# HOUSE PRICE PREDICTOR

*A project Submitted*
*in*

*Partial fulfilment for the award of the degree*
*of*

**Master of Computer Application**

*under*
**Cochin University Of Science and Technology**
*by*

**ALDRIN MARTIN SIMENTHY**
Reg. No. 38119008

**GREESHMA PRASAD P**
Reg. No. 38119025

**JYOTHIS K S**
Reg. No. 38119029

**VISMAYA DINESH**
Reg. No. 38119056

*under the guidance of*
**Mr. VIVEK M V**

**DEPARTMENT OF COMPUTER APPLICATIONS**
**COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY**
**COCHIN-22, KERALA**

**10th MAY 2021**

**DEPARTMENT OF COMPUTER APPLICATIONS**

**COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**COCHIN-22**

## CERTIFICATE

*This is to certify that the report entitled **"HOUSE PRICE PREDICTOR"** submitted in partial fulfilment of the requirements for the award of the Degree of Master of Computer Application is a record of bonafide work done by **ALDRIN MARTIN SIMENTHY, GREESHMA PRASAD P , JYOTHIS K S, VISMAYA DINESH** during the period from February 2021 to May 2021 of her study in the Department of Computer Application at Cochin University Of Science and Technology, Cochin under my supervision and guidance.*

**Signature of Examiner**
Dept. Of Computer Applications
CUSAT

**Head of the Department**
Dept. Of Computer Applications
CUSAT

DEPARTMENT OF COMPUTER APPLICATIONS
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY
COCHIN-22

## CERTIFICATE

*This is to certify that the report entitled* ***"HOUSE PRICE PREDICTOR"*** *submitted in partial fulfilment of the requirements for the award of the Degree of Master of Computer Application is a record of bonafide work done by* ***ALDRIN MARTIN SIMENTHY, GREESHMA PRASAD P , JYOTHIS K S, VISMAYA DINESH*** *during the period from February 2021 to May 2021 of her study in the Department of Computer Application at Cochin University Of Science and Technology, Cochin under my supervision and guidance.*

**Guide: Mr. VIVEK M V**
Department Of Computer Applications
CUSAT

# ACKNOWLEDGEMENT

With great pleasure I hereby acknowledge that the help given to our team by various individuals throughout the project and this project itself is an acknowledgement to the inspiration driven and technical assistance contributed by many individuals.

We wish to express our sincere gratitude to **Dr. K.N MADHUSOODANAN**, Vice-Chancellor of Cochin University Of Science and Technology, Cochin.

I extend our sincere thanks to all the non-teaching staff for providing the necessary facilities and help. I thank Lord God, the almighty for his immeasurable blessing upon my life.

I am pleased my indebtedness to **Dr.M.K SABU**, Head of the Department, Department of Computer Application, CUSAT for his gracious encouragement.

I am also thankful to **Mr. VIVEK M V**, my guide for her support .We are obliged to the teaching staff for being helpful and co-operative during the period of project. I extend my heartfelt thankful to parents, friends and well wishes for their support and timely help.

**Date : 10-05-2021**

**ALDRIN MARTIN SIMENTHY**
**GREESHMA PRASAD P**
**JYOTHIS K S**
**VISMAYA DINESH**

# DECLARATION

I hereby declare that the project work entitled with **"HOUSE PRICE PREDICTOR"** is submitted in the partial fulfillment of the requirements for the IVth semester, Master of Computer Application and it is a report of the original work done by us in the Department Of Computer Applications, Cochin University of Science And Technology, Cochin.

**Date : 10-05-2021**

**ALDRIN MARTIN SIMENTHY**
**GREESHMA PRASAD P**
**JYOTHIS K S**
**VISMAYA DINESH**

# ABSTRACT

Nowadays, everything is changing its mode of operation to online. likewise, a single five-inch device can help you with everything you need in day-today life. But still real estate is not fully operate in online mode.

Nowadays, predicting the price of a House/Property is little much difficult without moving towards a broker. To solve this issue, we are introducing a new android and web application for predicting starting price for a property based on the geographical variables, numbers of rooms or facilities, no. of bathrooms, carpet area, built-up area, the floor, age of the property, zip code, latitude and longitude of the property and availability of schools, transport facilities, shopping facilities, medical facilities etc.. in that area and other developments in that area. By breaking down past market patterns and value ranges, and coming advancements future costs will be anticipated. This application helps to predict house prices with different machine learning models. It will help clients to put resources into a bequest without moving towards a broker.

This application is one of the prime fields to apply the ideas of machine learning on how to enhance and foresee the costs with high accuracy. The objective of this application is the prediction of the market value of a real estate property.

This app will be developed by using the machine learning and user interface is in Android and also in web application. Flutter framework (Dart language) is used to develop an android application and Flask (a python framework)is used for a web application.

# Contents

# List of Figures

# Chapter 1

# Introduction

## 1.1 Overview

Every single organization in today's real estate business is operating fruitfully to achieve a competitive edge over alternative competitors. There is a need to simplify the process for a normal human being while providing the best results. This paper proposes a system that predicts house prices using a regression machine learning algorithm. In case you're going to sell a house, you have to recognize what sticker price to put on it. This regression model is built not only for predicting the price of the house which is ready for sale but also for houses that are under construction.

Housing prices are an important reflection of the economy, and housing price ranges are of great interest for both buyers and sellers. In this project. house prices will be predicted given explanatory variables that cover many aspects of residential houses, such as physical features of the house, age of the property, educational facilities, medical facilities, transportation facilities, etc. As continuous house prices, they will be predicted with various regression techniques including Linear Regression, Decision Tree Regressor, Lasso Regression, Ridge Regression, Elastic Net Regressor, XG Boost, Extra Tree Regressor, SV regressor, Gradient Boosting, and Light GBM.

To make final predictions, we blended above models predictions together to get more robust predictions...The goal of this project is to create a regression model and a classification model that are able to accurately estimate the price of the house given the features. Our models are evaluated on the Root-Mean-Squared-Error (RMSE).The result of this research proved gives an accuracy of more than 85%.

After building the model and successfully giving the result, the next step is to do the integration with the UI. Develop the UI of this model in Android application and also in Web Application. Kivy framework is used to develop android application and Flask framework is used to develop a web application.

## 1.2 Purpose

Traditional house price prediction is based on cost and sale price comparison lacking of an accepted standard and a certification process. Therefore, the availability of a house price prediction model helps fill up an important information gap and improve the efficiency of the real estate market. In this Modern era, it becomes a risky factor to contact a broker directly. Its just creates a middleman and increases the cost of houses, so a virtual platform to perform this procedure becomes necessary.

By using various classifications done using the aspects of the property and the regression methods used we are provided with the accurate price prediction of the property mentioned. House prices increase every year, so there is a need for a system to predict house prices in the future. House price prediction can help the developer determine the selling price of a house and can help the customer to arrange the right time to purchase a house. Therefore the relevance of our project stand out in this social situation.

## 1.3 Scope

Housing prices is an important reflection of the economy, the scope of housing prices is a major concern for both the buyer and the seller. With more and more demand for housing in metropolitan cities, there is a definite increase in the number of private builders that provide real estate with additional amenities to attract more customers. Different factors considered for predicting the house prices are median Income, Crime Rate, Public schools, Hospitals, features of the house and developments nearby its location.

In this pandemic era it becomes inconvenient to ontact a broker directly, so a virtual platform to perform this procedure becomes necessary. Changes occurring in our lifestyle also are reflected in various aspects of our traditional methods. As a development in the structure of our long established methods we can convert the existing system with the online system. So there is no doubts in the scope of our proposed system in the future.

## 1.4 Sources

We collected data for Cities real estate properties from various real estate websites. The data would be having attributes such as Location, carpet area, built-up area, age of the property, zip code, etc. We must collect the quantitative data which is structured and categorized.

Our project is a machine learning app, based on certain specifications of your future home , it will try to predict the most accurate price. Our project focus will be predicting the house prices in Kochi, Chennai, Bangalore, Mumbai, Delhi and Kolkata. According to the city we prefer to buy the property we can consider the aspects aforementioned to predict the house price in the preferred place. As the city changes the features of the aspects also changes, this is reflected in the predicted price of the property. We use the characteristics of each city as the sources to predict the price of the house.

# Chapter 2

# System Analysis

## 2.1 Existing system

In the present situation, the customer visits a real estate agent so that he/she can suggest suitable show places for his investments. But the above method is risky as the agent may forecast wrong prices to the customer and that will lead to loss of customer's investment. This manual technique which is currently used in the market is outdated and has a high risk. Some risky factors are mentioned below:

### Disadvantages

- It is not possible to predict the future prices of homes mentioned by the customer. Because of this, investing in an apartment increases the risk or part of it significantly.
- Buyers are generally not aware of factors that influence the house prices.
- To contact a broker directly. Its just creates a middleman and increases the cost of houses

So as to overcome these drawback, we a need for an updated and automated system.

## 2.2 Proposed system

The world is shifting from manual to automated systems. The objective of our project is to reduce the problems faced by the customer. Proposed work aims at predicting the availability of houses based on different features of the houses and also the facilities available nearby the location of the houses. Work also includes the price prediction of the houses based on the features of the house and facilities nearby its location.

A real time dataset is prepared by analysing the major cities in India. The dataset contains the following features of the houses such as Number of bedrooms, age of the house, transport facility, schools available in the nearby location and shopping facilities.

The proposed method helps to search houses in big cities based on the following attributes.

1. Number of bedrooms (1BHK, 2BHK and 3BHK etc..).
2. Number of bathrooms, carpet area, built-up area, the floor etc..
3. Age of the property.
4. Education facilities.
5. Medical facilities.
6. Transport facility such as availability of bus, trains and flights.
7. Shopping facility such as small markets, general stores, shopping malls.
8. Analyse the crime rate.

The proposed work is implemented using Scikit Learn, a machine learning tool.

### Advantages

- Saves time and energy.
- Easy to access the system anywhere and anytime.
- Its helps to avoid middleman and decrease to cost od house.
- This system can be used by multiple people to get the report.

## 2.3 Feasibility study

Feasibility study is concerned to select the best system that meets performance requirements. These entities are an identification description, an evaluation of the candidate system for the job. Feasibility analysis evaluates the candidate systems and determines the best system that gives performance requirements. The purpose of the feasibility study is to investigate the present system, evaluate the cost and effectiveness of the proposed system, evaluate impact of the proposed system on existing personnel and assert the need for new personnel. Feasibility is carried for it the proposed system is technically, economically and operationally feasible. To determine the feasibility of this project, various areas such as technical, economical and behavioural feasibility are considered during the feasibility study, so feasibility is a test of a system proposed according to workability, impact on the organization ability to meet users, and effective use of resources.

Following are the feasibility studies employed in various areas as mentioned above.

1. Economical feasibility
2. Technical feasibility
3. Behavioural feasibility

### 2.3.1 Economical Feasibility

Economic analysis is the most frequently used method for evaluating the effectiveness of the candidate system, this study is carried out to check the economic impact of the system will have on the organization. Economic feasibility is the most frequently used method to evaluate the effectiveness of the candidate system which is known as cost beneficial analysis. The actual cost concerned with the manual labour exceeds the cost required, when the overall installation and maintenance procedure of the proposed system costs less than the above cost.

### 2.3.2 Technical Feasibility

It determines whether a computer solution exists for the problem. My project is compatible with any configuration even though the technologies keep fading at a fast pace. It holds all the technical guarantee of accuracy, reliability, ease of access and data security. If required, the proposed project can be upgraded. The proposed system has been verified with varying input and output speeds that has been estimated to be achieved and the quality of screen interface are taken care of in a smooth way that any user can understand from its basic and simple outlook.

