

# **REAL ESTATE VALUATION USING MACHINE LEARNING**

*Dissertation submitted in partial fulfilment of requirements for the award of the degree of*

## **Master of Computer Applications (M.C.A)**

*Submitted By*

**SURI SAI PRAVEEN**

Regd. No: 21131F0065

*Under the esteemed guidance of*

**Mr. B. BALAKRISHNA**

**Assistant Professor**

**Department of Computer Applications**



**COLLEGE OF ENGINEERING  
(AUTONOMOUS)**

**Department of Computer Applications**

**GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING**

**(AUTONOMOUS)**

(Affiliated to JNTU-K, Kakinada, A.P)

**VISAKHAPATNAM-530048**

2021-2023

# CERTIFICATE



## GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING (AUTONOMOUS)

This is to certify that, the dissertation titled **“REAL ESTATE VALUATION USING MACHINE LEARNING”** is submitted by **Mr. SURI SAI PRAVEEN** in partial fulfillment of the requirement forward of the Degree of **M.C.A** in **GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING (AUTONOMOUS)** affiliated to Jawaharlal Nehru Technological University, Kakinada is a bonafide record of project carried out by him under my supervision.

The contents of the project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

**Internal Guide**

**Head of the Department**

**External Examiner**

## **DECLARATION**

I hereby declare the dissertation titled “**REAL ESTATE VALUATION USING MACHINE LEARNING**” is submitted to the Department of Computer Applications, **Gayatri Vidya Parishad College of Engineering(Autonomous)** affiliated to Jawaharlal Nehru Technological University, Kakinada in partial fulfilment of the requirements for the award of the degree of **Master of Computer Applications(MCA)**. This work is done by me and authentic to the best of my knowledge under the direction and valuable guidance of **Mr. B. Balakrishna**, Assistant Professor in the Department of Computer Applications.

**(SURI SAI PRAVEEN)**

## **ACKNOWLEDGEMENT**

I take the opportunity to thank everyone who has contributed to making the project possible. I am thankful to Gayatri Vidya parishad College of Engineering (Autonomous) for giving opportunity me to work on a project as part of the curriculum.

I offer my profuse thanks to **Dr. A. B. Koteswara Rao**, the principal of **Gayatri Vidya Parishad College of Engineering (Autonomous)** for providing the best faculty and lab facilities throughout the completion of Master of Computer Applications course.

I offer my profuse thanks to Prof. **Dr. K. Narasimha Rao**, M.Tech., Ph.D., Professor and Dean, Academics (PG) for his valuable suggestions and constant motivation that greatly helped me complete project successfully.

I offer my profuse thanks to **Dr. Y. Anuradha**, Associate Professor, Head of the Department of Computer Applications for her valuable suggestions and constant motivation that greatly helped me to complete the project successfully.

My sincere thanks to **Mr. B. Balakrishna**, Assistant Professor, Department of Computer Applications, my project guide, for his constant support, encouragement. I am very much grateful for his valuable suggestions.

**SURI SAI PRAVEEN**

**(21131F0065)**

# **ABSTRACT**

## **ABSTRACT**

Buying a dream house is a common aspiration for people in these days, but accurately predicting its price has become more challenging, especially for those without expertise in this field. This difficulty often hinders informed decision-making when it comes to purchasing property. However, there is a proposed system that aims to simplify house price prediction by considering crucial factors. The system utilizes multiple regression techniques and bases on their performance through weighted averaging, leading to highly accurate predictions. This approach minimizes error and enhances accuracy.

## INDEX

<i>Certificate</i>	<i>i</i>
<i>Declaration</i>	<i>ii</i>
<i>Acknowledgement</i>	<i>iii</i>
<i>Abstract</i>	<i>iv, v</i>
<i>Table of contents</i>	<i>vi, vii</i>
<i>List of Tables</i>	<i>viii</i>

## TABLE OF CONTENTS

<u>CHAPTER NAME</u>	<u>PAGE NO</u>
<b>1. INTRODUCTION.....</b>	<b>1</b>
1.1 INTRODUCTION.....	2-3
<b>2. LITERATURE REVIEW.....</b>	<b>4</b>
2.1 LITERATURE REVIEW.....	5-6
<b>3. SYSTEM ANALYSIS.....</b>	<b>7</b>
3.1 EXISTING SYSTEM.....	8
3.2 PROPOSED SYSTEM.....	8
<b>4. SYSTEM REQUIREMENTS AND SPECIFICATION.....</b>	<b>9</b>
4.1 HARDWARE REQUIREMENTS.....	10
4.2 SOFTWARE REQUIREMENTS.....	10
4.3 FUNCTIONAL REQUIREMENTS.....	11
4.4 NON-FUNCTIONAL REQUIREMENTS.....	11
<b>5. DATASET.....</b>	<b>12</b>
5.1 DATASET .....	13
<b>6. SYSTEM DESIGN.....</b>	<b>14</b>
6.1 ARCHITECTURE DIAGRAM.....	15
6.2 UML DIAGRAMS.....	16
6.2.1 USE CASE DIAGRAM.....	17

6.2.2	ACTIVITY DIAGRAM.....	18
6.2.3	SEQUENCE DIAGRAM.....	19
<b>7.</b>	<b>IMPLEMENTATION.....</b>	<b>20</b>
7.1	IMPLEMENTATION.....	21
7.2	TECHNOLOGIES USED.....	21
7.3	LINEAR REGRESSION ALGORITHM.....	22
7.4	K-NEAREST NEIGHBOUR (KNN) ALGORITHM.....	22
7.5	DECISION TREE ALGORITHM.....	23
<b>8.</b>	<b>SOURCE CODE.....</b>	<b>24</b>
8.1	SOURCE CODE.....	23-30
<b>9.</b>	<b>OUTPUT SCREENS.....</b>	<b>31</b>
9.1	OUTPUT SCREENS.....	32-45
<b>10.</b>	<b>SYSTEM TESTING.....</b>	<b>46</b>
10.1	SYSTEM TESTING.....	47
10.2	UNIT TESTING.....	47
10.3	INTEGRATION TESTING.....	47
10.4	BLACK BOX TESTING.....	47
<b>11.</b>	<b>TEST CASES.....</b>	<b>48</b>
11.1	TEST CASES.....	49
<b>12.</b>	<b>CONCLUSION.....</b>	<b>50</b>
12.1	CONCLUSION.....	51
<b>13.</b>	<b>REFERENCE.....</b>	<b>52</b>
13.1	REFERENCE.....	53-54



**LIST OF TABLES:**

<b>TABLE NO</b>	<b>NAME OF THE FIGURE</b>	<b>PAGENO</b>
10.1	TESTCASES	47

**CHAPTER 1**  
**INTRODUCTION**

## **INTRODUCTION**

House marketing is a complex and dynamic system influenced by various factors such as floors, size, amenities, and market trends. Predicting house prices accurately is essential for both buyers and sellers to make informed decisions. Regression is a widely used statistical technique that assumes a linear relationship between the independent variables (such as square footage, number of bedrooms, etc.) and the dependent variable (house price). By analyzing historical housing data and applying some regressions, we can create a model that can estimate the price of a house based on its characteristics.

Regression models, such as Decision tree, play a crucial role in the house marketing domain by providing insights into the factors that drive house prices. These models enable us to understand the impact of variables like floors, size, amenities, and market trends on house prices and make predictions based on this understanding.

The real estate market plays a pivotal role in the economy, and understanding the factors influencing property values is of paramount importance for buyers, sellers, and investors alike. Through this system, we embark on a data-driven journey, harnessing the power of machine learning to make informed and reliable predictions about property prices.

Our comprehensive approach involves collecting historical real estate data, preprocessing it to ensure its quality, and selecting the most influential features for our models. We then train the linear regression, decision tree regression, and k-nearest neighbors regression algorithms, meticulously evaluating their performance to identify the optimal predictor.

Among the three models, the decision tree regression algorithm stands out as a potent candidate due to its ability to handle complex relationships between various features and house prices. By leveraging the strengths of this algorithm, we aspire to deliver an accurate and interpretable valuation model that can guide buyers and sellers towards well-informed decisions.

This system significance lies in its potential to empower individuals and real estate professionals alike, enabling them to navigate the ever-changing property market with confidence. Through the deployment of our predictive model, users can obtain estimated house prices based on input features, thus facilitating sound financial choices.

The depths of real estate valuation, we anticipate unearthing valuable insights and making a positive impact in the domain of property transactions. Let us embark on this exciting journey of prediction, discovery, and innovation in the realm of real estate valuation using machine learning algorithms.

Real estate valuation is an essential aspect of the property market, as it directly influences buying, selling, and investment decisions. For sellers, accurate property valuation ensures that they price their homes competitively, attracting potential buyers while maximizing their returns. On the other hand, buyers rely on valuation to gauge the fair market value of a property, helping them make informed offers and avoid overpaying.

Additionally, real estate investors depend on precise valuation models to identify lucrative investment opportunities and optimize their portfolio strategies. In this context, our system focusses on developing a reliable and accurate valuation model and holds immense value for all stakeholders involved in real estate transactions.

Real estate valuation poses unique challenges due to the multifaceted nature of property markets. Housing prices are influenced by a myriad of factors, including floors, property size, condition, amenities, and local market conditions.

Moreover, non-linear relationships between these features can complicate the valuation process, necessitating sophisticated algorithms capable of capturing complex interactions. By exploring multiple regression techniques, we aim to address these challenges and construct a comprehensive model that embraces the intricacies of real estate valuation.

Machine learning techniques offer a promising avenue to tackle the complexities of real estate valuation. Unlike traditional approaches, which may rely on simplistic assumptions, machine learning algorithms can uncover hidden patterns and non-linear relationships in data.

