# A

# PROJECT REPORT

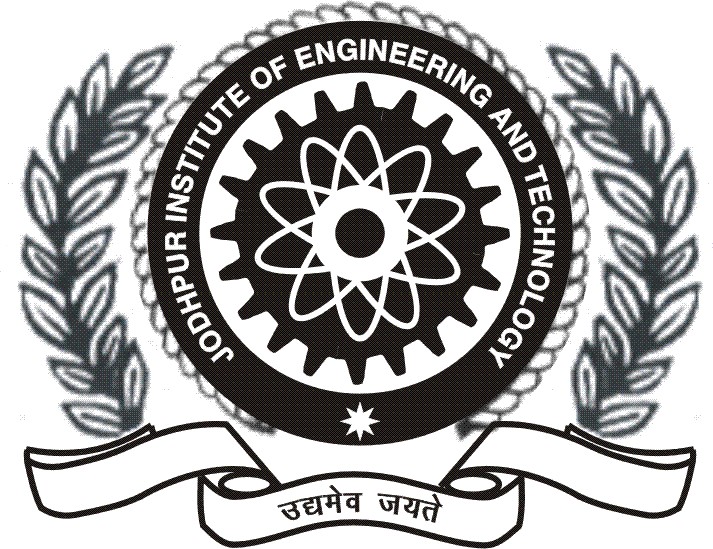
**ON**

**House Price Prediction ML App**

**In partial fulfillment of**

**B. Tech III year Computer Science & Engineering**

**`**



**Session 2020-2021**

# Submitted To: Submitted by:

## PRANITA KALLA AMAN KUMAR SHARMA

ASSISTANT PROFESSOR V Semester

## CSE 21EJICS015

Department of Computer Science & Engineering Jodhpur Institute of Engineering & Technology, Jodhpur June 2023-2024

# CERTIFICATE

This is to certify that the project entitled **“House Price Prediction ML App”** has been carried out by the student of “**Jodhpur Institute of Engineering and Technology, Jodhpur”** under my guidance and supervision in partial fulfilment of the degree of Bachelor of Engineering in Computer Science during the academic year 2023-2024.

## Aman Kumar Sharma Ms. Pranita Kalla

## Date: 23/10/2023 Assistant Professor

## CSE Department,

## JIET Jodhpur

# Table of Contents

1. [Introduction](#_Chapter_1) ……………………………………iii
2. [Experimental Steup](#_Chapter_2) ……………………………………iv
3. [Files and APK link](#_Chapter_3)  ……………………………………vii
4. [Results](#_Chapter_4) ……………………………………xii
5. [Conclusion & Future work](#_Chapter_5) …………………………………...xiii

# Chapter 1

Introduction:

In the ever-changing real estate market, making smart decisions when buying or selling a home is really important. That's where the House Price Prediction ML App comes in. It's like a game-changer for how you deal with real estate. This app uses Machine learning programs to guess how much a house should cost, and it's surprisingly accurate.

It's way better than just guessing or using limited information to figure out house prices. You can tell the app stuff about the house, like where it’s location, what is the total Square it, and how many bedrooms it has. Then, the app uses its smart technology, which has learned a lot about real estate, to give you a good guess about how much the house is worth.

Whether you're looking to buy a house, trying to decide how much to sell one for, or even thinking about investing in real estate, this app is like your personal real estate expert. It helps you understand the tricky world of real estate and makes sure you make money-smart decisions that match your financial goals. It's like having a real estate expert right in your pocket!

# Chapter 2

Experimental Setup :

1. Install Prerequisites:

a. Android Studio: Download and install the latest version of Android Studio from the official website (<https://developer.android.com/studio>).

b. Java Development Kit (JDK): Ensure you have a compatible JDK installed (e.g., OpenJDK or Oracle JDK). Set up the JAVA\_HOME environment variable.

c. Python 3: To build the machine learning model you need to python and its library like (numpy, pandas, flask, pickle, tenserflow, matplotlib etc.) you can download python from official website ([https://www.python.org/downloads](https://www.python.org/downloads/)/)

2. Set Up Android Emulator:

a. Open Android Studio and navigate to "Tools" -> "AVD Manager."

b. Click "Create Virtual Device," then choose a device definition and system image (e.g., Pixel 5 with Android 12).

c. Configure the hardware profile, such as RAM size and storage, and create the virtual device.