### 2.3.3 Operational Feasibility

The level of acceptance by the user solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. User level of confidence must be raised so that the user is also able to make some constructive criticism, which is welcomed as that user is the final user of the system. An estimate should be made of how strong a reaction the user staff is likely to have toward the development of a computerized system. Its feasibility is concerned with human, organizational and political aspects. The system doesn't critically affect the nature and scope of the eventual recommendations at all. Users don't need any special skills or knowledge to use the proposed system since it is fully user friendly. All the operational aspects regarding this project are considered carefully and thus accomplished operational feasibility.

# Chapter 3

# Methodology

The below passages describe about the methodology used in the real estate house price predictions and the architecture flow diagram is given.

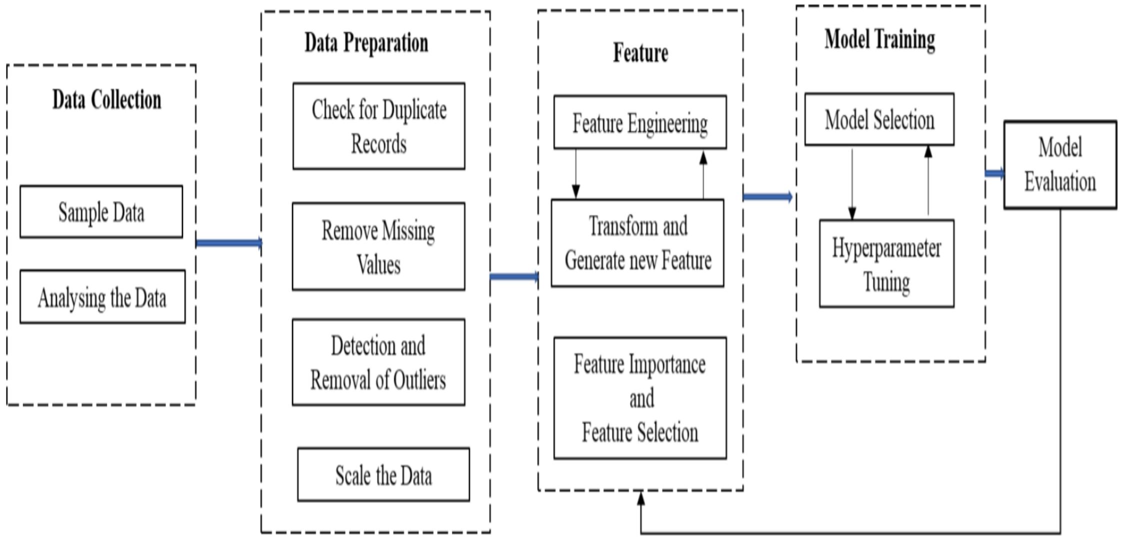


Figure 3.1: Research framework for the housing price problem.

## 3.1 Data Processing

### Phase 1: Collection of data

Data processing techniques and processes are numerous. We collected data for Kochi's real estate properties from various real estate websites. The data would be having attributes such as Location, carpet area, built-up area, age of the property, zip code, etc. We must collect the quantitative data which is structured and categorized. Data collection is needed before any kind of machine learning research is carried out. Dataset validity is a must otherwise there is no point in analysing the data.

### Phase 2: Data preprocessing

Data preprocessing is the process of cleaning our data set. There might be missing values or outliers in the dataset. These can be handled by data cleaning. If there are many missing values in a variable we will drop those values or substitute it with the average value.

### Phase 3: Training the model

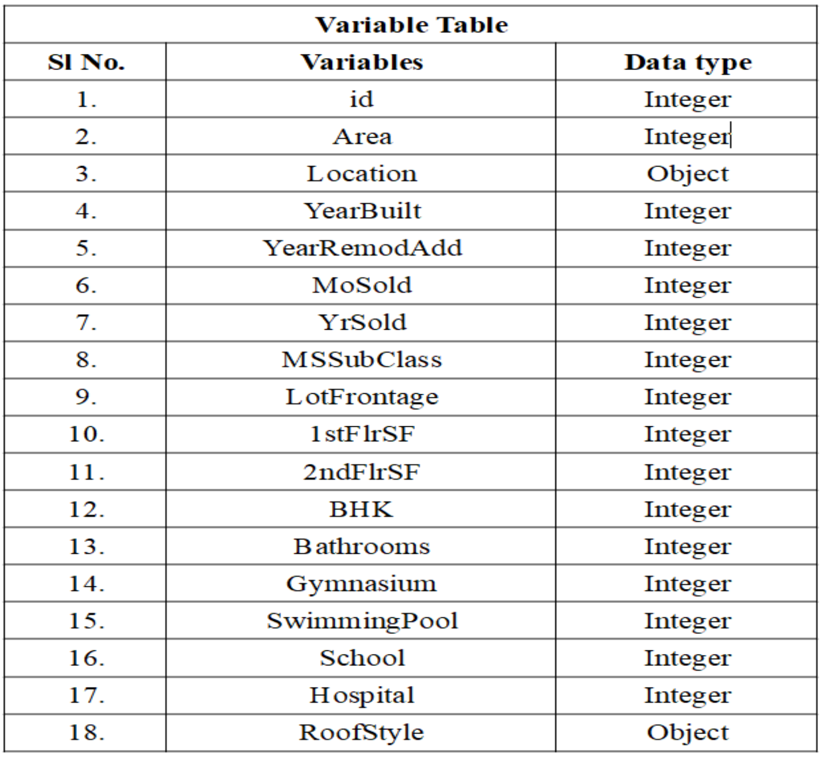
Since the data is broken down into two modules: a Training set and Test set, we must initially train the model. The training set includes the target variable. The decision tree regressor algorithm is applied to the training data set. The Decision tree builds a regression model in the form of a tree structure.

### Phase 4: Testing and Integrating with UI

The trained model is applied to test dataset and house prices are predicted. The trained model is then integrated with the front end using Kivy and Flask in python.

## 3.2 Description of Data-sets

The real estate housing data is used in this and it is taken from the machine learning repository and the data is spread across 3000 rows and has the more than 30 attributes the description of the data set is given below:



**Variable Table**

| Sl No. | Variables | Data type |
|---|---|---|
| 1. | id | Integer |
| 2. | Area | Integer |
| 3. | Location | Object |
| 4. | YearBuilt | Integer |
| 5. | YearRemodAdd | Integer |
| 6. | MoSold | Integer |
| 7. | YrSold | Integer |
| 8. | MSSubClass | Integer |
| 9. | LotFrontage | Integer |
| 10. | 1stFlrSF | Integer |
| 11. | 2ndFlrSF | Integer |
| 12. | BHK | Integer |
| 13. | Bathrooms | Integer |
| 14. | Gymnasium | Integer |
| 15. | SwimmingPool | Integer |
| 16. | School | Integer |
| 17. | Hospital | Integer |
| 18. | RoofStyle | Object |

Figure 3.2: Variable Table.

Here, shows only some important features.In this system we measure more than 40 features such as: Facilities of houses,Education facilities, Medical facilities, Shopping facility etc..

## 3.3 Detection Of Outliers

An outlier is an extremely high or extremely low-value value in the data it can be identified if whether the value is greater than interquartile range Q3 + 1.5 or Q1 - 1.5 detecting the interquartile range is arrange the data in an order from the lower value to the higher value, now the mean is taken for the first set of values and second set values now by subtracting both mean we can get the interquartile range the formula is Q3 + (1.5)(quartile range) and for Q1-(1.5)(quartile range) and I have calculated using the R program.

## 3.4 Tools Used

### TensorFlow

Tensorflow is an open-source library developed by Google primarily for deep learning applications. It also supports traditional machine learning. TensorFlow was originally developed for large numerical computations without keeping deep learning in mind.TensorFlow is a Python library for fast numerical computing created and released by Google. It is a foundation library that can be used to create Deep Learning models directly or by using wrapper libraries that simplify the process built on top of TensorFlow.

### Keras

Keras is an open-source neural-network library written in Python. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, R, Theano, or PlaidML. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible.

### Scikit Learn

The Scikit-Learn (SK Learn) is a Python Scientific toolbox for machine learning and is based on SciPy, which is a well-established Python ecosystem for science, engineering and mathematics. Scikit-learn provides an ironic environment with state of the art implementations of many wellknown machine learning algorithms, while sustaining an easy to use interface tightly integrated with the Python language [12],[13]. Scikit-learn features various functionalities like Clustering algorithms, Regression, Classification including random forests, gradient boosting, support vector machines, k-means and DBSCAN, and it has been designed to interoperate in conjunction with the Python scientific and numerical libraries SciPy and NumPy.

The step by step implementation using Scikit Learn is as follows.

Step 1: Import the required libraries.
Step 2: Load the dataset.
Step 3: Assign the values of columns 1 to 6 in the Dataset to “X”.
Step 4: Assign the values of column 7 which is the class label to “Y”.
Step 5: Fit decision tree classifier to the dataset.
Step 6: Predict the class label for the test data.

### Kivy

After building the model and successfully giving the result, the next step is to do the integration with the UI, for this purpose Kivy is used.

Kivy is a free and open source Python framework for developing mobile apps and other multitouch application software with a natural user interface. This means Kivy provides you with tools, libraries, and technologies that allow you to build an android application. Kivy is easy to put away routes together and this framework is mainly used for integrating python models.

### Flask

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

It is developed by Armin Ronacher, who leads an international group of Python enthusiasts named Pocco. Flask is based on the Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

## 3.5 Machine Learning Models

### 3.5.1 Ridge Regression

Ridge regression addresses some of the problems of Ordinary Least Squares by imposing a penalty on the size of coefficients. The new equation of a penalized residual sum of squares. Higher the values of alpha, bigger is the penalty and therefore the magnitude of coefficients is reduced. It is worth noting that we used Ridge CV which does implicit leave-one-out cross-validation to choose the best alpha.

### 3.5.2 Lasso Regression

The Mathematics behind lasso regression is quite similar to that of the ridge. The only difference is instead of adding squares of theta, we add the absolute value of that estimates sparse coefficients, i.e., it reduces the number of variables upon which the given solution is dependent. It does a kind of feature selection and thus leads to a less complex final model. For instance, when there are correlated features it will choose one and set the coefficient of the other to zero. the objective function of Lasso is defined by a linear model with an added regularization term.

### 3.5.3 Linear Regression

Linear Regression works on the line equation , y=mx+c , trend line is set through the data points to predict the outcome.The variable we are predicting is called the criterion variable and is referred to as Y. The variable we are basing our predictions on is called the predictor variable and is referred to as X.

### 3.5.4 Decision tree regressor

Decision tree regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete, i.e., it is not represented just by a discrete, known set of numbers or values.

The decision tree regressor algorithm is applied to the training data set. The Decision tree builds a regression model in the form of a tree structure.

### 3.5.5 Extra tree regressor

An extra-trees regressor, This class implements a meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The number of trees in the forest.

### 3.5.6 Elastic Net Regression

Elastic net regression combines the power of ridge and lasso regression into one algorithm. What this means is that with elastic net the algorithm can remove weak variables altogether as with lasso or to reduce them to close to zero as with ridge. All of these algorithms are examples of regularized regression.

This post will provide an example of elastic net regression in Python. Below are the steps of the analysis.

1. Data preparation.
2. Baseline model development.
3. Elastic net model development.

### 3.5.7 Elastic Net Model

Elastic net, just like ridge and lasso regression, requires normalize data. This argument is set inside the ElasticNet function. The second thing we need to do is create our grid. This is the same grid as we create for ridge and lasso in prior posts. The only thing that is new is the l1 ratio argument.