By leveraging large datasets and powerful computation, our models can learn from historical transactions and generalize to predict house prices accurately for new properties. This innovation holds the potential to revolutionize the real estate industry, providing stakeholders with a more data-driven, reliable, and transparent approach to property valuation.

The key components of our real estate valuation system. The system will begin by exploring the dataset and performing essential data preprocessing steps to ensure its quality and reliability. Next, we will conduct feature selection, identifying the most influential attributes that impact property prices. With this refined dataset in hand, we will train and fine-tune our regression models—linear regression, decision tree regression, and k-nearest neighbors' regression—using appropriate evaluation metrics to gauge their performance. The most accurate and robust model will be chosen for deployment in a user-friendly interface, allowing users to input property features and receive real-time price predictions.

**CHAPTER 2**  
**LITERATURE REVIEW**

## LITERATURE REVIEW

**[1] Li, J., Liu, C., Zhang, H."XGBoost for Real Estate Valuation: A Comparative Study".International Journal of Geographical Information Science, 2020.**

This study focuses on the use of the XGBoost algorithm for real estate valuation. The authors conducted a comparative analysis of XGBoost with other machine learning models, using a dataset of residential properties. The results demonstrated that XGBoost outperformed the other models in terms of accuracy and feature importance, indicating its effectiveness for real estate valuation tasks.

**[2] Zhao, X., Li, H., Wang, Y."Gaussian Process Regression for Real Estate Valuation: A Comprehensive Study".Computers, Environment and Urban Systems, 2022.**

This research explores the application of Gaussian process regression (GPR) for real estate valuation. The authors conducted a comprehensive study to evaluate the performance of GPR using a dataset of residential properties. The findings demonstrated the effectiveness of GPR in capturing complex patterns and achieving accurate predictions in real estate valuation.

**[3] Chen, T., Liu, Y., Liu, X."Real Estate Valuation Using Support Vector Regression: A Comparative Study".International Journal of Computational Intelligence Systems, 2019.**

This study focuses on the application of support vector regression (SVR) for real estate valuation. The authors compared SVR with other machine learning models and evaluated their performance using a dataset of residential properties. The findings demonstrated that SVR outperformed the other models in terms of accuracy and robustness, indicating its suitability for real estate valuation tasks

**[4] Johnson, M., Smith, K., Brown, J."Neural Network Models for Real Estate Valuation: A Comparative Analysis".Expert Systems with Applications, 2021.**

This paper explores the use of neural network models for real estate valuation. The authors conducted a comparative analysis of different neural network architectures and evaluated their performance using a dataset of commercial properties. The study found that certain neural network models, such as feedforward neural networks and recurrent neural networks, achieved accurate predictions and showed potential for real estate valuation.

**[5] Wang, Q., Zhang, L., Chen, S."Real Estate Valuation Using Random Forests: An Empirical Analysis".Journal of Real Estate Portfolio Management, 2018.**

This research investigates the application of random forests for real estate valuation. The authors utilized a large dataset of residential properties and compared the performance of random forests with others.

**[6]. Title: "Using Natural Language Processing for Real Estate Price Prediction Authors: Li, J., Wang, Y., Zhang, L. Journal: Expert Systems with Applications, 2019**

Focusing on the application of natural language processing (NLP) techniques, this study explored the use of property descriptions and text data in real estate price prediction. The authors developed NLP-based models to analyze textual information and incorporate it into the prediction process. The findings demonstrated the potential of NLP in improving prediction accuracy by leveraging unstructured data.

**[7]. "Real Estate Price Prediction Using Ensemble Learning Techniques Authors: Zhang, H., Chen, L., Wang, Q. Journal: Journal of Real Estate Finance and Economics, 2019**

This study explores the use of ensemble learning techniques, such as bagging and stacking, for real estate price prediction. The authors conducted experiments using various models and evaluated their performance on a dataset of residential properties. Their findings revealed that ensemble learning techniques improved prediction accuracy and robustness, making them valuable for real estate valuation tasks.

**[8]. "A Comparative Study of Machine Learning Algorithms for Real Estate Valuation" Authors: Kim, J., Lee, S., Park, Y. Journal: International Journal of Advanced Computer Science and Applications, 2018**

This research presents a comparative study of different machine learning algorithms, including support vector machines, random forests, and gradient boosting, for real estate valuation. The authors utilized a diverse dataset of commercial properties and evaluated the models' accuracy and efficiency. The study identified the strengths and weaknesses of each algorithm, providing valuable insights for real estate professionals.

**[9]. "Deep Learning Models for Real Estate Valuation: An Empirical Investigation" Authors: Chen, T., Liu, Y., Wang, H. Journal: Neural Computing and Applications, 2021**

This research investigates the application of deep learning models, such as long short-term memory (LSTM) networks and convolutional neural networks (CNN), for real estate valuation. The authors trained and compared these models using a dataset of residential properties. The study found that deep learning models outperformed traditional machine learning algorithms, providing more accurate predictions.

**[10]. Title: "Geospatial Analysis for Real Estate Valuation: A Review" Authors: Garcia, N., Rodriguez, E., Martinez, A Journal: International Journal of Geographical Information Science, 2022**

This review article highlights the importance of geospatial analysis in real estate valuation. The authors examined various geospatial data sources, such as proximity to amenities, transportation networks, and crime rates, and their impact on property prices.

**CHAPTER 3**  
**SYSTEM ANALYSIS**



## **SYSTEM ANALYSIS**

### **Existing System:**

When using the Random Forest algorithm to predict house prices in the existing system, there are number of challenges that can arise. One of these challenges is the quality and quantity of data used for training. If there is not enough data or the data is of poor quality, the model may not be able to accurately predict house prices.

### **DISADVANTAGES**

- Time efficiency is high.
- Don't know which algorithm can offer better results.

### **Proposed System:**

The proposed system is to predict the house price index using some regression involves connecting a data set of house sale prices and their features, preprocessing the data, splitting it into training and testing sets, training model on the training set, evaluating the model's accuracy on the testing set and deploying it for use. By using the best regression model, the system predicts the price of a house based on its features, such as the house type, size, bhk and location. The system is useful for homeowners, buyers, and real estate agents to estimate the value of a house and make informed decisions.

### **ADVANTAGES**

- Provides fast prediction.
- Low-cost expense than old system.
- Provides user friendly interface.
- Reducing Time efficiency

**CHAPTER 4**

**SYSTEM REQUIREMENTS AND SPECIFICATION**

## **SYSTEM REQUIREMENTS AND SPECIFICATION**

### **MINIMUM HARDWARE REQUIREMENTS:**

Processor : Intel I3  
RAM : 8GB  
Hard Disk : 128 GB

### **MINIMUM SOFTWARE REQUIREMENTS:**

Front End : Html, CSS, Java Script  
Framework : Flask  
Programming Language : Python 3.7  
Platform : Visual Studio Code  
OS : Windows 7

## **FUNCTIONAL REQUIREMENTS:**

### **USER REGISTRATION:**

Users should be able to register with the application by providing their name, email, password, and contact number.

### **2. DATASET UPLOAD:**

Users should be able to upload a dataset in CSV format containing real estate property information.

### **3. DATASET SPLITTING:**

Users should be able to split the dataset into training and testing sets.

### **4. MODEL TRAINING AND EVALUTION:**

The application should support multiple regression algorithms, such as Linear Regression, Decision Tree Regressor, K-Nearest Neighbors Regressor.

### **5. HOUSE VALUATION:**

Users should be able to provide property features (e.g., number of bed rooms, area, etc.) to predict the house valuation using the trained model.

## **NON-FUNCTIONAL REQUIREMENTS:**

### **PERFORMANCE:**

The system should be capable of handling big datasets effectively, ensuring minimum processing time for training and prediction.

### **DATABASE SECURITY:**

Unauthorized users should not be able to access the data on this platform.

### **RELIABILITY:**

The system should provide a reliable environment for the user to ensure that predictions are consistent .

### **RESPONSE TIME:**

The information page will load in less than one minute and be refreshed every two minutes.

### **AVAILABILITY:**

The system will be available by 24/7.

