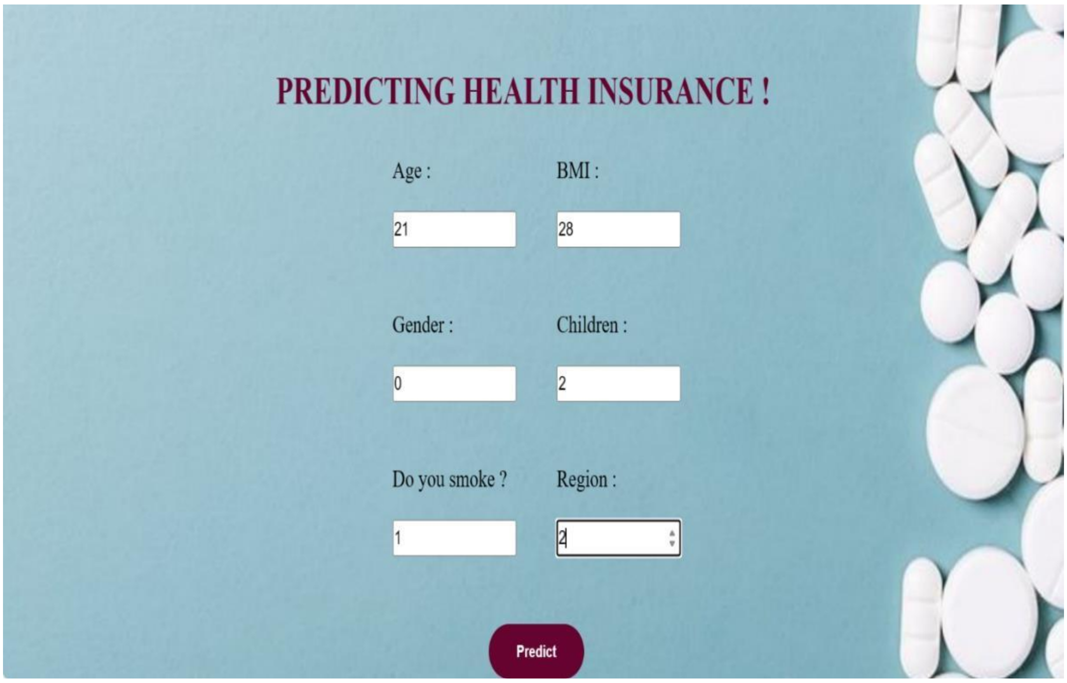
Gender : 0-Male / 1-Female

Children : 0 for None

Do you smoke ? 1-Yes / 0-No

Region : 1-SW / 2-SE / 3-N

Figure: 4.5.2 Prediction Form



PREDICTING HEALTH INSURANCE !