### 3.5.8 Support vector Regression(SVR)

Supervised Machine Learning Models with associated learning algorithms that analyze data for classification and regression analysis are known as Support Vector Regression. SVR is built based on the concept of Support Vector Machine or SVM. It is one among the popular Machine Learning models that can be used in classification problems or assigning classes when the data is not linearly separable.

It is one of the classic examples of supervised Machine learning technique. We could say it's one of the more powerful models which can be used in classification problems or assigning classes when the data is not linearly separable. I would give a classic kitchen example; I am sure most of us love chips? Of course, I do. I wanted to make home made chips. I bought potatoes from the vegetable market and hit my kitchen. All I did was follow a YouTube video. I started slicing the potatoes in the hot oil, the result was a disaster, and I ended up getting chips which were dark brown/black.

### 3.5.9 Gradient Boosting

Gradient Boosting to produce a predictive model from an ensemble of weak predictive models. Gradient boosting can be used for regression and classification problems. Here, we will train a model to tackle a diabetes regression task. We will obtain the results from GradientBoostingRegressor with least squares loss and 500 regression trees of depth 4.

The Gradient Boosting Machine is a powerful ensemble machine learning algorithm that uses decision trees.

Boosting is a general ensemble technique that involves sequentially adding models to the ensemble where subsequent models correct the performance of prior models. AdaBoost was the first algorithm to deliver on the promise of boosting.

Gradient boosting is a generalization of AdaBoosting, improving the performance of the approach and introducing ideas from bootstrap aggregation to further improve the models, such as randomly sampling the samples and features when fitting ensemble members.

### 3.5.10 Light GBM

LightGBM is a gradient boosting framework based on decision trees to increases the efficiency of the model and reduces memory usage.

It uses two novel techniques: Gradient-based One Side Sampling and Exclusive Feature Bundling (EFB) which fulfils the limitations of histogram-based algorithm that is primarily used in all GBDT (Gradient Boosting Decision Tree) frameworks. The two techniques of GOSS and EFB described below form the characteristics of LightGBM Algorithm. They comprise together to make the model work efficiently and provide it a cutting edge over other GBDT frameworks.

Gradient-based One Side Sampling Technique for LightGBM: Different data instances have varied roles in the computation of information gain. The instances with larger gradients(i.e., under-trained instances) will contribute more to the information gain. GOSS keeps those instances with large gradients (e.g., larger than a predefined threshold, or among the top percentiles), and only randomly drop those instances with small gradients to retain the accuracy of information gain estimation. This treatment can lead to a more accurate gain estimation than uniformly random sampling, with the same target sampling rate, especially when the value of information gain has a large range.

## 3.6 Evaluation

The idea of a regression is to predict a real value which means number in regression model we can compute the several values the most common terms are explained below.

**Visualizing numerical predictor variables with Target Variables.**

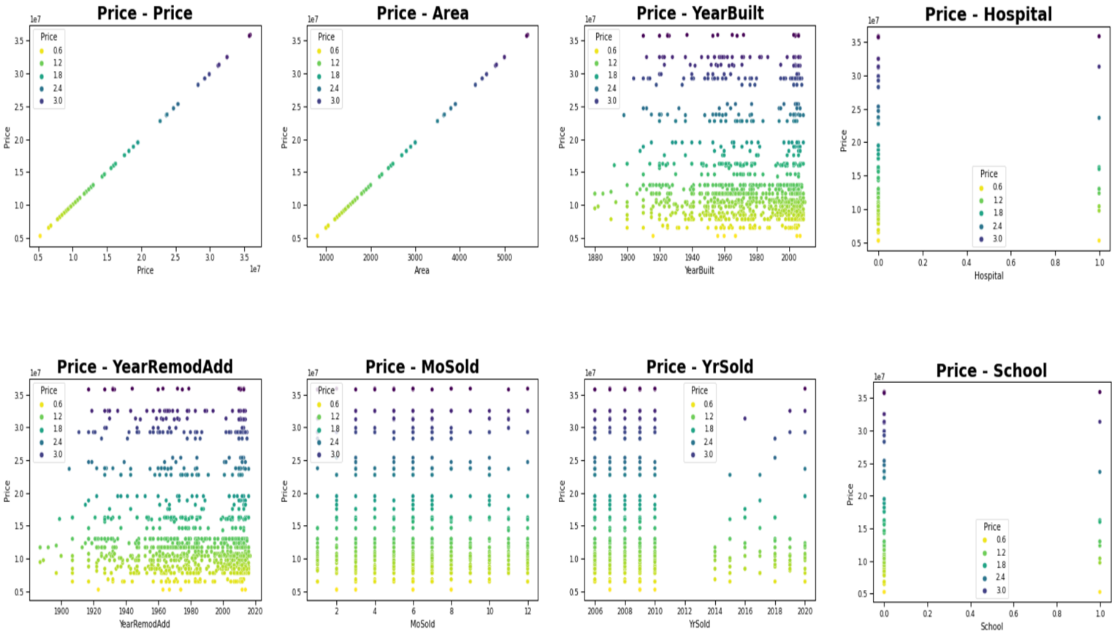


Figure 3.3: Predictor v/s Target Variables.

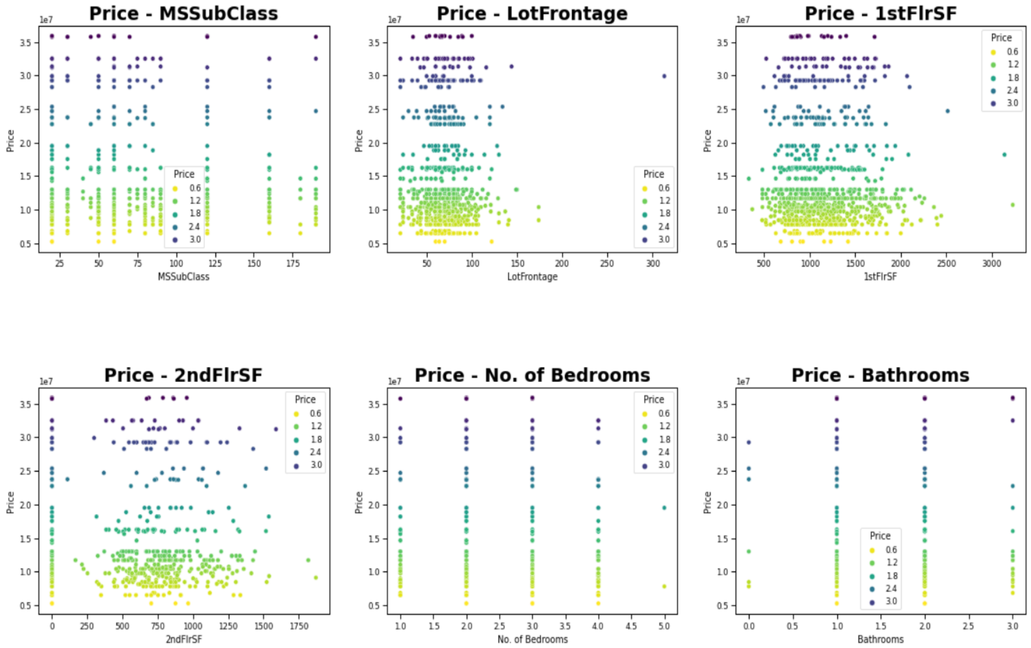


Figure 3.4: Predictor v/s Target Variables.

## 3.7 Root Mean Square Error

RMSE is a popular formula to measure the error rate of a regression model, however, it can only be compared between models whose errors are measured in the same units it can be measured using the given formula.

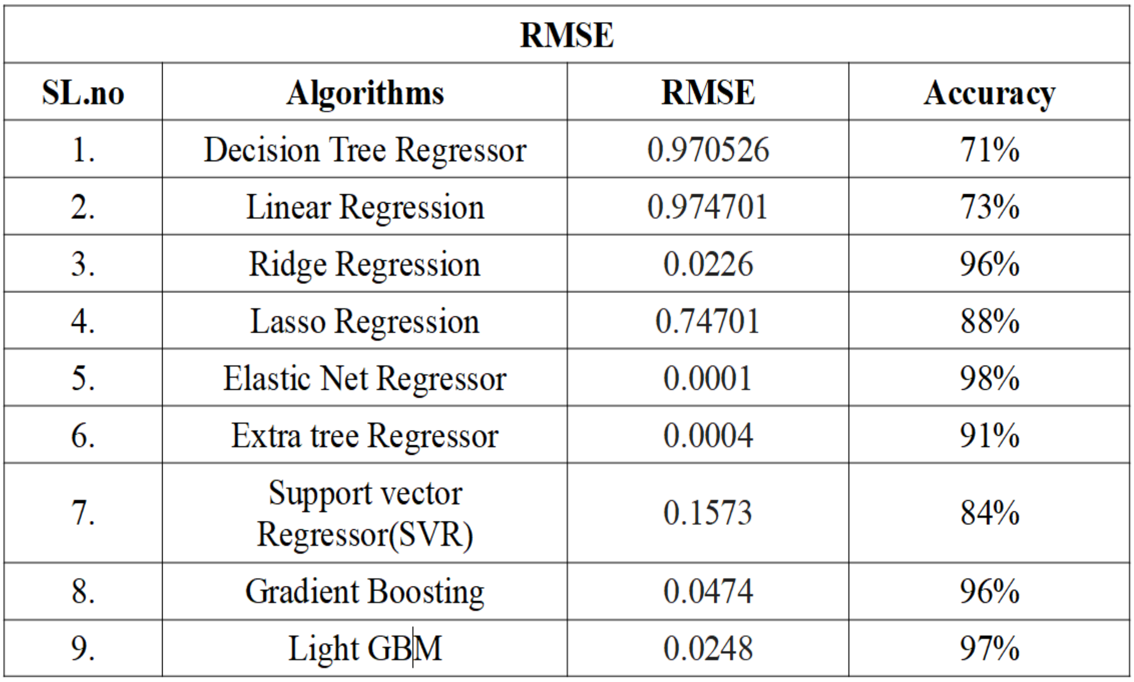
$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$$

Figure 3.5: RMSE formula.

Where n is the number of instances in the data, P is the predicted value for the I instance and O is the actual value the key concept is the predicted value is subtracted by the actual value square that and get the sum of all instances and divided it by number of instances, the RMSE will be achieved.

As discussed, the essential variables are used to calculate the error value and help to determine the how well can the algorithm predict the future prices, the below table describes the Root mean square error for the various algorithms and displayed below.

## 3.8 Evaluation of Result



| RMSE | | | |
|---|---|---|---|
| **SL.no** | **Algorithms** | **RMSE** | **Accuracy** |
| 1. | Decision Tree Regressor | 0.970526 | 71% |
| 2. | Linear Regression | 0.974701 | 73% |
| 3. | Ridge Regression | 0.0226 | 96% |
| 4. | Lasso Regression | 0.74701 | 88% |
| 5. | Elastic Net Regressor | 0.0001 | 98% |
| 6. | Extra tree Regressor | 0.0004 | 91% |
| 7. | Support vector Regressor(SVR) | 0.1573 | 84% |
| 8. | Gradient Boosting | 0.0474 | 96% |
| 9. | Light GBM | 0.0248 | 97% |

Figure 3.6: Algorithm's RMSE and Accuracy.