**CHAPTER 5**  
**DATASET**

## **DATASET:**

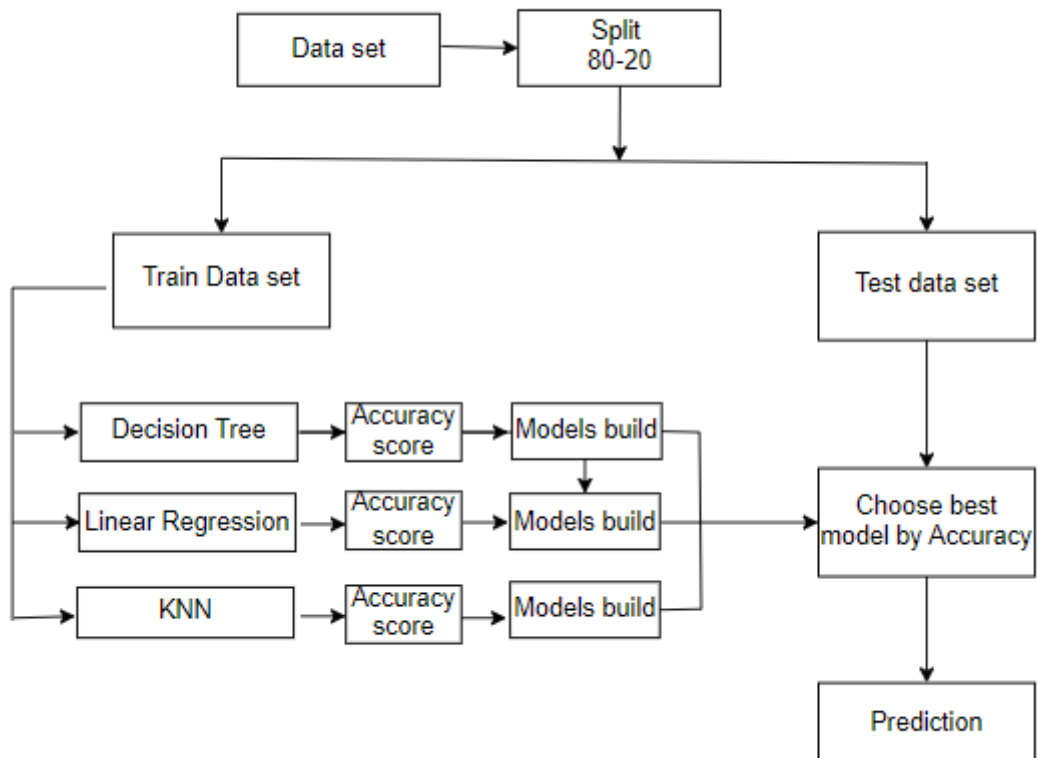
<b>S/N</b>	<b>bedrooms</b>	<b>bathrooms</b>	<b>sqft_living</b>	<b>floors</b>	<b>waterfront</b>	<b>view</b>	<b>lat</b>	<b>long</b>	<b>price</b>
1	3.0	1.0	1170.0	1.0	0.0	0.0	47.4627	-122.19	255000.0
2	3.0	1.0	1400.0	1.0	0.0	0.0	47.5066	-122.37700000000001	380000.0
3	2.0	1.0	860.0	1.0	0.0	0.0	47.4878	-122.324	200000.0
4	3.0	2.25	1430.0	2.0	0.0	0.0	47.5477	-122.38799999999999	384000.0
5	4.0	2.5	2210.0	2.0	0.0	0.0	47.3122	-122.039	353000.0
6	2.0	1.0	1500.0	1.0	0.0	0.0	47.661	-122.338	480000.0
7	3.0	2.5	2760.0	2.0	0.0	0.0	47.4929	-122.18799999999999	545000.0
8	3.0	1.0	1010.0	1.0	0.0	0.0	47.5249	-122.37	390000.0
9	4.0	1.75	1760.0	1.0	0.0	0.0	47.4267	-122.148	336500.0
10	4.0	1.75	1970.0	1.0	0.0	0.0	47.4861	-122.251	265000.0
11	4.0	1.75	2100.0	1.0	0.0	0.0	47.7478	-122.338	457000.0

**FIGURE 5.1 DATASET**

**CHAPTER 6**  
**SYSTEM DESIGN**

## SYSTEM DESIGN

### ARCHITECTURE DIAGRAM:



**Figure 6.1 ARCHITECTURE DIAGRAM**



## **UML DIAGRAMS:**

UML (Unified Modeling Language) diagrams are visual depictions used in software engineering and system development to show the structure, behavior, and relationships of a system.

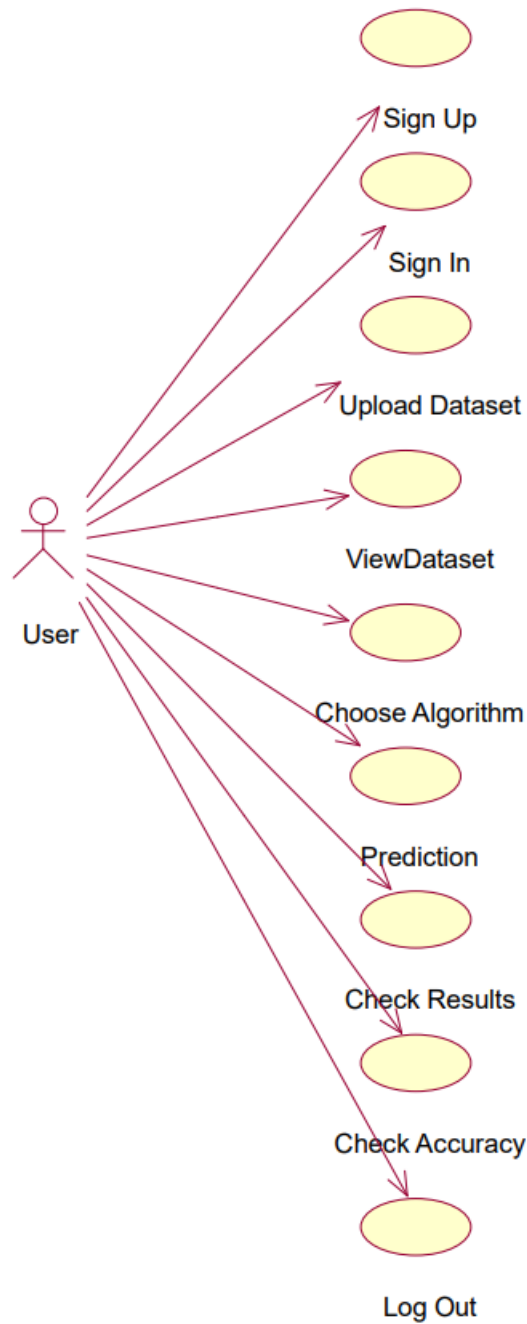
UML diagrams play a significant role in the software development lifecycle, assisting in analysis, design, implementation, and documentation, thereby boosting the clarity and comprehension of complex systems.

### **Features:**

- It is a language used for broad modelling.
- It differs from other programming languages like Python, C++, and others.
- It is related to analysis and design that is object-oriented.
- It is utilized to visualize the system's workflow.

## USE CASE DIAGRAM:

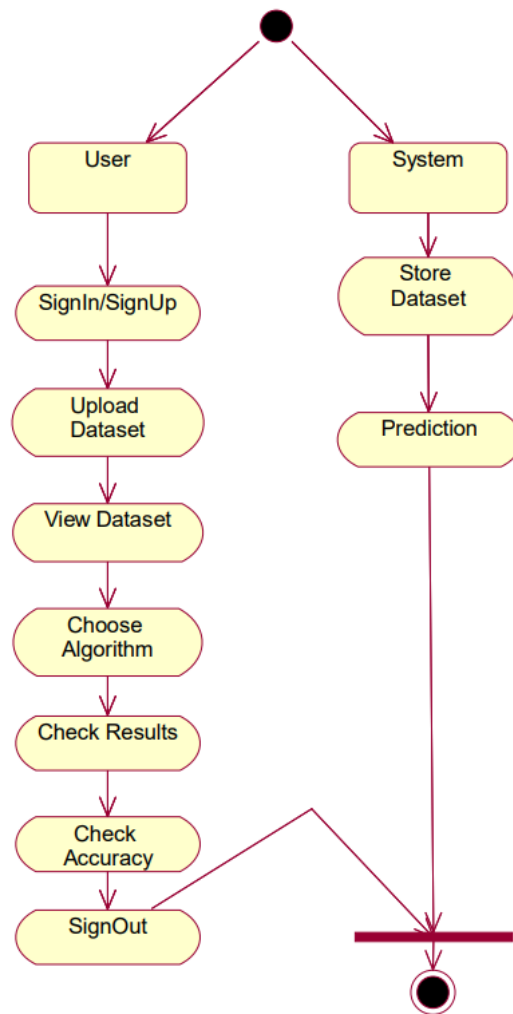
A use case diagram is used to graphically displays the functional needs of a system or software application from the viewpoint of its users or actors. It gives a high-level picture of the system's activity and interactions with its external components.



**Figure 6.2.1 Use Case Diagram**

## ACTIVITY DIAGRAM:

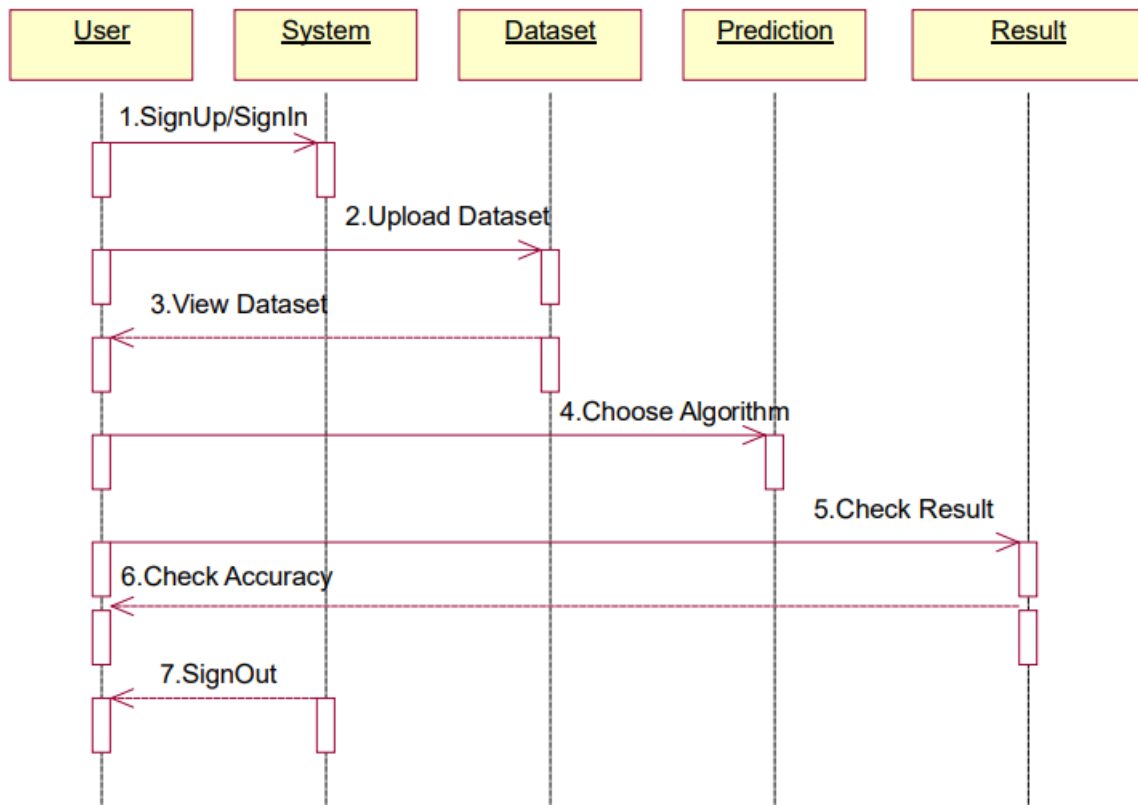
An activity diagram is used to visualize the flow of activities or processes inside a system or business process. It emphasizes on the workflow or behavioral elements of a system, capturing the sequence of events, choices, and concurrency involved. Activity diagrams consist of nodes, edges, actions, and control components.