Age : 21

BMI : 28

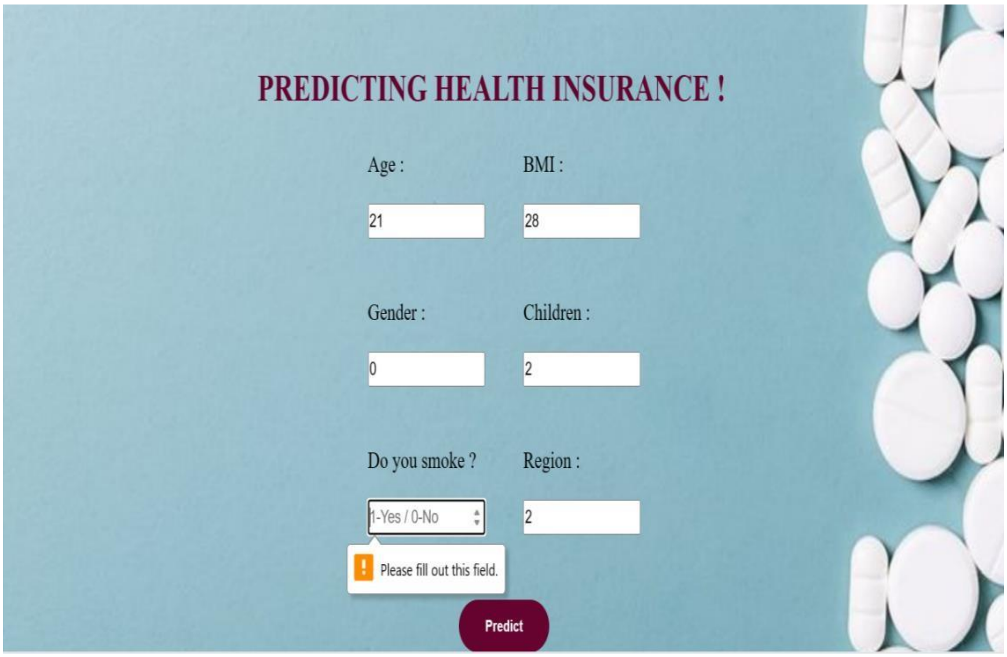
Gender : 0

Children : 2

Do you smoke ? 1

Region : 2

Figure: 4.5.3Form Validation



PREDICTING HEALTH INSURANCE !

Age : 21

BMI : 28

Gender : 0

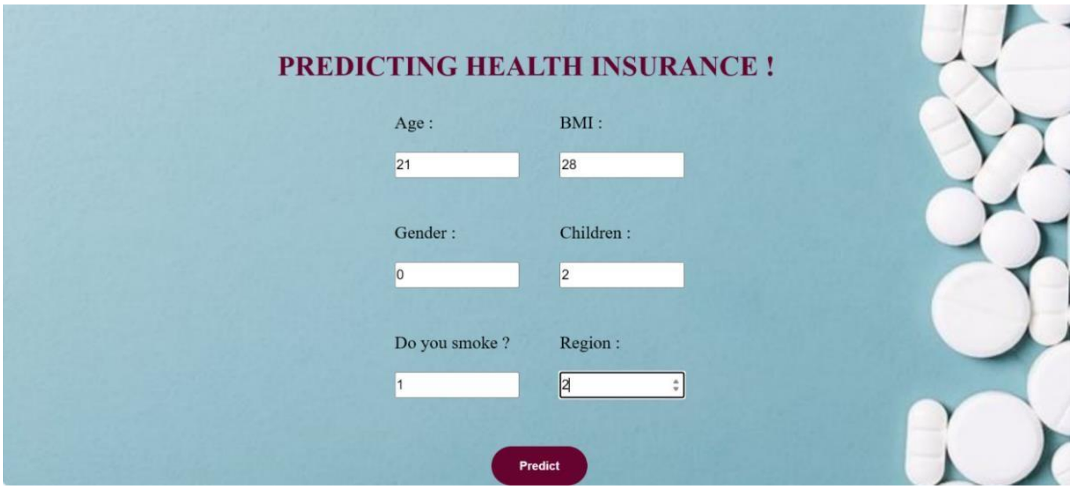
Children : 2

Do you smoke ? 1-Yes / 0-No

Please fill out this field.

Region : 2

Figure:4.5.4 Missing Field Validation



PREDICTING HEALTH INSURANCE !