## 3.9 RMSE Scores of All Models

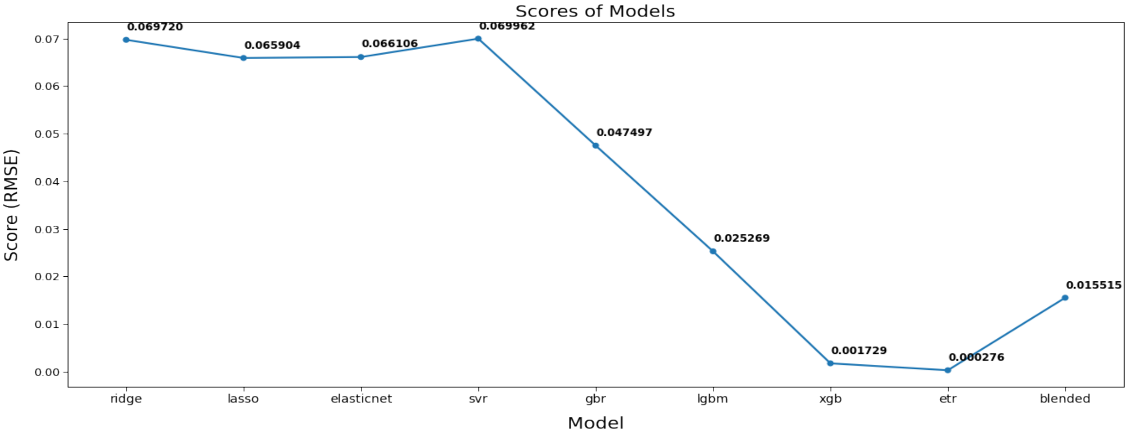


Figure 3.7: RMSE Scores of Each Models.

## 3.10 Over all Case Study

The first case study describes the results of all regression models, below the following graph shown the actual vs predicted value.

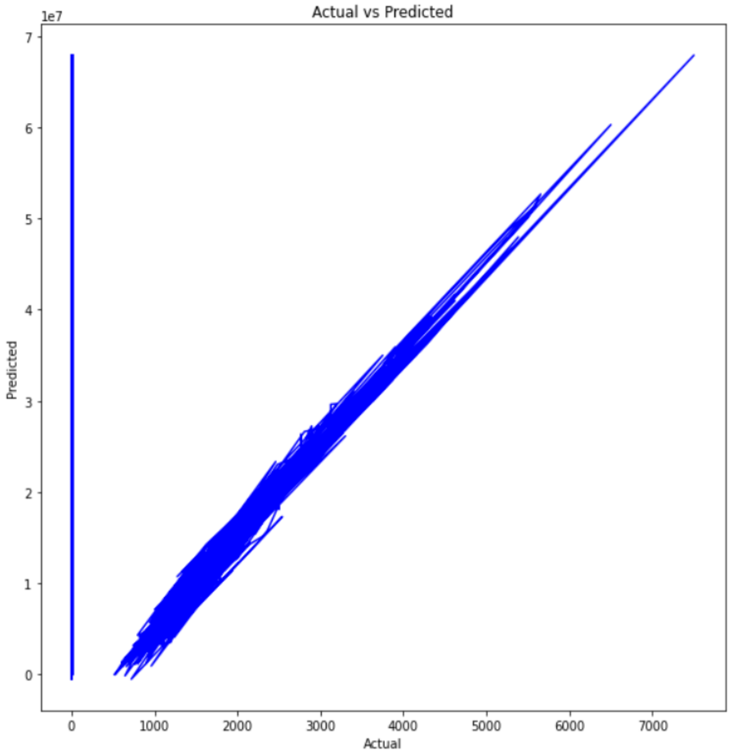
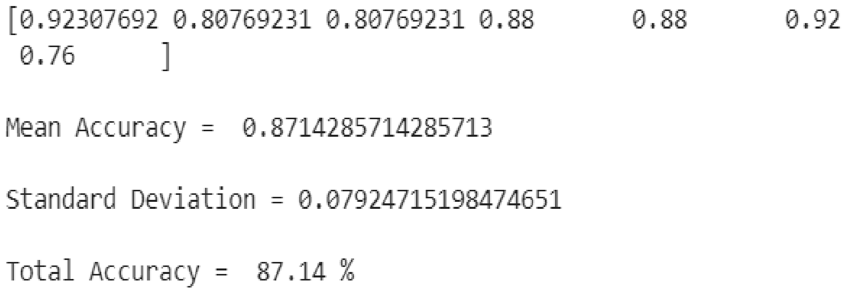


Figure 3.8: Actual vs Predicted Value.

Here the scatter plot shows the actual and predicted values are spread across linearly so that the price predicted is accurately about 90% with the RMSE value of 0.012.



```
[0.92307692 0.80769231 0.80769231 0.88       0.88       0.92
 0.76      ]

Mean Accuracy =  0.8714285714285713

Standard Deviation = 0.07924715198474651

Total Accuracy =  87.14 %
```

Figure 3.9: Total Accuracy.

# Chapter 4

# System Requirements

## 4.1 Hardware Requirements

### User Side

- **Processor** : Snapdragon 400+ or any equivalent
- **RAM** : 1 GB
- **Storage** : 500 MB

This application will run on any android smartphone with a minimum of 2 GB RAM and 500 MB of memory space. The amount of RAM is for user's pleasant user experience. Also, the amount of memory space needed will vary according to the user's preferences, Like having more documents will take up more space.

### Developer side

We need a laptop/PC with a minimum of 8 GB RAM. Also, an Intel i3 processor or any equivalent processor. These requirements are for running the Anaconda, PyCharm, android studio and flutter SDK.

## 4.2 Software Requirements

- **Operating System** : Android 6.0 or above or any other Operating System.
- **Front End** : Flutter ( Dart ) and Web Browser(html).

## 4.3 Technologies Used

### 4.3.1 Machine learning

Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.

In general, any machine learning problem can be assigned to one of two broad classifications:

1. Supervised learning.
2. Unsupervised learning.

### Supervised learning

In supervised learning, we are given a data set and already know what our correct output should look like, having the idea that there is a relationship between the input and the output. Supervised learning problems are categorized into ”regression” and ”classification” problems.

- **Regression problem**: We are trying to predict results within a continuous output, meaning that we are trying to map input variables to some continuous function.
- **Classification problem** We are instead trying to predict results in a discrete output. In other words, we are trying to map input variables into discrete categories.

### Unsupervised learning

Unsupervised learning allows us to approach problems with little or no idea what our results should look like. We can derive structure from data where we don’t necessarily know the effect of the variables. We can derive this structure by clustering the data based on relationships among the variables in the data. With unsupervised learning there is no feedback based on the prediction results.

### 4.3.2 Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. It’s high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development. Also, python used for use as a scripting or glue language to connect existing components. Python’s simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms.

### 4.3.3 Dart

Dart is a client-optimized programming language for apps on multiple platforms. Dart is a language by Google and is used to build mobile, desktop, server, and web applications. Dart is an object-oriented, class-based, garbage-collected language with C-style syntax. Dart can compile to either native code or JavaScript. The C-style syntax helps the new developers to learn the programming language in no time.

## 4.4 Platform Used

### 4.4.1 Android Studio

Android Studio is an IDE exclusively designed for developing Android applications. It consists of all Android SDK tools to design, develop, maintain, test, debug and publish our app. The IDE is very efficient in making the developing job easy. The IDE will automatically sense the variables, methods, classes, built-in functions. It also supports Git as a version control system to maintain the app changes and push them into Github. After coding the project, it can be compiled to an APK ( Android package ) file, in which we can run that APK file on any device and use the application. Also, the ide helps developing applications with the flutter framework.

### 4.4.2 Android SDK

Android SDK is the essential tool for the development of android applications. It packages many core features into one SDK, and we can include them in the application so that the development of applications using android SDK is a lot faster.

### 4.4.3 Flutter SDK

Flutter is an open-source UI software development kit created by Google. It used to develop applications that run on Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase. The availability of packages makes the development of user interfaces very easy. Also, using google's material UI, we can develop a better-looking UI for the users. Flutter makes use of dart programming language for developing applications.

### 4.4.4 Anaconda

Anaconda is a popular distribution of the Python and R programming languages for scientific computing, such as data science, machine learning applications, large-scale data processing, predictive analytics, etc just in one install, that aims to simplify package management and deployment. So it's great for having a short and simple setup. The distribution includes data-science packages suitable for Windows, Linux, and macOS. Like Virtualenv, Anaconda also uses the concept of creating environments so as to isolate different libraries and versions.

### 4.4.5 Pycharm

It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django as well as data science with Anaconda. PyCharm is cross-platform, with Windows, macOS and Linux versions.

PyCharm is an integrated development environment or IDE. While you don't compile Python code the same way you do Java or C sharp, PyCharm acts like a true IDE for Python, but unlike Visual Studio, PyCharm is geared specifically and only for Python development.

# Chapter 5

# System Design

## 5.1 System Architecture

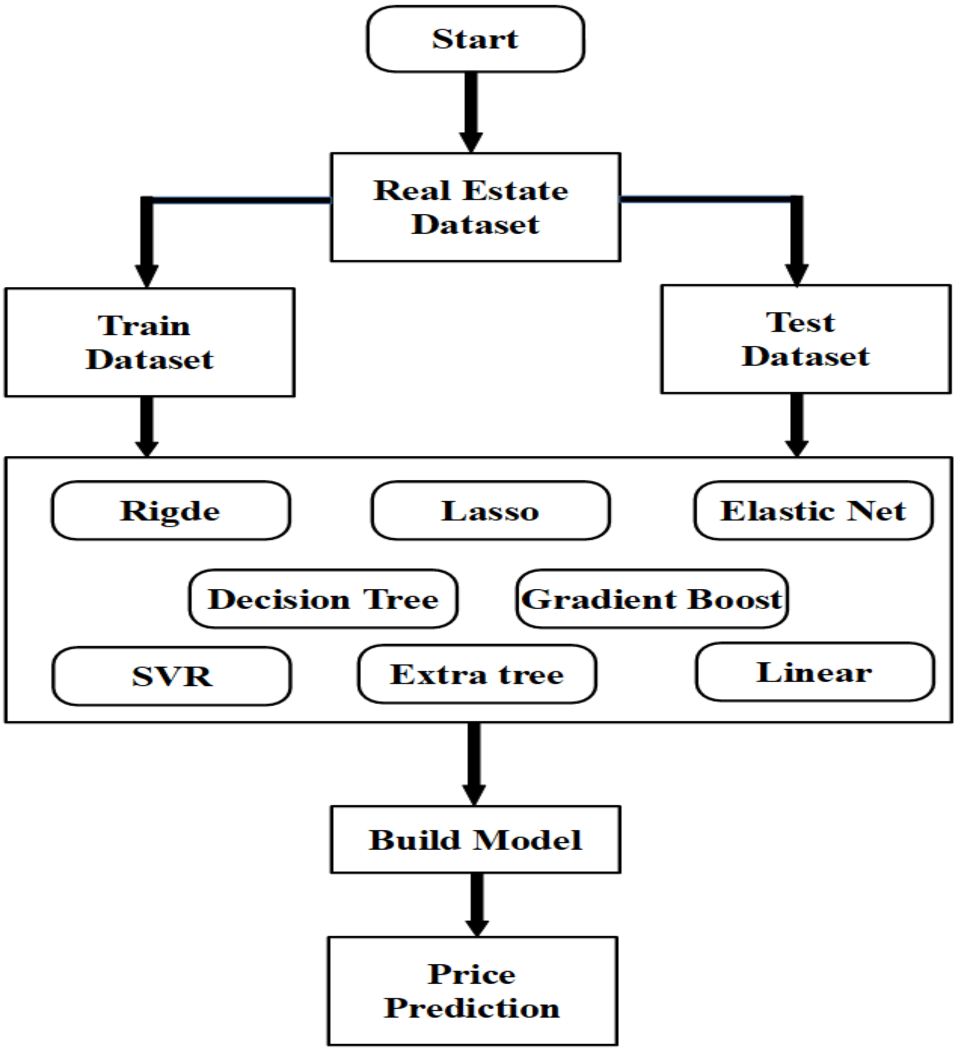


Figure 5.1: The Generic Flow of Development.