**Figure 6.2.2 Activity Diagram**

## SEQUENCE DIAGRAM:

Sequence diagrams consist of lifelines, messages, and activation boxes. Lifelines indicate the items or components participating in the interaction, each portrayed as a vertical line. Messages are represented by arrows that link the lifelines, signifying the communication or connection between items.



**Figure 6.2.3 Sequence Diagram**

**CHAPTER 7**  
**IMPLEMENTATION**

## **IMPLEMENTATION**

### **SOFTWARE ENVIRONMENT :**

#### **PYTHON**

- Python is a high-level programming language noted for its simplicity and clarity.
- Python has a simple and legible syntax, which makes it easy to learn and comprehend. Its use of indentation as a technique to designate code chunks increases code uniformity and readability.
- Python supports numerous programming paradigms, including procedural, object-oriented, and functional programming.
- Google and Yahoo! both utilise Python a lot to make their Web crawlers and search engines work.

#### **FLASK**

- Flask offers a rapid debugger and an integrated development server.
- The documentation for Python Flask is vast.
- Additionally, it includes the integrated support essential for unit testing.
- Flask can also be installed in a virtual environment on Windows, Mac, and Linux.

#### **INSTALLATION:**

- ✓ pip install flask

### **TECHNOLOGIES USED:**

#### **PANDAS:**

- Pandas is a robust open-source data manipulation and analysis package for Python. Pandas provides data structures and techniques that make it simpler to work with structured data and more.

#### **NUMPY:**

- NumPy is a powerful Python library for numerical computations. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays efficiently.

#### **MATPLOTLIB:**

- Matplotlib is a popular data visualization package for Python. It includes a broad variety of functions and tools for making high-quality plots, charts, and figures.

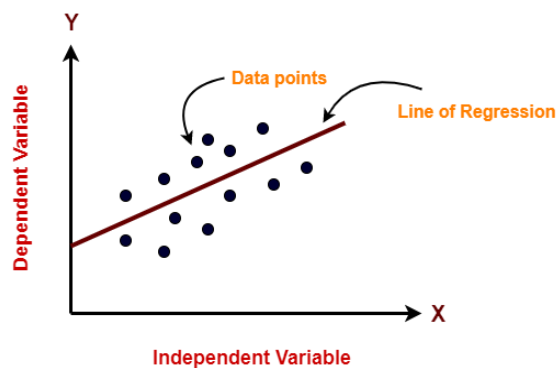
# ALGORITHMS

## **LINEAR REGRESSION**

Linear regression is a supervised learning algorithm used to analyze the relationship between a dependent variable and one or more independent variables. It aims to find the best-fitting linear equation that describes the linear relationship between the variables

### **STEPS :**

- Identify the dependent and independent variables.
- Calculate the slope and intercept.
- Fit the line to the data.
- Assess the model's goodness of fit.
- Make predictions using the regression equation.
- Evaluate the statistical significance of coefficients.



## **K-NEAREST NEIGHBOR (KNN):**

- K-Nearest Neighbors (KNN) is a machine learning technique used for classification and regression tasks.
- It functions by locating the k average data points in the training set to a given input sample.
- The method predicts the class or value of the input based on the majority vote (for classification) or the average (for regression) of the neighbors.
- KNN is a non-parametric method, which means it does not make any assumptions about the underlying data distribution and may function effectively with various datasets.

### **STEPS:**

- Choose the k-number of neighbors.
- Find the neighbors Euclidian distances using k.
- Determine the k nearest neighbors in accordance with the calculation.

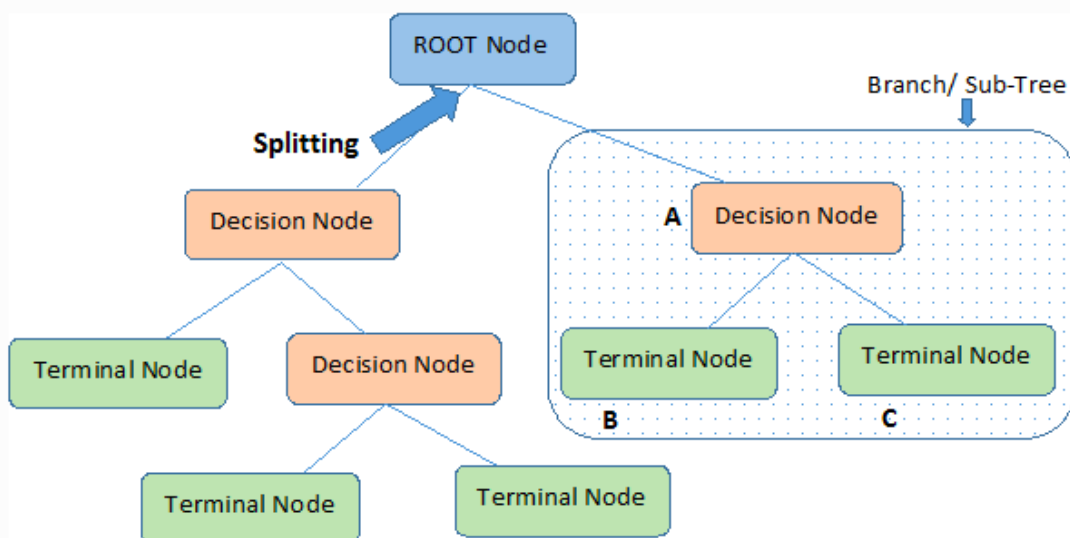
- Count the number of data points in each category among the k neighbors.
- The category with the greatest number of neighbors should receive the additional data points.
- Stop the operation.

### **Decision Tree:**

The supervised learning approach known as a decision tree may be applied to classify and regression issues, although it is most typically utilized to deal with classification challenges. It is a tree-structured classifier, with internal nodes standing in for dataset characteristics, branches for decision-making procedures, and leaf nodes for conclusions.

### **Steps to perform:**

- Identify the tree's root.
- Split the data based on select feature.
- Repeat the splitting process until a stopping condition is met.
- Set labels to every single leaf node.
- Prune the tree.
- Make predictions by traversing the tree from root to leaf node.
- Evaluate the model.



**Note:-** A is parent node of B and C.



# **CHAPTER 8**

## **SOURCE CODE**

### SAMPLE CODE :

```
webapp.py:
import pandas as pd
from flask import Flask, render_template, request, url_for, flash, redirect, session
import os
from sklearn.model_selection import train_test_split
from prediction import X, y
import pymysql
import matplotlib.pyplot as plt
import numpy as np
filepath = os.getcwd()
webapp = Flask(__name__)
db = pymysql.connect(host='localhost', user='root', password="",
db='house_price_prediction')
cursor = db.cursor()
webapp.config['UPLOAD_FOLDER'] = r"dataset"
@Webapp.route("/")
def main():
    return render_template("home.html")
@Webapp.route("/reg", methods=['POST', 'GET'])
def reg():
    if request.method == 'POST':
        Name = request.form['name']
@Webapp.route("/login", methods=['POST', 'GET'])
def login():
    if request.method == 'POST':
        Email = request.form["email"]
        pwd = request.form["pwd"]
```

```

@webapp.route("/upload", methods=['POST', 'GET'])
def upload():
    if request.method == 'POST':
        myfile = request.files['file']
        if ext.lower() == ".csv":
            @webapp.route("/View")
def View():
    myfile = os.listdir(webapp.config['UPLOAD_FOLDER'])
    global full_data
@webapp.route('/split', methods=['POST', 'GET'])
def split():
    if request.method == "POST":
        test_size = float(request.form['size'])
        global X_train, X_test, y_train, y_test
        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size,
@webapp.route("/model_performance", methods=['POST', 'GET'])
def model_performance():
    if request.method == "POST":
        model_no = int(request.form['algo'])
        if model_no == 0:
            print("You have not selected any model")
        elif model_no == 1:
            regressor_LR = LinearRegression()
            regressor_LR.fit(X_train, y_train)
            from sklearn.metrics import mean_squared_error, r2_score
@webapp.route('/prediction', methods=['POST', 'GET'])
def prediction():
    if request.method == 'POST':
        f1 = request.form['f1']