3. Clone or Create Your Project:

a. Create a new Android Studio project or clone an existing project for your House Price Prediction ML App.

b. Configure project settings, including the package name, language (Java), and \*minimum Android version.

4. Install SDK Packages:

a. Open the SDK Manager within Android Studio by navigating to "Tools" -> "SDK Manager."

b. Ensure you have the required Android SDK platforms, build tools, and system images installed, based on your project's compatibility requirements.

5. Configure Dependencies:

a. Go to Gradle Scripe>build.gradle(Module:app) then add dependencies “implementation 'com.android.volley:volley:1.2.1'”

6. Code Development:

a. first develop python API to predict the price and we have to deploy this api to the live server so that we use api in android app.

b. Begin developing your House price prediction ML App by writing code for the Application functionalities and features as outlined in your project's scope.

7. Debug and Test:

a. Use the built-in debugging tools in Android Studio to identify and resolve issues in your code.

b. Test the Application on the Android Emulator by deploying the app and running it on the virtual device.

8. UI Design:

a. Create or modify the user interface of your browser using XML layout files. Design a user-friendly and visually appealing UI based on Material Design guidelines.

9. Performance Optimization:

a. Optimize your app's performance by managing resource usage, reducing memory leaks, and enhancing rendering speed.

10. Testing on Real Devices:

a. Test the Application on physical Android devices to ensure compatibility and performance. Connect your device to your development machine and deploy the app for testing.

11. Deployment:

a. Prepare your app for deployment by configuring the release build in Android Studio.

b. Sign your app with a keystore to ensure authenticity and security.

c. Publish your Android App to the Google Play Store or another distribution platform, following the platform's guidelines and procedures.

# Chapter 3

Files:

**Java Code:**  
package com.example.bhpmlproject;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Spinner;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import com.android.volley.Request;  
import com.android.volley.RequestQueue;  
import com.android.volley.Response;  
import com.android.volley.VolleyError;  
import com.android.volley.toolbox.StringRequest;  
import com.android.volley.toolbox.Volley;  
  
import org.json.JSONException;  
import org.json.JSONObject;  
  
import java.util.HashMap;  
import java.util.Map;  
  
public class MainActivity extends AppCompatActivity {  
 Spinner s;  
 EditText input\_sqft,input\_bath,input\_bhk;  
 Button pred;  
 TextView tv;  
 String url = "https://bhpml.pythonanywhere.com/predict";  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 String[] arraySpinner = new String[] {  
 "1st block jayanagar", "1st phase jp nagar", "2nd phase judicial layout", "2nd stage nagarbhavi", area", "bommenahalli", "brookefield", ""yeshwanthpur"};// add all location from datset  
 s = findViewById(R.id.*spinner*);  
 ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,  
 android.R.layout.*simple\_spinner\_item*, arraySpinner);  
 adapter.setDropDownViewResource(android.R.layout.*simple\_spinner\_dropdown\_item*);  
 s.setAdapter(adapter);  
  
 input\_sqft = findViewById(R.id.*sqft*);  
 input\_bath = findViewById(R.id.*bath*);  
 input\_bhk = findViewById(R.id.*bhk*);  
 pred = findViewById(R.id.*Predictbtn*);  
 tv = findViewById(R.id.*show*);  
  
 pred.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 // hit the api --> Bolley  
 StringRequest rs = new StringRequest(Request.Method.*POST*, url,  
 new Response.Listener<String>() {  
 @Override  
 public void onResponse(String response) {  
 try {  
 JSONObject jsonObject = new JSONObject(response);  
 //String res = jsonObject.getString("Price");  
 //tv.setText("estimated price is:"+res+" Lakh");  
 double price = Double.*parseDouble*(jsonObject.getString("Price"));  
 String formattedPrice = String.*format*("estimated price is: %.2f Lakh", price);  
 tv.setText(formattedPrice);  
 } catch (JSONException e) {  
 throw new RuntimeException(e);  
 }  
 }  
 },  
 new Response.ErrorListener() {  
 @Override  
 public void onErrorResponse(VolleyError error) {  
 Toast.*makeText*(MainActivity.this, error.getMessage().toString(), Toast.*LENGTH\_SHORT*).show();  
 }  
 }){  
 @Override  
 protected Map<String, String> getParams() {  
 Map<String, String> params = new HashMap<String, String>();  
 params.put("loc", s.getSelectedItem().toString());  
 params.put("sqft", input\_sqft.getText().toString());  
 params.put("bath", input\_bath.getText().toString());  
 params.put("bhk", input\_bhk.getText().toString());  
  
 return params;  
 }  
 };  
  