Age : 21

BMI : 28

Gender : 0

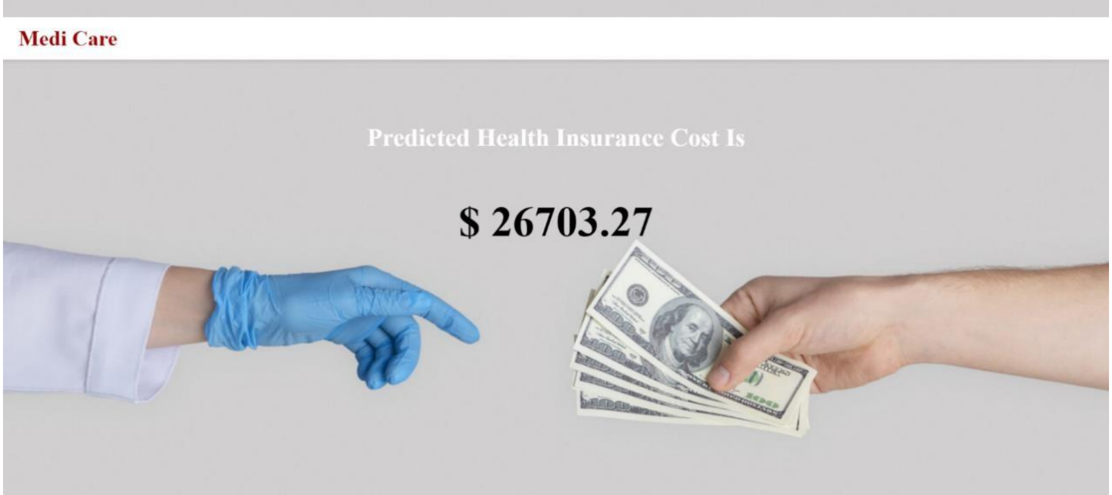
Children : 2

Do you smoke ? 1

Region : 2

Figure:4.5.5 Inserting values in the form

The cost of medical insurance accurately that can by an individual based on their basic information as parameters such as age,BMI,gender,children,smoke,religion.



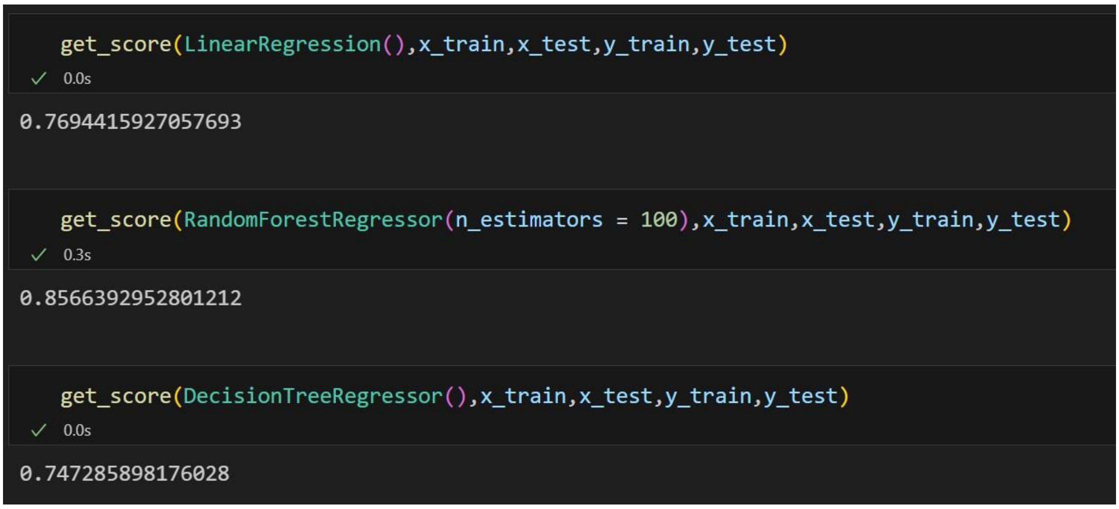
Medi Care

Predicted Health Insurance Cost Is

$ 26703.27

Figure: 4.5.6 Output Screen

Predicting the final output health insurance cost of a person.



```
get_score(LinearRegression(),x_train,x_test,y_train,y_test)
```

```
0.7694415927057693
```

```
get_score(RandomForestRegressor(n_estimators = 100),x_train,x_test,y_train,y_test)
```

```
0.8566392952801212
```

```
get_score(DecisionTreeRegressor(),x_train,x_test,y_train,y_test)
```

```
0.747285898176028
```

Figure:4.5.7 Final Scores

The output screens illustrates the accuracy of each algorithm,revealing variations in performance across different categories with some achieving perfect accuracy linear regression, Randomforest regression, Decession tree regression.