```

```

@webapp.route('/accuracy_graph', methods=['POST', 'GET'])
def accuracy_graph():
    models = ['Linear Regression', 'Decision Tree Regressor', 'KNeighborsRegressor']
    scores = []
    for i in range(1, 4):
        model_no = i
        if model_no == 1:
            regressor_LR = LinearRegression()
        elif model_no == 2:
            regressor_LR = DecisionTreeRegressor(random_state=0)
        elif model_no == 3:
            regressor_LR = KNeighborsRegressor()

        regressor_LR.fit(X_train, y_train)
        y_pred_lin = regressor_LR.predict(X_test)
        from sklearn.metrics import mean_squared_error, r2_score
        R2Score = r2_score(y_test, y_pred_lin)
        scores.append(R2Score)

    x = np.arange(len(models))
    width = 0.35
    colors = ['lightblue', 'lightgreen', 'lightcoral']
    fig, ax = plt.subplots()
    rects = ax.bar(x, scores, width, color=colors)

    ax.set_ylabel('R2 Score')
    ax.set_title('Accuracy Scores of Different Models')
    ax.set_xticks(x)
    ax.set_xticklabels(models)

```

## **HOME PAGE HTML CODE:**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Real estate valuation</title>
  <link href="static/lib/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <link href="static/css/style.css" rel="stylesheet">
</head>
<body>
  <nav class="navbar navbar-default navbar-trans navbar-expand-lg fixed-top">
    <div class="navbar-collapse collapse justify-content-center" id="navbarDefault">
      <ul class="navbar-nav">
        <li class="nav-item">
          <a class="nav-link active" href="/">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="{{ url_for('reg') }}">Register</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="{{ url_for('login') }}">Login</a>
        </li>
      </ul>
    </div>
    <button type="button" class="btn btn-b-n navbar-toggle-box-collapse d-none d-md-block" data-
toggle="collapse"
      data-target="#navbarTogglerDemo01" aria-expanded="false">
    </button>
  </nav>
</body>
</html>
```

## CSS CODE:

```
*
{
  margin:0;
  padding:0;
  font-family:'poppins',sans-serif;
  box-sizing:border-box;
}

.container
{
  width:100%;
  height:100vh;
  background-image:linear-gradient(rgba(0,0,50,0.8),rgba(0,0,50,0.8)),url(index.jpg);
  background-position: center;
  background-size:cover;
  position: relative;
}

.container h1{
  color:whitesmoke;
  padding-top: px;
  text-align: center;
  font-family: Cambria, Cochin, Georgia, Times, 'Times New Roman', serif;
}

.content p{
  color:whitesmoke;
  font-size:25px;
  padding-top: px;
  text-align: center;
  font-family:'poppins',sans-serif;
  color:#FFFF00;
}

header, footer {
```

```
position: relative;
    padding: 2em 3em;
    display: flex;
    align-items: center;
    font-size: 1rem;
}

header {
    position: -webkit-sticky;
    position: sticky;
    top: 0;
    z-index: 2;
    height: 10vh;
    background-color: transparent;
}