 RequestQueue queue = Volley.*newRequestQueue*(MainActivity.this);  
 queue.add(rs);  
  
 }  
 });

}

}

XML Code:

Activity\_main:

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"  
 android:orientation="vertical">  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="House price predictor"  
 android:textSize="25dp"  
 android:textAlignment="center"  
 android:layout\_marginTop="25dp"  
 />  
 <EditText  
 android:id="@+id/sqft"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Enter Total Square feet"  
 android:layout\_marginTop="25dp"  
 android:textAlignment="center"  
 />  
 <EditText  
 android:id="@+id/bath"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Enter No of bathroom"  
 android:textAlignment="center"  
 android:layout\_marginTop="25dp"  
 />  
 <EditText  
 android:id="@+id/bhk"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Enter BHK"  
 android:textAlignment="center"  
 android:layout\_marginTop="25dp"  
 />  
 <Spinner  
 android:id="@+id/spinner"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textAlignment="center"  
 android:layout\_marginTop="20dp"  
 android:autofillHints="Location"/>  
  
<Button  
 android:id="@+id/Predictbtn"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Predict"  
 android:layout\_marginTop="20dp"  
 android:layout\_gravity="center"  
 android:textSize="20dp"  
 />  
 <TextView  
 android:id="@+id/show"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="20dp"  
 android:textSize="25dp"  
 android:textAlignment="center"  
  
 />  
  
</LinearLayout>

Python code to Build API:

First save model 🡪 bhp\_model.pkl (Download from my [GitHub](https://github.com/amansetu03/))

app.py:

from flask import Flask,request,jsonify

from api import predict\_price

app = Flask(\_\_name\_\_)

@app.route('/')

def home():

    return "hello world"

@app.route('/predict',methods = ['POST'])

def predict():

    loc = request.form.get('loc')

    sqft = int(request.form.get('sqft'))

    bath = int(request.form.get('bath'))

    bhk = int(request.form.get('bhk'))

    price = predict\_price(loc,sqft,bath,bhk)

    result = {"Price":str(price)}

    return jsonify(result)

if \_\_name\_\_ == "\_\_main\_\_":

    app.run(host='http://codeplace.pythonanywhere.com', port=2545,debug=True)

api.py

import pickle

import numpy as np

import pandas as pd

#plz change the value of X with all location from dataset.

X = pd.DataFrame(columns=["total\_sqft", "bath", "bhk", "1st block jayanagar"])

model = pickle.load(open('model/bhp\_model.pkl','rb'))

def predict\_price(location, sqft, bath, bhk):

    loc\_index = -1

    if location in X.columns:

        loc\_index = np.where(X.columns == location)[0][0]

    x = np.zeros(len(X.columns))