## 5.2 Use Case Diagram

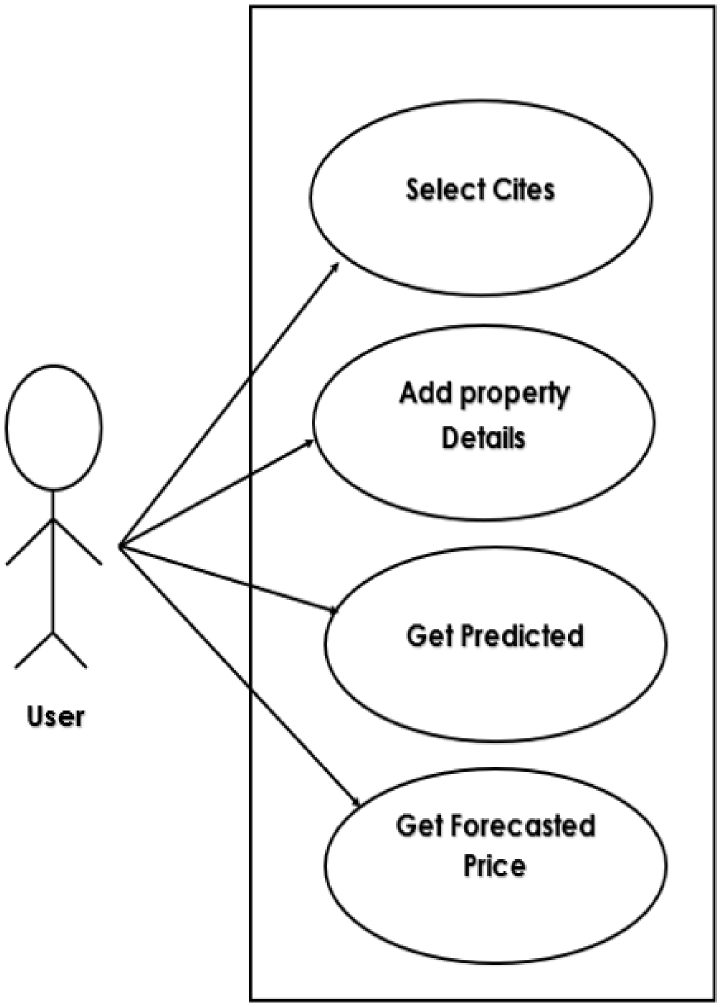


Figure 5.2: Use Case Diagram.

## 5.3 Use Case Description

This House Prediction System helps the user to check the price of the houses, not only for predicting the price of the house which is ready for sale but also for houses that are under construction.

This system is now available in around six major cities in India. Such as: Kochi, Chennai, Bangalore, Mumbai, Delhi and Kolkata.

User can select the available cities. then add the property details like area(square feet), no.of BHK's, no.of bathrooms etc.. and also select the location. Then the system shows the accurate prediction based on the given property details. This system is now available in android application and also in web application.

## 5.4 Context Diagram

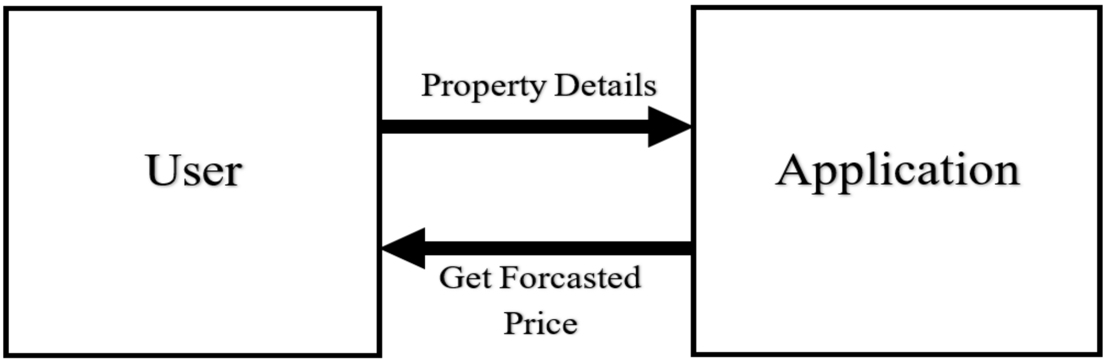


Figure 5.3: Context Diagram.

## 5.5 Sequence Diagram

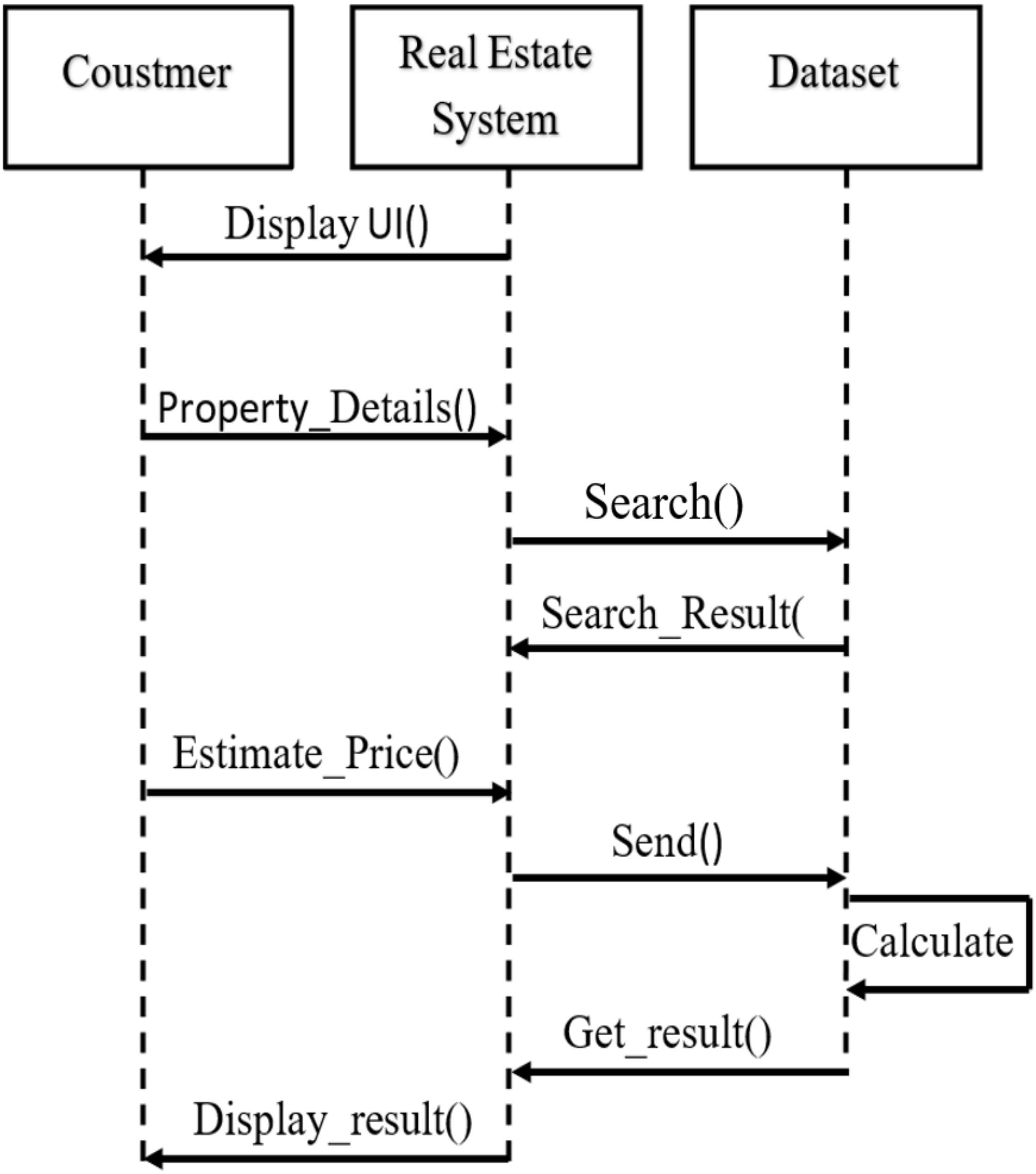


Figure 5.4: Sequence Diagram.

## 5.6 Front End Design

### 5.6.1 Navigation Diagram

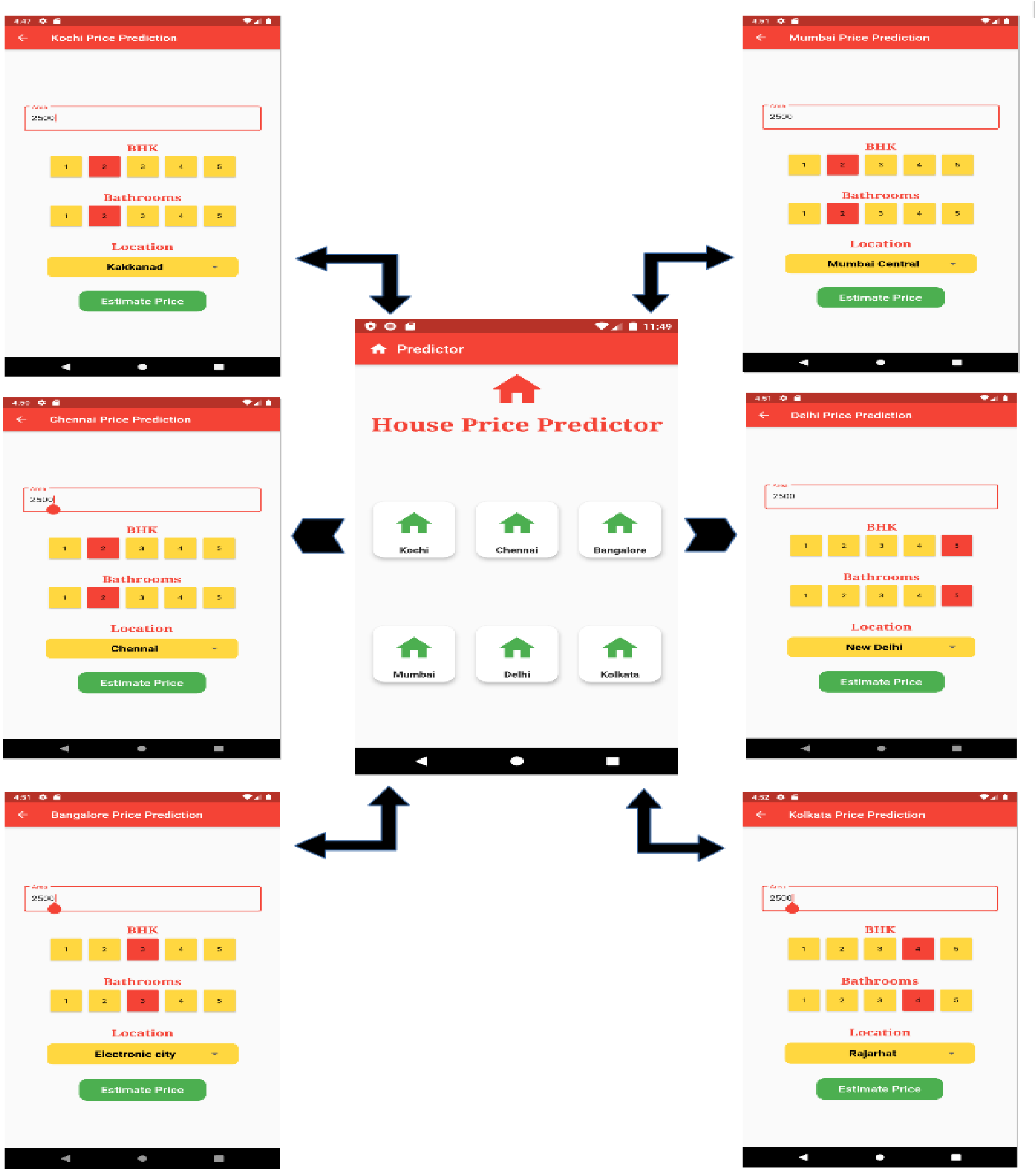


Figure 5.5: Navigation Diagram.