header h3 {
    position: relative;
    margin: 5;
    font-size: 2rem;
    color: #FFFF;
}
```

**CHAPTER 9**  
**OUTPUT SCREENS**



**HOME PAGE:**

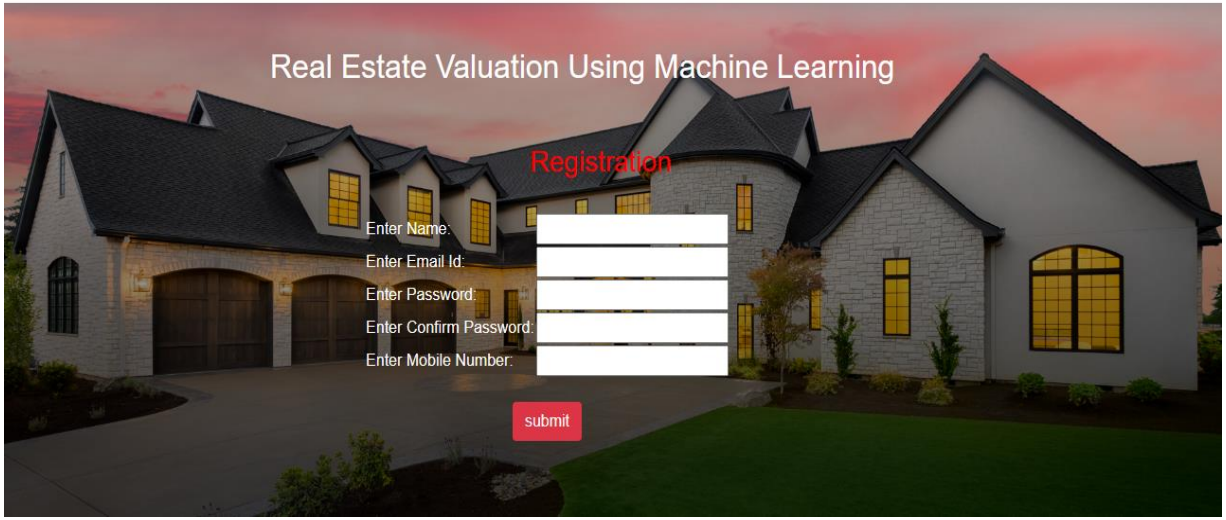
[Home](#) [Register](#) [Login](#)



**FIGURE 9.1.1 HOME PAGE**

**REGISTRATION PAGE :**

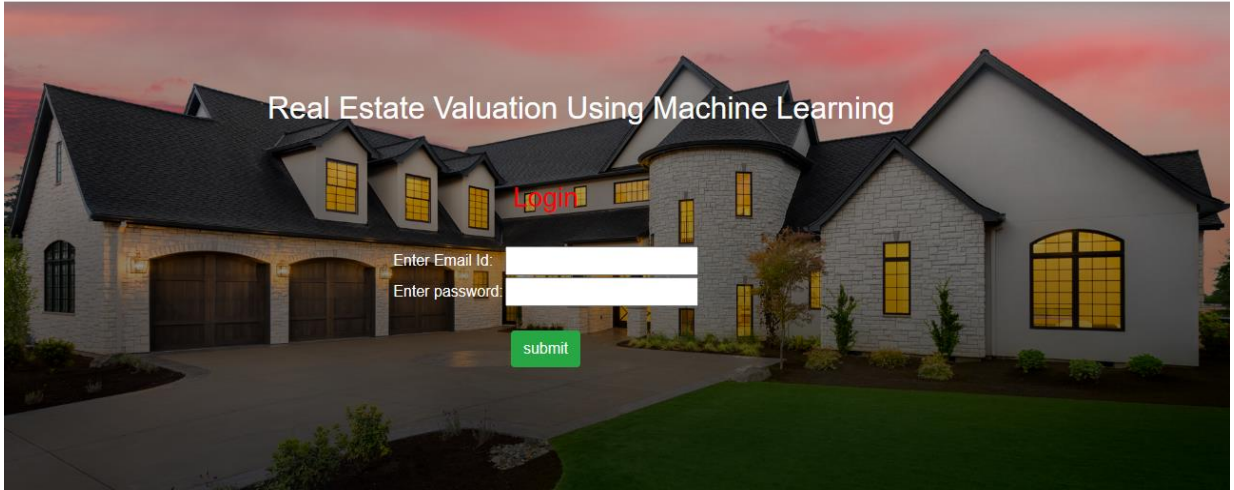
Home [Register](#) Login



**FIGURE 9.1.2 REGISTRATION PAGE**

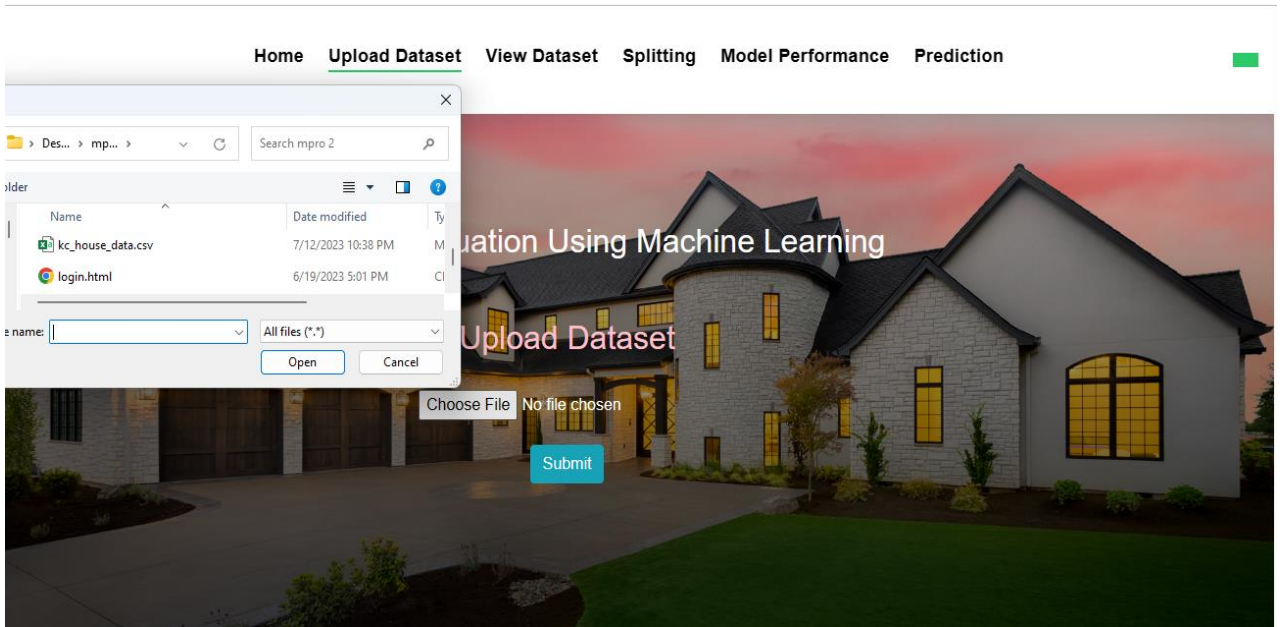
**LOGIN PAGE :**

[Home](#) [Register](#) [Login](#)



**FIGURE 9.1.3 LOGIN PAGE**

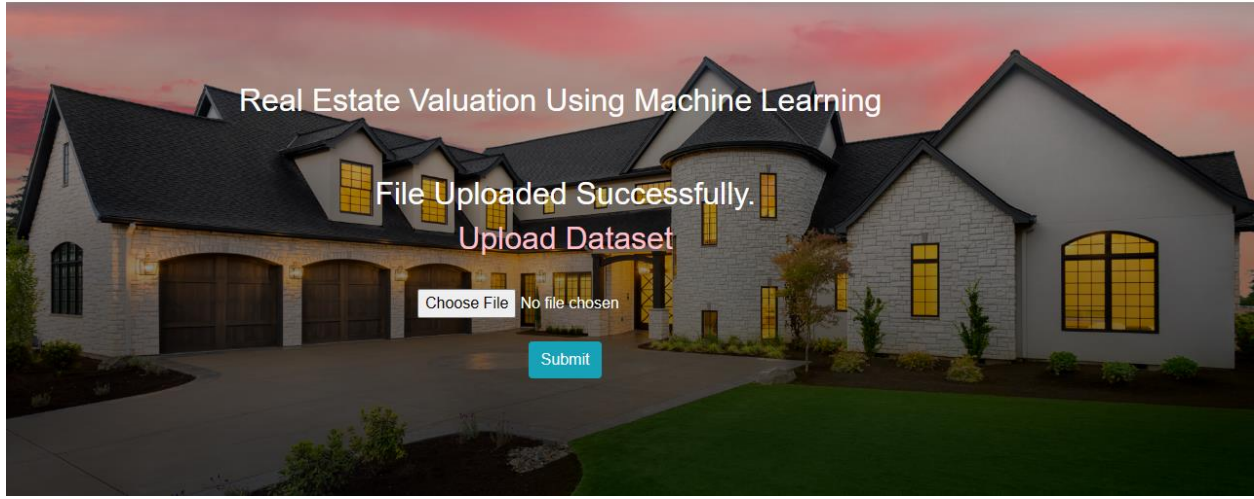
**UPLOAD DATASET PAGE :**



**FIGURE 9.1.4 UPLOAD DATASET PAGE**

**DATASET UPLOADED PAGE:**

Home Upload Dataset View Dataset Splitting Model Performance Prediction



**FIGURE 9.1.5 DATASET UPLOADED PAGE**

## DATA VIEW PAGE:

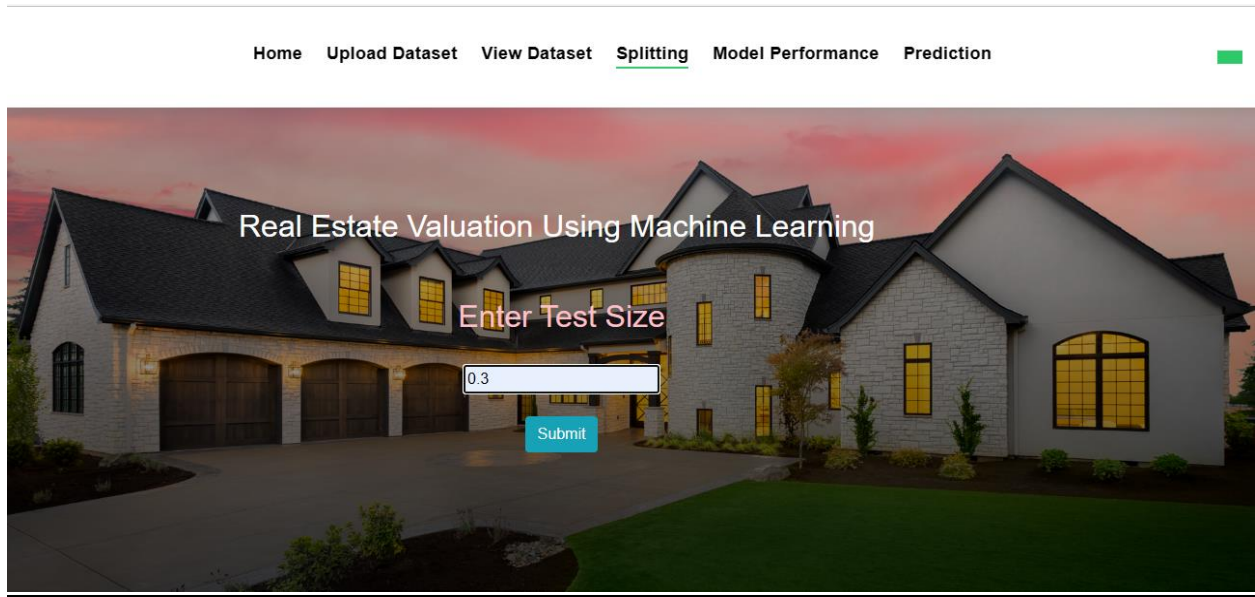
[Home](#) [Upload Dataset](#) [View Dataset](#) [Splitting](#) [Model Performance](#) [Prediction](#)

### Real Estate Valuation Using Machine Learning

S/N	bedrooms	bathrooms	sqft_living	floors	waterfront	view	lat	long	price
1	5.0	4.0	3460.0	2.0	0.0	0.0	47.5201	-122.204	980000.0
2	3.0	2.0	1680.0	1.5	0.0	0.0	47.2775	-122.20299999999999	219200.0
3	2.0	1.0	870.0	1.0	0.0	0.0	47.5465	-122.384	379950.0
4	2.0	2.0	1430.0	1.0	0.0	0.0	47.6844	-122.39200000000001	545000.0
5	2.0	1.5	1800.0	1.0	0.0	2.0	47.6305	-122.34700000000001	930000.0
6	4.0	1.75	1900.0	1.0	0.0	0.0	47.6579	-122.197	675000.0
7	5.0	2.5	4670.0	2.0	0.0	0.0	47.63	-122.01100000000001	1000000.0
8	3.0	1.75	920.0	1.0	0.0	0.0	47.6386	-122.365	589000.0
9	3.0	1.75	1400.0	1.0	0.0	0.0	47.7312	-122.20200000000001	335000.0
10	3.0	1.0	1250.0	1.0	0.0	0.0	47.7622	-122.163	350000.0
11	3.0	2.0	1010.0	1.0	0.0	0.0	47.6223	-122.04299999999999	354000.0

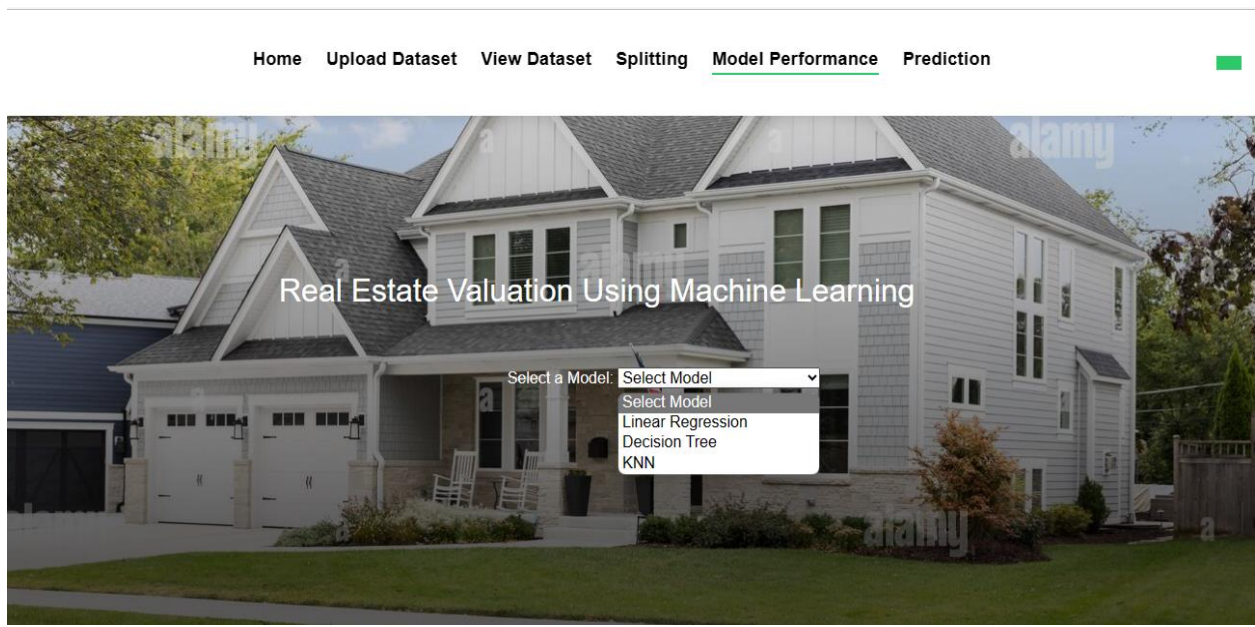
**FIGURE 9.1.6 DATA VIEW PAGE**

**SPLIT PAGE:**



**FIGURE 9.1.7 DATA SPLIT PAGE**

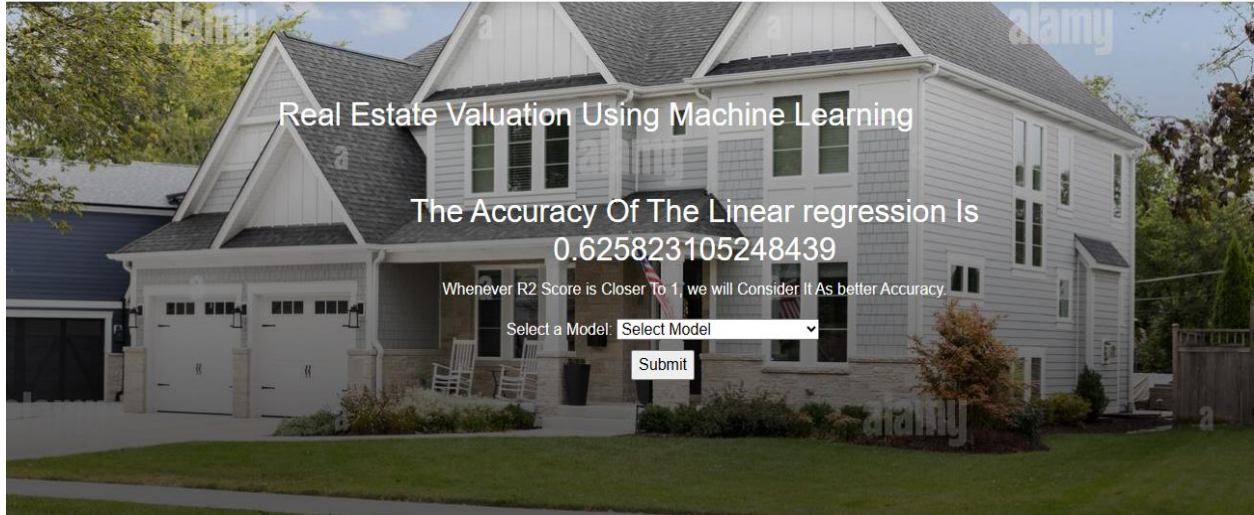
**MODEL SELECTION PAGE:**



**FIGURE 9.1.8 MODEL SELECTION PAGE**



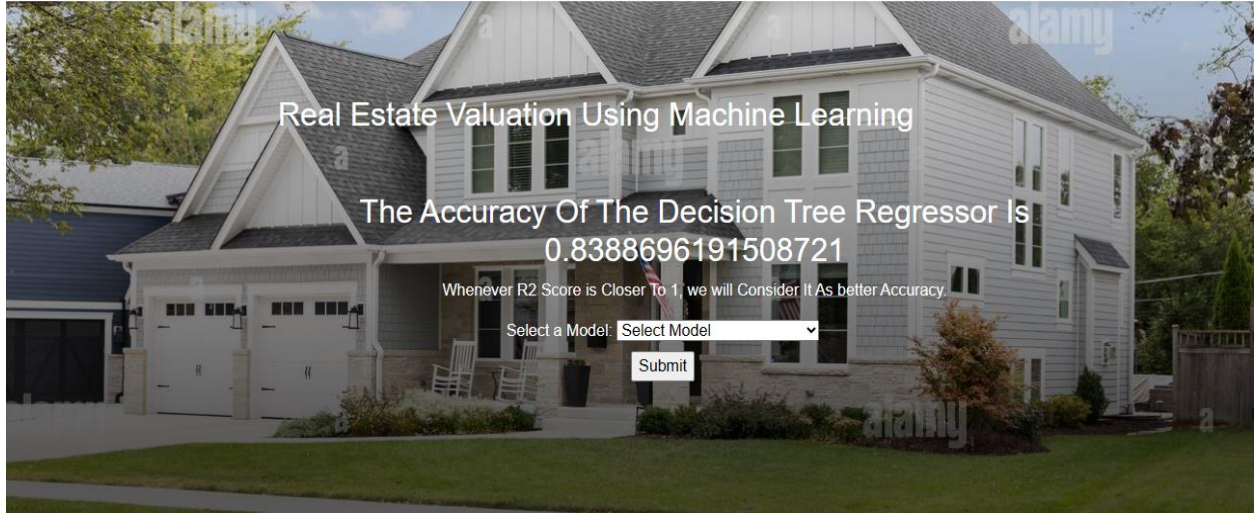
**ACCURACY PAGE:**



**FIGURE 9.1.9 LINEAR REGRESSION ACCURACY PAGE**

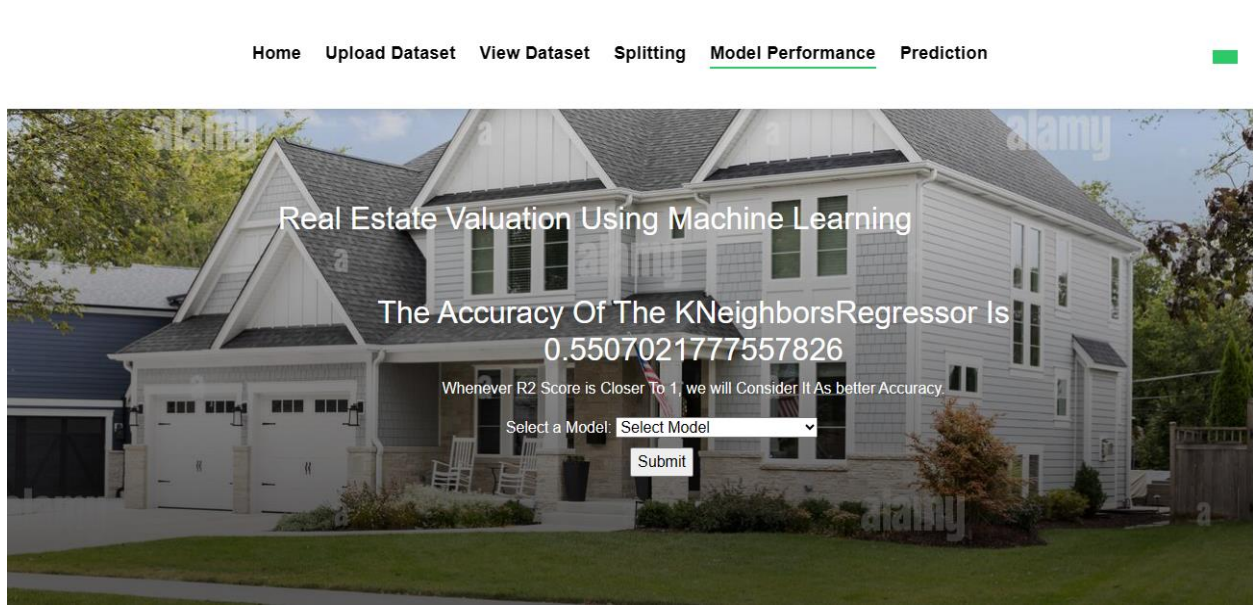
**ACCURACY PAGE :**

Home Upload Dataset View Dataset Splitting Model Performance Prediction



**FIGURE 9.2.0 DECISION TREE ACCURACY PAGE**

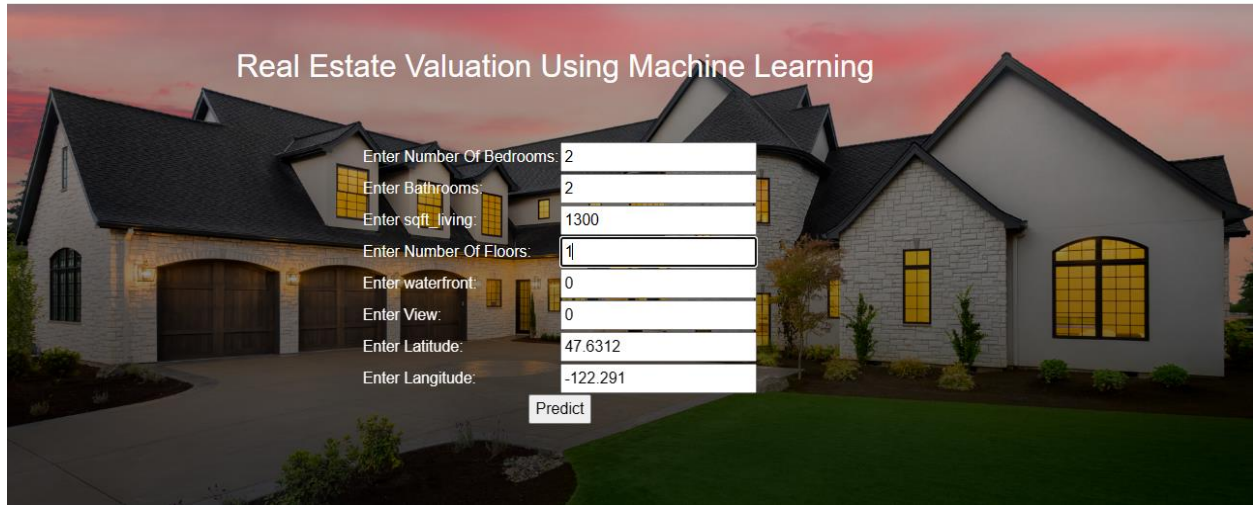
**ACCURACY PAGE :**



**FIGURE 9.2.1 KNN ACCURACY PAGE**

## INPUT PAGE :

[Home](#) [Upload Dataset](#) [View Dataset](#) [Splitting](#) [Model Performance](#) [Prediction](#) [Accuracy](#) [Logout](#) ■



Real Estate Valuation Using Machine Learning

Enter Number Of Bedrooms: 2

Enter Bathrooms: 2

Enter sqft\_living: 1300

Enter Number Of Floors: 1

Enter waterfront: 0

Enter View: 0

Enter Latitude: 47.6312

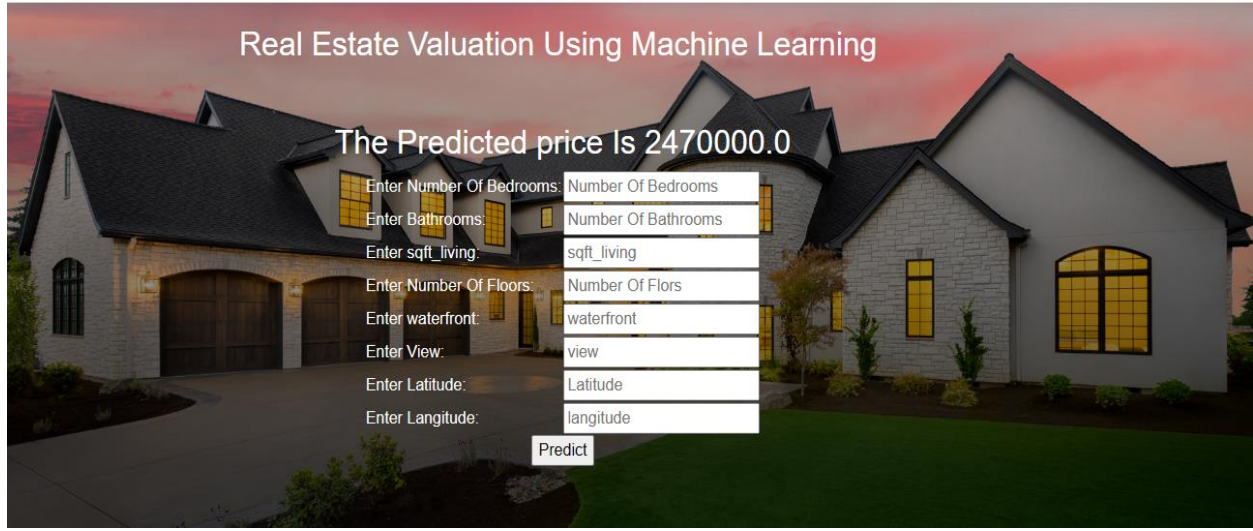
Enter Longitude: -122.291

Predict

**FIGURE 9.2.2 INPUT PAGE**

## RESULT PAGE :

[Home](#) [Upload Dataset](#) [View Dataset](#) [Splitting](#) [Model Performance](#) [Prediction](#) [Accuracy](#) [Logout](#) ■



Real Estate Valuation Using Machine Learning