CHAPTER-05

CONCLUSION AND FUTURE ENHANCEMENTS

BIBLIOGRAPHY

CHAPTER-05

CONCLUSION AND FUTURE ENHANCEMENTS
BIBLIOGRAPHY

5.1 Conclusion:

An insurance claim prediction system has been implemented by using machine learning algorithms. Machine learning (ML) is a vital component of computational intelligence that has the potential to solve a range of problems in various applications and systems by leveraging historical data. Here, machine learning regression models were used to forecast health insurance charges based on specific attributes using the medical cost personal dataset.

The main focus was on predicting the premium amount based on a person's health rather than the insurance company's terms and conditions. The predicted premiums generated by the models with the actual premiums were compared to assess the accuracies of the models. It was discovered that Random Forest Regression is the efficient model, exhibiting an accuracy of 93.5%. Hence, Random Forest Regression can be used for estimating insurance costs as it provides better performance than other regression models.

5.2 Future Enhancement:

The system can be made more flexible and scalable using some recommendations.

Please note that the system implemented here is just a prototype of idea presented via this
project. The recommendations are as follows:

- In future, different types of attributes can be used for the prediction of the medical insurance claim amount for better accuracy.
- The system can be made more flexible to allow different types insurance claim predictions based on situations.
- The system can be made to be available with different insurance claims at one system like health insurance, vehicle insurance etc.

5.3 BIBILOGRAPHY:

[1] Mukund Kulkarni, Dhammadeep D. Meshram, Bhagyesh Patil, Rahul More, Mridul Sharma and Pravin Patange, "Medical Insurance Cost Prediction using Machine Learning", International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue XII Dec 2022.

[2] Sujith Thota, Kotha Vishnu Sai and P Swarnalatha, "Machine Learning Implementation for Health insurance", International Journal of Advanced Trends in Computer Science and Engineering, ISSSN 2278-3091.

[3] Nidhi Bhardwaj & Rishabh Anand, "Health Insurance Amount Prediction," International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 9 Issue 05, May-2020.

[4] Rama Devi Burri, Ram Burri, Ramesh Reddy Bojja & Srinivasa Rao Buruga,"Insurance Claim Analysis Using Machine Learning Algorithms", InternationalJournal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8, Issue- 6S4, April 2019.