### 5.6.2 User Interface

### Android Application

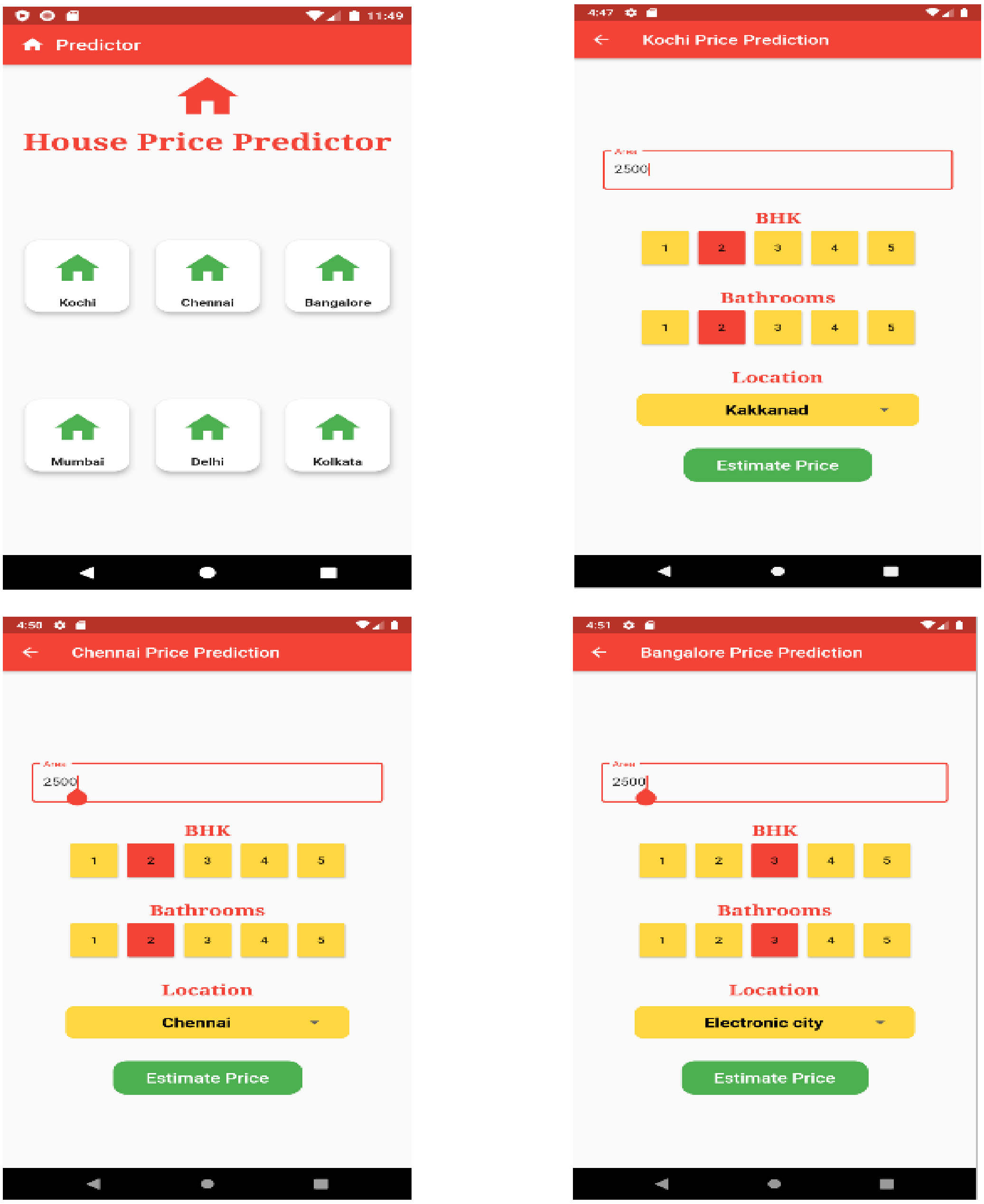
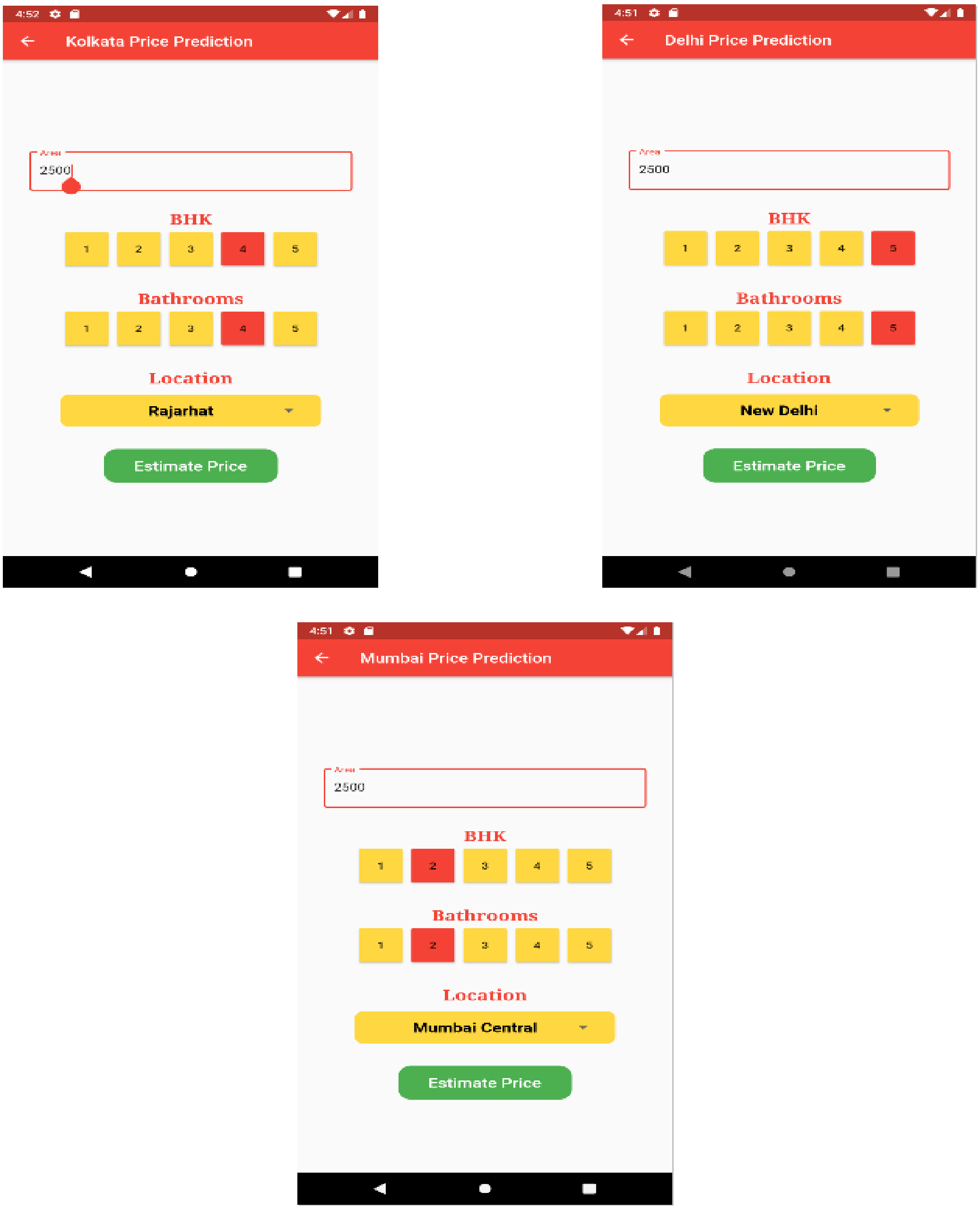


Figure 5.6: Android Application UI's.



Area
2500

BHK
1 2 3 [✓ 4] 5

Bathrooms
1 2 3 [✓ 4] 5

Location
Rajarhat

Area
2500

BHK
1 2 3 4 [✓ 5]

Bathrooms
1 2 3 4 [✓ 5]

Location
New Delhi

Area
2500

BHK
1 [✓ 2] 3 4 5

Bathrooms
1 [✓ 2] 3 4 5

Location
Mumbai Central

Figure 5.7: Android Application UI's.

## Web Application

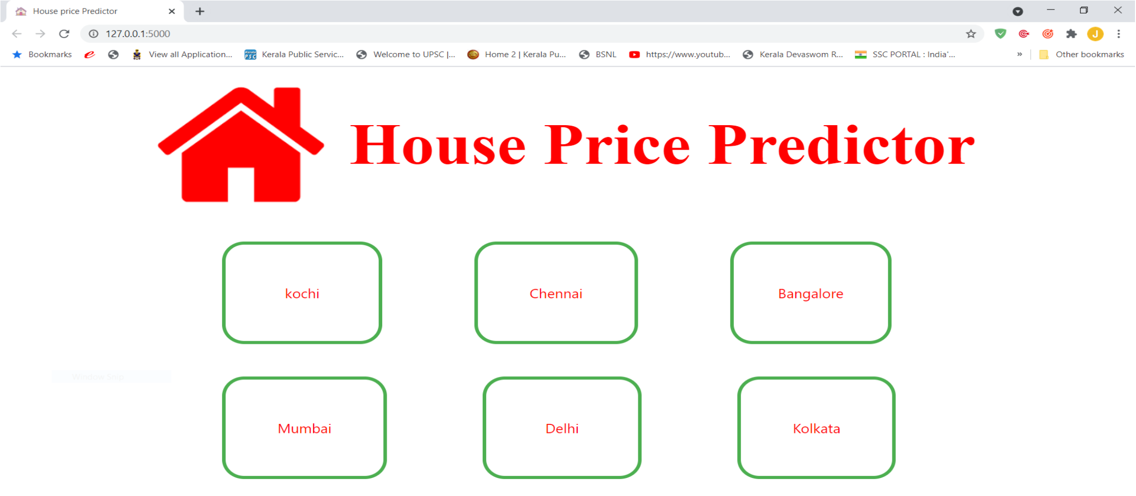
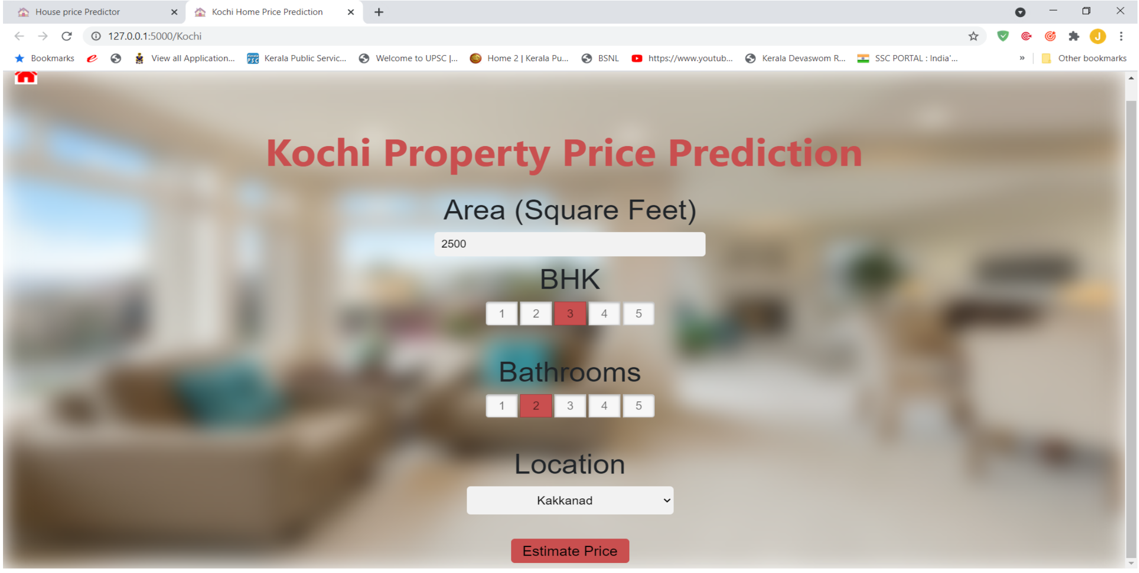


Figure 5.8: Web Application Home Page.