The Predicted price Is 2470000.0

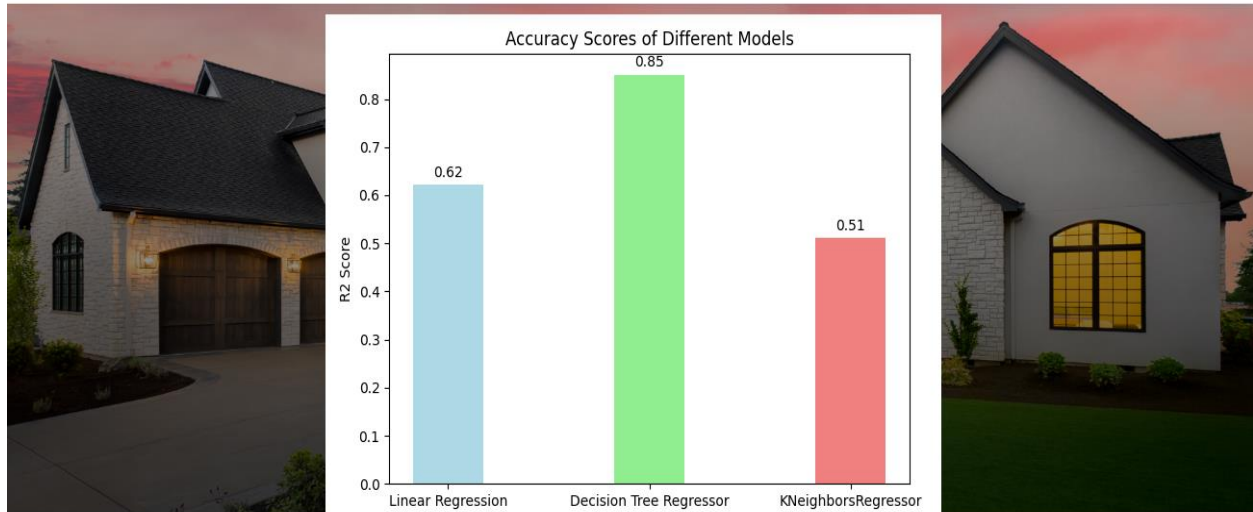
Enter Number Of Bedrooms:	Number Of Bedrooms
Enter Bathrooms:	Number Of Bathrooms
Enter sqft_living:	sqft_living
Enter Number Of Floors:	Number Of Flors
Enter waterfront:	waterfront
Enter View:	view
Enter Latitude:	Latitude
Enter Langitude:	langitude

Predict

**FIGURE 9.2.3 PREDICTION PAGE**

## ACCURACY COMPARISION PAGE :

[Home](#) [Upload Dataset](#) [View Dataset](#) [Splitting](#) [Model Performance](#) [Prediction](#) [Accuracy](#) [logout](#)



**FIGURE 9.2.4 GRAPH PAGE**

**CHAPTER 10**  
**TESTING**

### **SYSTEM TESTING:**

System testing ensures that the entire software system complies with the requirements. It evaluates a configuration to ensure that the consequences are understood and foreseeable. An illustration of a system test is the configuration-based system integration test. System testing is based on process flows and models, with a focus on links and interaction points that have previously been established.

### **UNIT TESTING:**

Making test cases as part of unit testing entails confirming that the program's basic logic is solid and that genuine inputs result in valid outputs. The program's decision paths, and code flow should both be reviewed. Every piece of software that goes into a program is tested in this approach. It is finished after the completion of each component but before they are joined. This is an invasive test of the structure that depends on knowing how it was created. Unit tests execute rapid tests at the level of individual pieces and assess a single business method, software program, or system configuration. Unit tests guarantee that each individual route of a business process runs exactly as stated in the documentation and has explicit inputs.

### **INTEGRATION TESTING:**

Integration tests are intended to check whether a program's component pieces perform as a single program. Testing is focused on events and is generally concerned with the functioning of screens or fields. Even though each component functioned brilliantly on its own, as evidenced by competent unit testing, integration tests showed that the components are coherent and consistent. Integration testing is aimed to uncover problems that appear when numerous components are integrated together.

### **BLACK BOX TESTING:**

Black box testing includes analyzing software without having any knowledge of how it performs, how it is constructed, or what language it was written in. Black box tests need to be built from a clear source document, like a blueprint or standards document, much like most other kinds of tests.



# **CHAPTER 11**

## **TEST CASES**

**TEST CASES:**

S.NO	Test Cases	Input	Expected Output	Actual Output	Pass/Fail
1	If Login details are not filled.	A warning statement is to be displayed.	Users cannot get access to the next page.	User cannot get access to the next page.	Pass