    x[0] = sqft

    x[1] = bath

    x[2] = bhk

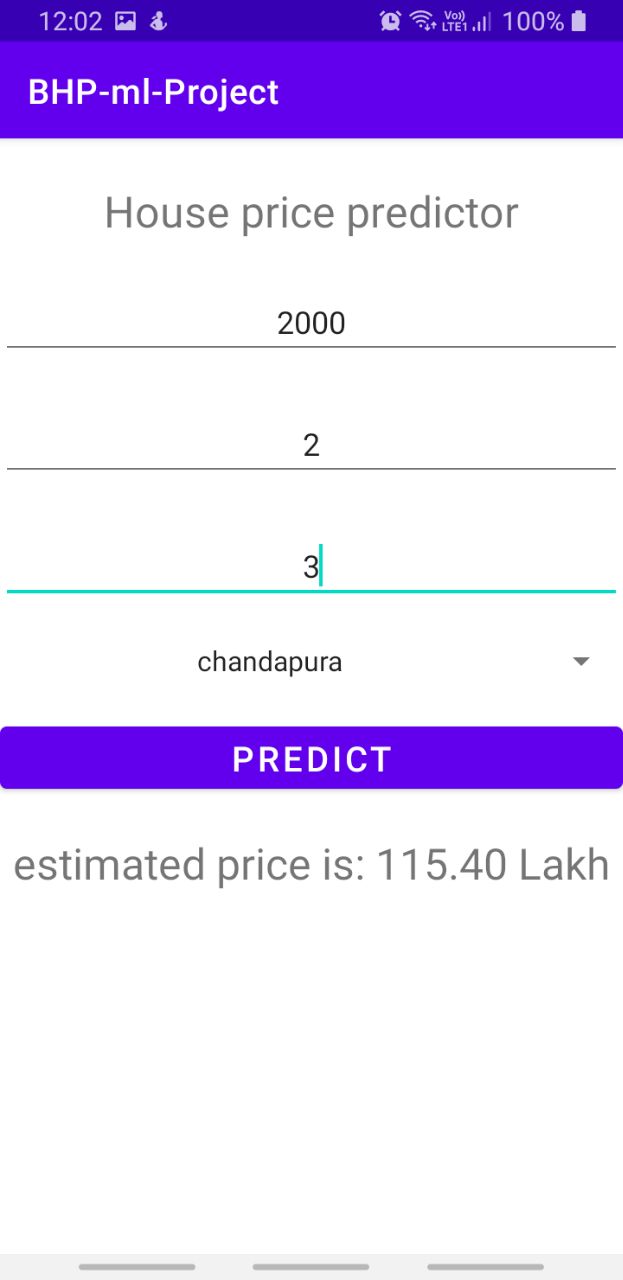
    if loc\_index >= 0:

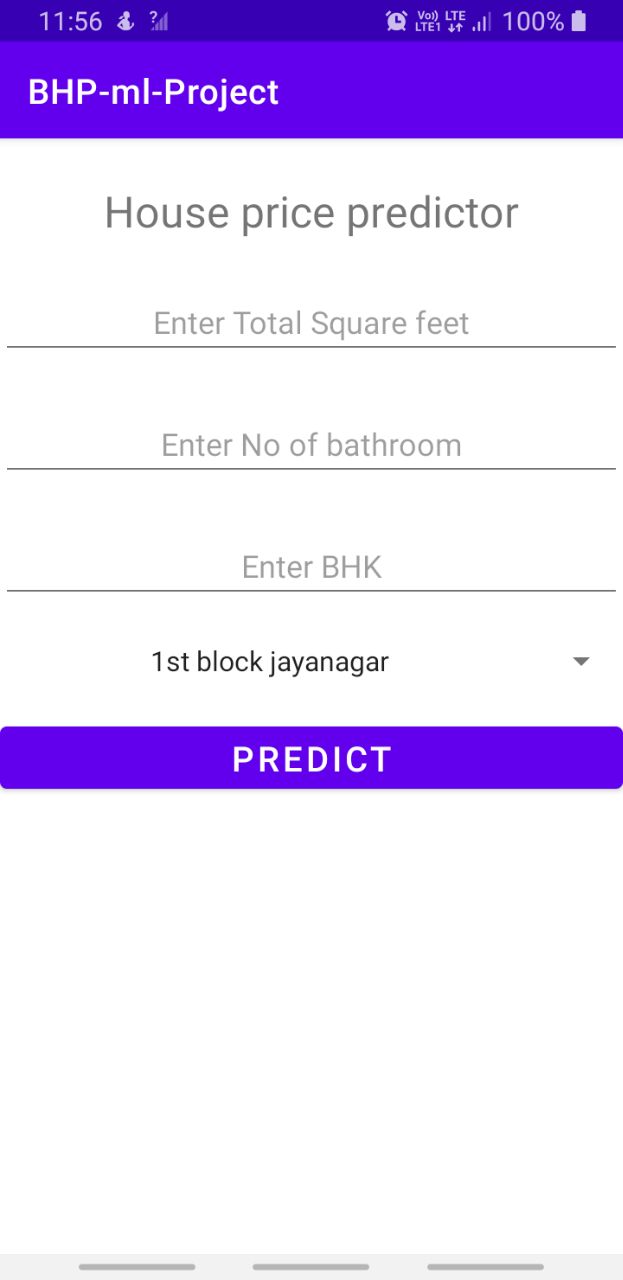
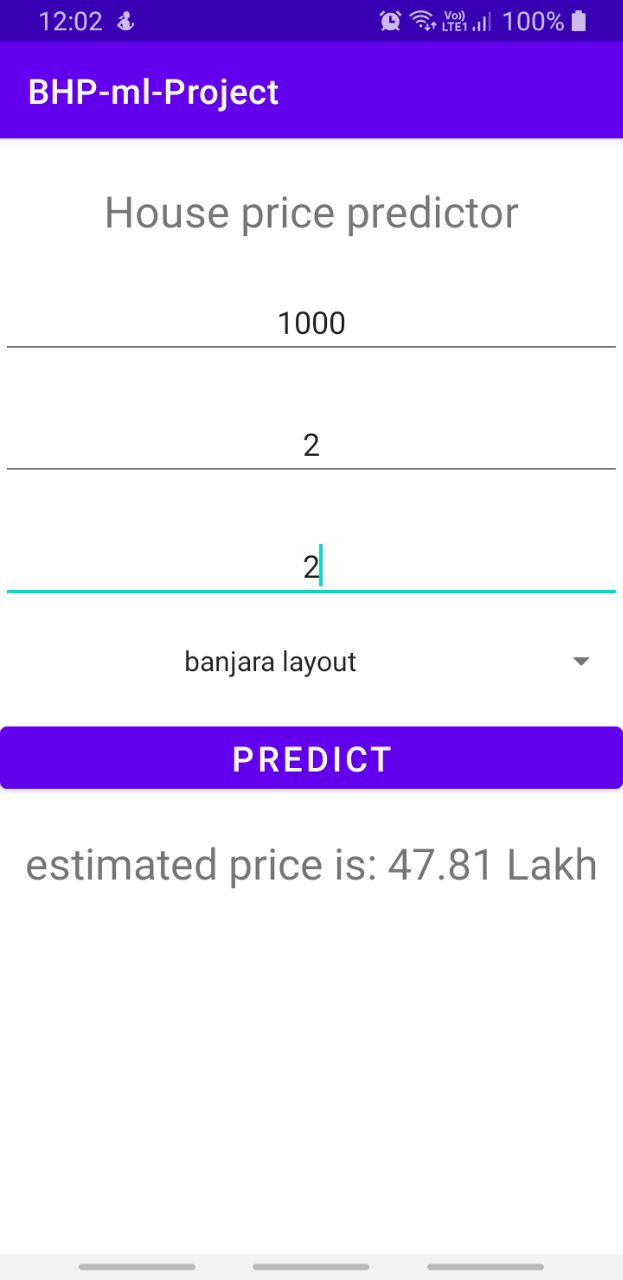
        x[loc\_index] = 1

    return model.predict([x])[0]

# Chapter 4

Results:

• Snapshot 1 • Snapshot 2 • Snapshot 3



# Chapter 5

Conclusion:

In conclusion, the House Price Prediction ML App is a game-changer in the world of real estate. It uses smart computer programs to guess how much a house should cost, helping people make better decisions when buying, selling, or investing in real estate.

The app is easy to use, and it gives you a good estimate of a house's price when you tell it about the house's features. It's like having a real estate expert in your pocket, and it helps you make smart choices in the ever-changing real estate market.

Future Work:

Looking ahead, we have exciting plans to make the app even better:

1. More Data: We'll use more information to make our predictions even more accurate.
2. More Places: We'll make the app work in more places with different real estate markets.
3. Listening to You: We want to hear from you to make the app easier to use and more helpful.
4. Smarter Model: We'll keep teaching our smart technology to keep up with the changing real estate world.
5. Partnerships: We'll work with real estate companies and others to give you a complete real estate experience.
6. Smart Recommendations: We'll add suggestions to help you make even better choices.
7. More Devices: We'll make the app work on different types of devices so more people can use it.

The future of the House Price Prediction ML App is full of exciting possibilities, making it an even more valuable tool for navigating the world of real estate.