Area (Square Feet)

2500

BHK

1 2 [✓ 3] 4 5

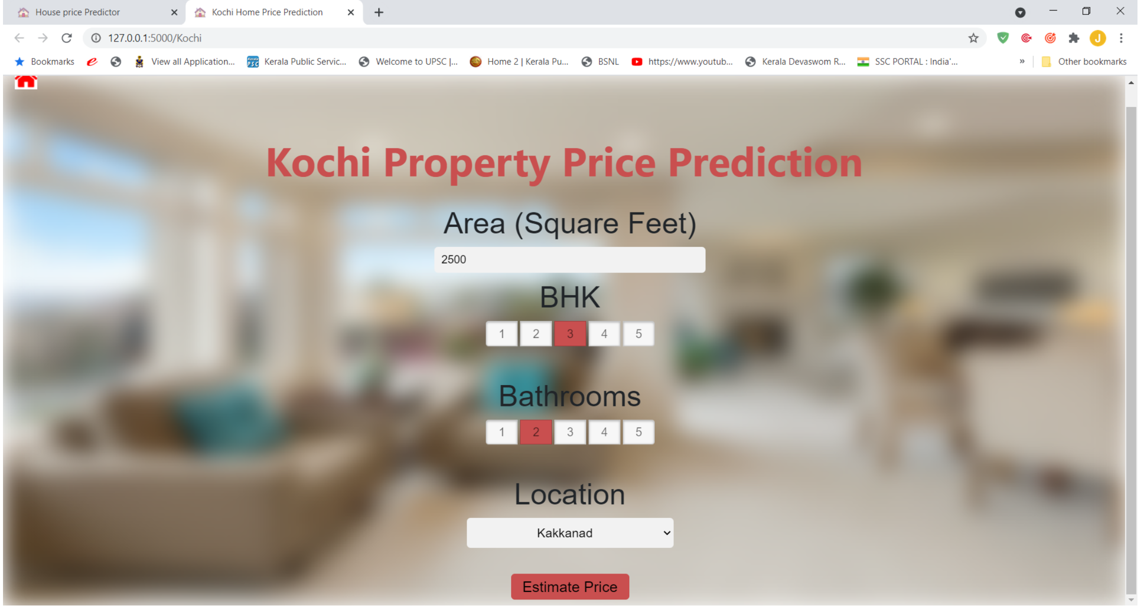
Bathrooms

1 [✓ 2] 3 4 5

Location

Kakkanad

Figure 5.9: Add Details Application .



Kochi Property Price Prediction

Area (Square Feet)

2500

BHK

1 2 [✓ 3] 4 5

Bathrooms

1 [✓ 2] 3 4 5

Location

Kakkanad

Figure 5.10: Price Prediction.

# Chapter 6

# System Testing

## 6.1 Testing Strategies

Testing is a process of creating a program with the explicit intention of finding error that is making the program fail. Successful test then, is one that finds an as yet undiscovered error. It makes a logical assumption that if all the parts of the system are correct, the goal will be successfully achieved. It is the stage of implementation, which ensures that the system works accurately and effectively before the live operation commences. It is a confirmation that all are correct and an opportunity to show users that the system must be tested and show that the system will operate successfully and produce expected results under expected conditions.

During testing the program to be tested is executed with a set of test data and the output of the test data is evaluated to determine if the program is performing as expected. A series of testing are performed for the proposed system before the system is ready.

### 6.1.1 White Box Testing

The white box testing strategy deals with the internal logic and structure of 30 the code. White box testing is also called glass, structural, open box, or clear box testing. The tests are written based on the white box testing strategy incorporating coverage of the code written, branches, paths, statements and internal logic of the code, etc. To implement white box testing, the tester has to deal with the code and hence is needed to possess knowledge of coding and logic i.e. internal working of the code. The white box test also needs the tester to look into the code and find out which unit/statement/chunk of the code is malfunctioning. Advantages of White box testing are : i. As the knowledge of internal coding structure is a prerequisite, it becomes very easy to find out which type of input/data can help in testing the application Effectively. ii. The other advantage of white box testing is that it helps in optimizing the code it helps in removing the extra lines of code, which can bring in hidden defects.

### 6.1.2 Black Box Testing

Black Box Testing is not a type of testing; it instead is a testing strategy, which does not need any knowledge of internal design or code etc. As the name ”black box” suggests no knowledge of internal logic or code structure is required. The types of testing under this strategy are based/focused on the testing for requirements and functionality of the work product/software application. Black box testing is sometimes also called ”Opaque Testing”, ”Functional/Behavioral Testing” and ”Closed Box Testing”. The base of the Black box testing strategy lies in the selection of appropriate data as per functionality and testing it against the functional specifications to check for normal and abnormal behavior of the system. Nowadays, it is becoming common to route the testing work to a third party as the developer of the system knows too much of the internal logic and coding of the system, which makes it unfit to test the application by the developer. To implement Black Box Testing Strategy, the tester is needed to be thorough with the requirement specifications of the system and as a user, should know, how the system should behave in response to the particular action. us testing types that fall under the Black Box Testing strategy are functional testing, stress testing, recovery testing, volume testing, User Acceptance Testing (also known as UAT), system testing, Sanity or Smoke testing, load testing, Usability testing, Exploratory testing, ad-hoc testing, alpha testing, beta testing, etc.

### 6.1.3 Unit Testing

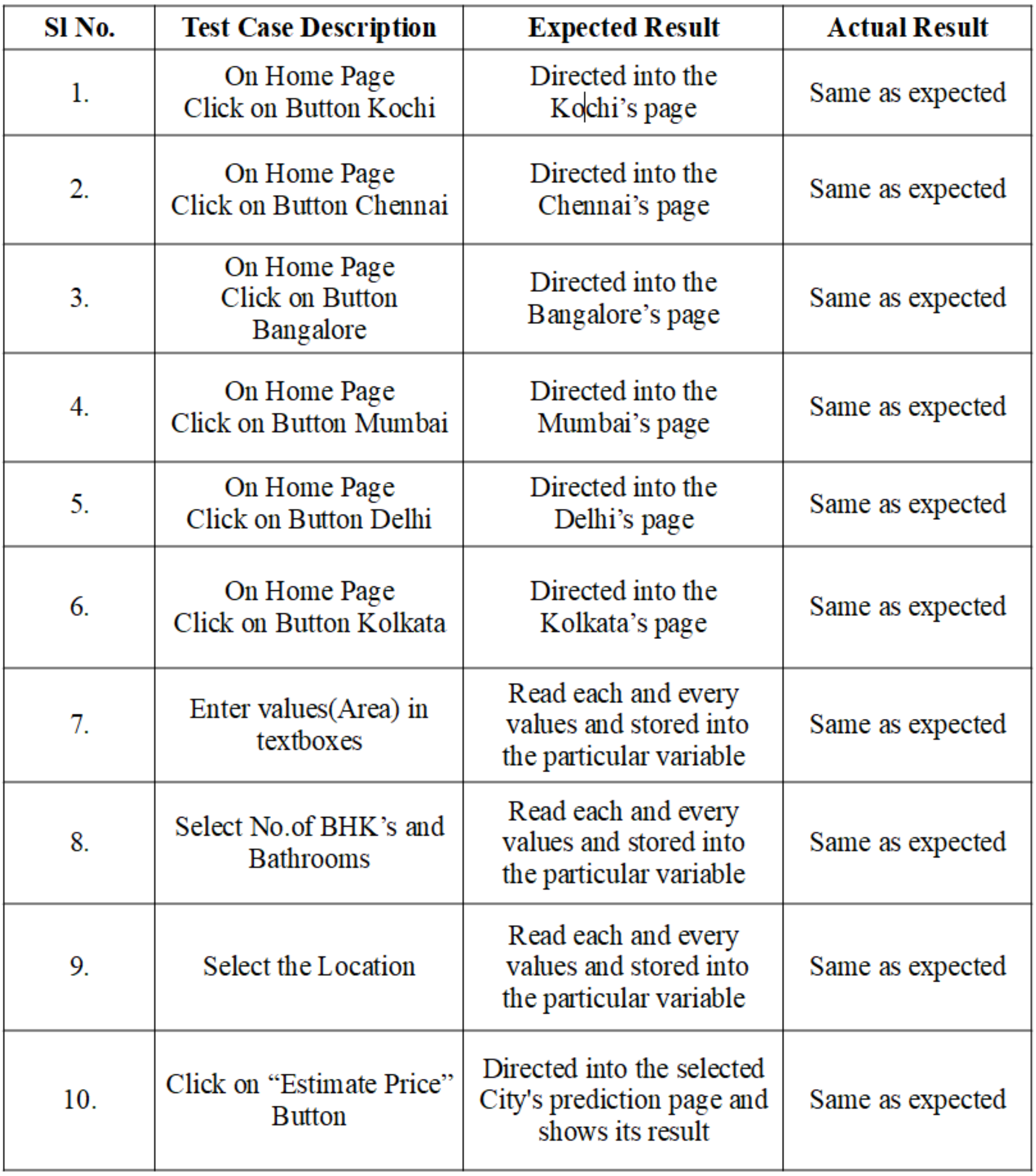
Unit testing focuses on the modules independently of one another to locate errors. In this application each module is tested individually and may caught errors such as save same data two times, does not clear the fields after the save, exceeding the length of textboxes etc. Also, validation is not properly done. These errors may clear under unit testing.

### 6.1.4 Integration Testing

Integration testing is a systematic testing for construction of the program structure while at the same time conducting tests to uncover Instead of testing the system as a whole, unit testing focuses on the module that makes up the system. Each module is taken up individually and tested for correctness in coding and logic One specified target of integration testing is the interface, whether errors associated with interfacing. The objective is to take until-tested modules and build a program structure that has been dictated by design.

### 6.1.5 Test Cases

The purpose of the test cases is to test the various inputs and see if the output produces any error or not. There are different test cases according to the system. The system is tested with different types of values, like single value multiple values, and see it can generate the expected output.



| Sl No. | Test Case Description | Expected Result | Actual Result |
|---|---|---|---|
| 1. | On Home Page<br>Click on Button Kochi | Directed into the<br>Kochi's page | Same as expected |
| 2. | On Home Page<br>Click on Button Chennai | Directed into the<br>Chennai's page | Same as expected |
| 3. | On Home Page<br>Click on Button<br>Bangalore | Directed into the<br>Bangalore's page | Same as expected |
| 4. | On Home Page<br>Click on Button Mumbai | Directed into the<br>Mumbai's page | Same as expected |
| 5. | On Home Page<br>Click on Button Delhi | Directed into the<br>Delhi's page | Same as expected |
| 6. | On Home Page<br>Click on Button Kolkata | Directed into the<br>Kolkata's page | Same as expected |
| 7. | Enter values(Area) in textboxes | Read each and every values and stored into the particular variable | Same as expected |
| 8. | Select No.of BHK's and Bathrooms | Read each and every values and stored into the particular variable | Same as expected |
| 9. | Select the Location | Read each and every values and stored into the particular variable | Same as expected |
| 10. | Click on "Estimate Price" Button | Directed into the selected City's prediction page and shows its result | Same as expected |

Figure 6.1: Test Cases.