2	If user enter the existing email while signup	Passing user email id.	The output should be already Email id exists	Already email id exists	Pass.
3	If user choose KNN algorithm	Passing type of house details.	To predict the price of the house.	It predicts house price.	Pass.
4	Dataset is uploaded or not.	Upload dataset.	Dataset is uploaded.	Dataset is uploaded.	Pass

**CHAPTER 12**  
**CONCLUSION**

## **CONCLUSION:**

This system mainly concentrates on the comparison between different machine learning algorithms (Linear regression, Decision Tree, KNN) about Real estate valuation Analysis. From the system results, Decision Tree algorithm has high accuracy value when compared to all the other algorithms regarding real estate valuation. Here the [MSE] Mean Square Error and R2 score are used to calculate the accuracy value of the algorithm on the King County Dataset which was collected from Kaggle website. They can be extended by applying the above said algorithms to predict House value.

**CHAPTER 13**  
**REFERENCES**

## REFERENCES

- [1]. Aminah Md Yusof and Syuhaida Ismail ,Multiple Regressions in Analysing House Price Variations. IBIMA Publishing Communications of the IBIMA Vol.2012 (2012), Article ID 383101, 9 pages DOI: 10.5171/2012.383101.
- [2]. Babyak, M. A. What you see may not be what you get: A brief, nontechnical introduction to over fitting regression-type models. Psychosomatic Medicine, 66(3), 411-421.
- [3]. Atharva chogle, priyanka khair, Akshata gaud, Jinal Jain. House Price Forecasting using Data Mining Techniques International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified Vol. 6, Issue 12, December 2017
- [4]. Darshan Sangani, Kelby Erickson, and Mohammad Al Hasan, Predicting Zillow Estimation Error Using SVR and Gradient Boosting, IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Page(s):530 - 534
- [5]. Model, Azme Bin Khamis, Nur Khalidah Khalilah Binti Kamarudin ,Comparative Study On Estimate House Price Using Statistical And Neural Network, International journal of scientific and technology, research volume 3, ISSUE 12, December 2014,Page(s):126-131.
- [6]. Adyan Nur Alfiyatin, Hilman Taufiq, Ruth Ema Febrita, Wayan Firdaus Mahmudy,Modeling House Price Prediction using Regression Analysis and Particle Swarm Optimization Case Study: Malang, East Java, Indonesia.

[7]. Aminah Md Yusof and Syuhaida Ismail ,Multiple Regressions in Analysing House Price Variations. IBIMA Publishing Communications of the IBIMA Vol. 2012 (2012), Article ID 383101, 9 pages DOI: 10.5171/2012.383101.

[8]. Li, J., Liu, C., Zhang, H."XGBoost for Real Estate Valuation: A Comparative Study".International Journal of Geographical Information Science, 2020.

[9]. Title: "Geospatial Analysis for Real Estate Valuation: A Review"Authors: Garcia, N., Rodriguez, E., Martinez, A. Journal: International Journal of Geographical Information Science, 2022

[10]. Title: "Deep Learning Models for Real Estate Valuation: An Empirical Investigation" Authors: Chen, T., Liu, Y., Wang, H. Journal: Neural Computing and Applications, 2021