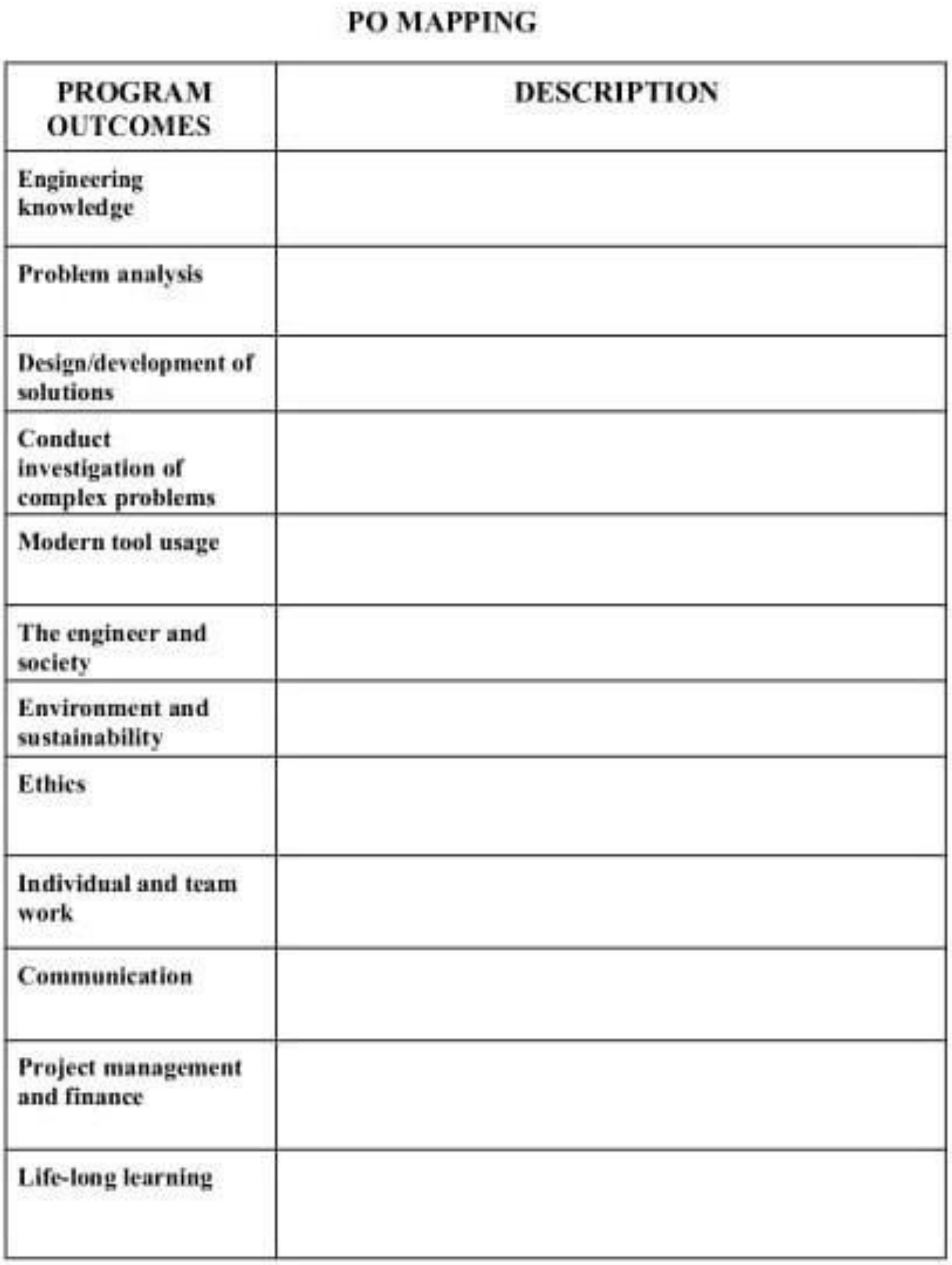
[5] M. A. Fauzan and H. Murfi, "The accuracy of XGBoost for insurance claim prediction," International Journal of Advanced Software Computer Applications, vol. 10, no. 2, 2018

[6] A. Tike and S. Tavarageri, "A medical price prediction system using hierarchical decision trees," in Proceedings of the IEEE International Conference on Big Data (Big
Data), pp. 3904 –3913, IEEE, Boston, MA, USA, December 2017.

[7] Belhadji, E., G. Dionne, and F. Tarkhani, ―A Model for the Detection of InsuranceFraud, Geneva Papers on Risk and Insurance Theory‖, 25: 517-538, May 2012.

[8] Palanisamy, V., & Thirunavukarasu, R. (2019). Implications of big data analytics in developing healthcare frameworks – a review.
10.1016/j.jksuci.2017.12.007.HEALTH INSURANCE CLAIM PREDICTION USINGMACHINE LEARNING Department of CSE, SVPCET 2020-2024 .

[9] L. S. Chen and J. C. Chen, "Using data mining methods to detect medical fraud.



PO MAPPING

PROGRAM OUTCOMES	DESCRIPTION
Engineering knowledge	
Problem analysis	
Design/development of solutions	
Conduct investigation of complex problems	
Modern tool usage	
The engineer and society	
Environment and sustainability	
Ethics	
Individual and team work	
Communication	
Project management and finance	
Life-long learning	