# Chapter 7

# Conclusion And Future Scope

## 7.1 Scope for future development

There are many ways in which the application could be improved upon. In this part, I add some of the features for the application which are not yet present but could be implemented in future development.

- **Add Additional features like air quality and crime rate etc.**
  In future, take a measurement of air quality, crime rate etc..and it will help to get more accuracy of prediction. These features provide a valuable contribution towards predicting property prices since the higher values of these features will lead to a reduction in house prices.
- **Expanding it to other cities.**
- **To make the system even more informative and user-friendly.**
  This facility will help the user to analyze more about the location. It also helps to analyze the developments in near by the location.
- **Including Google map facility.**
- **Add a facility to buying and Selling in virtual platform.**
- **Develop an Advanced Real Estate System.**
  In this modern era, The world is shifting from manual to automated systems. So, real estate has also completely shifting to the virtual system. Every thing will be include in this system such as buying, selling and also all paper works.

## 7.2 Conclusion

In this project, we have used machine learning algorithms to predict the house prices. We have used different procedures to analyse the dataset and find the correlation between the parameters. Thus we can select the parameters which are not correlated to each other and are independent in nature. These feature sets were then given as an input to our algorithms and house prices are predicted.

Improvement in computing technology has made it possible to examine social information that cannot previously be captured, processed and analysed. New analytical techniques of machine learning can be used in property search. In this study, our models are trained with different housing property data utilising Lasso regression, Ridge regression, Support Vector regression (SVR), Gradient Boosting machine (GBM), Grid Search CV, Extra tree regression, Decision tree regression and Linear regression . We have demonstrated that advanced machine learning algorithms can achieve very accurate prediction of property prices, as evaluated by the performance metrics.

We must be cautious that these machine learning tools also have their own limitations. There are often many potential features for researchers to choose and include in the models so that a very careful feature selection is essentia.

Many conventional estimation methods produce reasonably good estimates of the coefficients that unveil the relationship between output variable and predictor variables. These methods are intended to explain the real-world phenomena and to make predictions, respectively. They are used for developing and testing theories to perform causal explanation, prediction, and description. Based on these estimates, investigators can interpret the results and make policy recommendations. However, machine learning algorithms are often not developed to achieve these purposes. Although machine learning can produce model predictions with tremendously low errors, the estimated coefficients (or weights, in machine learning terminology) derived by the models may sometimes make it hard for interpretation.

To conclude, the application of machine learning in property research is still at an early stage. We hope this study has moved a small step ahead in providing some methodological and empirical contributions to property appraisal, and presenting an alternative approach to the valuation of housing prices. Future direction of research may consider incorporating additional property transaction data from a larger.

# Appendix A

# Sample Code

## Web Application : app.py

```
from flask import Flask, request, render_template
import pickle
import numpy as np
import json

app = Flask(__name__)


@app.route('/')
def home():
    return render_template("home.html")


def kochi():
    @app.route('/Kochi')
    def Kochi():
        return render_template('Kochi.html')

    __locations = None
    __data_columns = None
    model = pickle.load(open("Kochi_home_prices_model.pickle", "rb"))

    def get_estimated_price(input_json):
        try:
         loc_index = __data_columns.Kochi(input_json['location'].lower())
        except:
            loc_index = -1
        x = np.zeros(3)
        x[0] = input_json['sqft']
        x[1] = input_json['bath']
        x[2] = input_json['bhk']
        if loc_index == 0:
            x[loc_index] = 1
        result = round(model.predict([x])[0], 2)
        print(result)
        return result

    def get_location_names():
        return __locations

    def load_saved_artifacts():
        print("Loading the saved artifacts...start !")
        global __data_columns
        global __locations
        global model

        with open("columns.json") as f:
            __data_columns = json.loads(f.read())["data_columns"]
            __locations = __data_columns[3:]

    @app.route("/Kochi_prediction", methods=["POST"])
    def Kochi_prediction():
        if request.method == 'POST':
            input_json = {
                "location": request.form['sLocation'],
                "sqft": request.form['Squareft'],
                "bhk": request.form['uiBHK'],
                "bath": request.form['uiBathrooms']
            }
            result = get_estimated_price(input_json)

            if result > 10000000:
                result = round(result / 10000000, 2)
                result = str(result) + ' Crore'
            else:
                result = round(result / 100000, 2)
                result = str(result) + ' Lakhs'

        return render_template('Kochi_prediction.html', result=result)
kochi()
chennai()
bangalore()
mumbai()
delhi()
kolkata()

if __name__ == '__main__':
    print("Starting Python Flask Server")
    app.run(debug=True)
```

# Android Application : main.dart

```
import 'package:price_predictor/Kochii/Kochi.dart';
import 'Kochii/Kochi.dart';
import 'package:price_predictor/Chennai/Chennai.dart';
import 'package:price_predictor/Bangalore/Bangalore.dart';
import 'package:price_predictor/Mumbai/Mumbai.dart';
import 'package:price_predictor/Delhi/Delhi.dart';
import 'package:price_predictor/Kolkata/Kolkata.dart';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';

void main() {
  WidgetsFlutterBinding.ensureInitialized();
  runApp(Predictor());
}

// App Class
class Predictor extends StatelessWidget {
  Predictor({Key key}) : super(key: key);
  Widget build(BuildContext context) {
    return new MaterialApp(
      title: 'Predictor',
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        primarySwatch: Colors.red,
      ),
      home: new Home(title: '  Predictor'),
    );
  }
}

// App Home Page Class
class Home extends StatelessWidget {
  final String title;

  Home({this.title}) : super();
  Widget build(BuildContext context) {
    return new Scaffold(
      appBar: new AppBar(
          title: Row(
              //mainAxisAlignment: MainAxisAlignment.center,
              children: [
                new Icon(Icons.home),
                Text(title)
                ]),
          actions: <Widget>[]),
      body: new HomePage(),
    );
  }
}

// Home Activity
class HomePage extends StatelessWidget {
  HomePage() : super();

  // Navigates to Kochi Price Prediction
  void _navToKochi(context) {
    Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => new Kochi(title: "Kochi Price Prediction"),
        ));
  }
  // Navigates to Chennai Price Prediction
  void _navToChennai(context) {
    Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => new Chennai(title: "Chennai Price Prediction"),
        ));
  }
  // Navigates to Bangalore Price Prediction
  void _navToBangalore(context) {
    Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => new Bangalore(title: 'Bangalore Price Predictio
        ));
  }
  // Navigates to Mumbai Price Prediction
  void _navToMumbai(context) {
    Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => new Mumbai(title: 'Mumbai Price Prediction'),
        ));
  }
  // Navigates to Delhi Price Prediction
  void _navToDelhi(context) {
    Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => new Delhi(title: 'Delhi Price Prediction'),
        ));
  }
  // Navigates to Kolkata Price Prediction
  void _navToKolkata(context) {
    Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => new Kolkata(title: 'Kolkata Price Prediction'),
        ));
  }

  Widget build(BuildContext context) {
    return new Column(
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
      children: [
        IconButton(
            icon: Icon(Icons.home_sharp),
            color: Colors.red,
            iconSize: 72.0,
            onPressed: () {}
        ),
        Text(
            "House Price Predictor",
            style: TextStyle(fontSize:32,
              fontFamily:'Times New Roman',
              fontWeight: FontWeight.bold,
              color: Colors.red,)),
        new Spacer(),
        new Row(
          mainAxisAlignment: MainAxisAlignment.spaceEvenly,
          children: [
            createButtons('Kochi', _navToKochi, context),
            createButtons('Chennai', _navToChennai, context),
            createButtons('Bangalore', _navToBangalore, context),
          ],
        ),
        new Spacer(),
        new Row(
          mainAxisAlignment: MainAxisAlignment.spaceEvenly,
          children: [
            createButtons('Mumbai', _navToMumbai, context),
            createButtons('Delhi', _navToDelhi, context),
            createButtons('Kolkata', _navToKolkata, context),
          ],
        ),
        new Spacer(),
      ],
    );
  }
}

// Returns the button in card
Widget createButtons(String label, onPress, context) {
  return new GestureDetector(
    onTap: () {
      onPress(context);
    },
    child: new Card(
      shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(15)),
      elevation: 5,
      child: new Container(
        height: 100,
        width: 100,
        child: new Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            new Spacer(),
            IconButton(
              icon: Icon(Icons.home_sharp),
              color: Colors.green,
              iconSize: 52.0,
                onPressed:(){}
            ),
            new Spacer(),
            new Text(label,
                textAlign: TextAlign.center,
                style: TextStyle(
                  fontWeight: FontWeight.bold,
                  color: Colors.black87,
                )),
            new Spacer(),
          ],
        ),
      ),
    ),
  );
}
```

# Appendix B

# References

[1] Sifei Lu, Zengxiang Li, Zheng Qin, Xulei Yang, Rick Siow Mong Goh, “A Hybrid Regression Technique for House Price Prediction”, December 2017.

[2] J.R. Quinlan, “C4.5: programs for Machine Learning”, Morgan Kaufmann, New York, 1993.

[3] J.R. Quinlan, “Induction of Decision Trees”, Machine Learning 1, 1986, pp.81-106.

[4] SamDrazin and Matt Montag”, Decision Tree Analysis using Weka”, Machine Learning-Project II, University of Miami.

[5] Timothy C. Au, “Random Forests, Decision Trees, and Categorical Predictors: The Absent Levels Problem”, Journal of Machine Learning Research 19, 2018, pp. no.1- 30.

[6] Gang-Zhi Fan, Seow Eng Ong and Hian Chye Koh, “Determinants of House Price: A Decision Tree Approach”, Urban Studies, Vol. 43, No. 12, November 2006, PP.NO.2301- 2315. [7] Ong, S. E., Ho, K. H. D. and Lim, C. H., “A constan-tquality price index for resale public housing flats in Singapore”, Urban Studies, 40(13), 2003, pp. 2705 –2729.

[8] Berry, J., McGreal, S., Stevenson, S., “Estimation of apartment submarkets in Dublin, Ireland”, Journal of Real Estate Research, 25(2), 2003, pp. 159–170.

[9] Lakshmi, B. N., and G. H. Raghunandhan. ”A conceptual overview of data mining.” 2011 National Conference on Innovations in Emerging Technology. IEEE, 2011.

[10] Neelam Shinde, Kiran Gawande, “Valuation of house prices using Predictive Techniques”, International Journal of Advances in Electronics and Computer Science, ISSN: 2393-2835, Volume-5, Issue-6, Jun.-2018 pp. 34 to 40.

[11] Adyan Nur Alfiyatin , Hilman Taufiq, Ruth Ema Febrita and Wayan Firdaus

Mahmudy, “Modeling House Price Prediction using Regression Analysis and Particle Swarm Optimization”, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 8, No. 10, 2017, pp. 323 to 326.

[12] Nihar Bhagat, Ankit Mohokar, Shreyash Mane ”House Price Forecasting using Data Mining” International Journal of Computer Applications,2016.

[13] `http://scikit-learn.org/stable/index.html`

[14] `http://scikit-learn.org/stable/auto_examples/index.html`

[15] `https://kivy.org/doc/stable/guide/android.html`

[16] `https://flask.palletsprojects.com/en/1.1.x/`

[17] `https://www.youtube.com/watch?v=x0uinJvhNxI`

[18] `https://flutter.dev/docs`