A

Major Project

On

IMAGE BASED APPRAISAL OF REAL ESTATE PROPERTIES

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

BADAVATH SURESH (187R1A05D1)

KALYANI YASH (187R1A05F4)

KISTAPURAM SHIVANI (187R1A05F7)

Under the Guidance of

Dr. G. SomaSekhar

(Associate Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC,NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi)

Recognized Under Section 2(f) & 12(B) of the UGC Act.1956,

Kandlakoya (V), Medchal Road, Hyderabad-501401.

2018-2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled "**IMAGE BASED APPRAISAL OF REAL ESTATE PROPERTIES**" being submitted by **BADAVATH SURESH (187R1A05D1), KALYANI YASH (187R1A05F4) & KISTAPURAM SHIVANI (187R1A05F7)** in partial fulfillment of the requirements for the award of the degree of B. Tech in Computer Science and Engineering of the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by him/her under our guidance and supervision during the year 2021-2022.

The results embodied in this thesis have not been submitted to any other University or Institute for the awards of any degree in diploma.

Dr. G. SomaSekhar
Associate Professor
INTERNAL GUIDE

Dr. A. Raji Reddy
DIRECTOR

Dr. K. Srujan Raju
HOD

EXTERNAL EXAMINER

Submitted on viva voce Examination held on _____

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project. We take this opportunity to express my profound gratitude and deep regard to my guide,

Dr. G. SomaSekhar, Assoc. Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Project Review Committee (PRC) **Mr. J. Narasimha Rao, Dr. T.S. Mastan Rao, Dr. Suwarna Gothane, Mr. A. Uday Kiran, Mr. A. Kiran Kumar, Mrs. G. Latha** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to the Head of the Department **Dr. K. Srujan Raju** for providing excellent infrastructure and a nice atmosphere for completing this project successfully.

We are obliged to our Director **Dr. A. Raji Reddy** for being cooperative throughout the course of this project. We would like to express our sincere gratitude to our Chairman **Sri. Ch. Gopal Reddy** for his encouragement throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed and who are contributing to this project, was vital for the success of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement without which this assignment would not be possible. We sincerely acknowledge and thank all those who gave support directly and indirectly in completion of this project.

BADAVATH SURESH (187R1A05D1)

KALYANI YASH (187R1A05F4)

KISTAPURAM SHIVANI (187R1A05F7)

ABSTRACT

Real estate appraisal, is the process of estimating the price for real estate properties, which is crucial for both buyers and sellers. Real estate appraisal is based for negotiation and transaction. Traditionally, the repeat sales model has been widely adopted to estimate real estate price. However, it depends the design and calculation of a complex economic related index, which is challenging to estimate accurately. Today, real estate brokers provide easy access to detailed online information on real estate properties to their clients. We are interested in estimating the real estate price from these large amounts of easily accessed data. In particular, we analyze the prediction power of online house pictures, which is one of the key factors for online users to make a potential visiting decision. The development of robust computer vision algorithms makes the analysis of visual content possible. In this work, we employ a Recurrent Neural Network (RNN) to predict real estate price using the state-of-the-art visual features. The experimental results indicate that our model outperforms several of other state-of-the-art baseline algorithms in terms of both mean absolute error (MAE) and mean absolute percentage error (MAPE).

LIST OF FIGURES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

1. INTRODUCTION

1. INTRODUCTION

INTRODUCTION

Real estate appraisal, is the process of estimating the price for real estate properties, it is crucial for both buys and sellers as it is the basis for negotiation and transaction. Real estate plays a vital role in all aspects of our contemporary society. In a report published by the European Public Real Estate Association (EPRA http://alturl.com/7snxx), it was shown that real estate in all its forms accounts for nearly 20% of the economic activity. Therefore, accurate prediction of real estate prices or the trends of real estate prices help governments and companies make informed decisions.

1.1 PROBLEM DEFINITION

For most of the working class, housing has been one of the largest expenses. A right decision on a house, which heavily depends on their judgment on the value of the property, can possibly help them save money or even make profits from their investment in their homes. From this perspective, real estate appraisal is also closely related to people's lives. Current research from both estate industry and academia has reached the conclusion that real estate value is closely related to property infrastructure, traffic, online user reviews and so on. Generally speaking, there are several different types of appraisal values. In particular, we are interested in the market value, which refers to the trade price in a competitive auction setting. Typically, a buyer will look at those pictures to obtain a general idea of the overall property in a selected area before making his next move. Both real estate industry professionals and researchers have relied on a number of factors, such as economic index, house age, history trade and neighborhood environment and so on to estimate the price. Indeed, these factors have been proved to be related to the house price, which is quite difficult to estimate and sensitive to many different human activities. Therefore, researchers have devoted much effort in building a robust house price index. In addition, quantitative features including Area, Year, Stores, Rooms and Centre are also employed to build neural network models for estimating house prices.

However, pictures, which is probably the most important factor on a buyer's initial decision making process. One advantage with images and videos is that they act like universal languages. People with different backgrounds can easily understand the main content of an image or video. In the real estate industry, pictures can easily tell people exactly how the house looks like, which is impossible to be described in many ways using language. For the given house pictures, people can easily have an overall feeling of the house.

1.2 PURPOSE AND NEED

Today's computational infrastructure is also much cheaper and more powerful to make the analysis of computationally intensive visual content analysis feasible. Deep learning has enabled robust and accurate feature learning, in this project we are interested in solving the challenging real estate appraisal problem using deep visual features. In particular, for images related tasks, Convolutional Neural Network (CNN) are widely used due to the usage of convolutional layers. It takes into consideration the locations and neighbors of image pixels, which are important to capture useful features for visual tasks. Convolutional Neural Networks have been proved very powerful in solving computer vision related tasks.

1.3 PROJECT OVERVIEW

- Incorporating the pictures for the task of real estate price estimation.
- Using the visual features in the image, which are a reflection of a real estate property, can help estimate the real estate price.
- Quantify and analysing the impact of visual content on real estate price estimation.
- Assessing the attributes for the possibility of our work to the Convolutional Neural Networks (CNNs).
- Random walks to generate house sequences according to the locations of each house.
- Recurrent Neural Networks (RNNs) to predict real estate properties and achieve accurate results.

2. LITERATURE SURVEY

2. LITERATURE SURVEY

Real estate appraisal has been studied by both real estate industrial professionals and academia researchers. Earlier work focused on building price indexes for real properties. The seminal work in built price index according to the repeat prices of the same property at different times. They employed regression analysis to build the price index, which shows good performances. Another widely used regression model, Hedonic regression, is developed on the assumption that the characteristics of a house can predict its price. However, it is argued that the Hedonic regression model requires more assumptions in terms of explaining its target. They also mentioned that for repeat sales model, the main problem is lack of data, which may lead to failure of the model. Recent work in [9] employed locations and sale price series to build an autoregressive component. Their model is able to use both single sale homes and repeat sales homes, which can offer a more robust sale price index. More studies are conducted on employing feed forward neural networks for real estate appraisal. However, their results suggest that neural network models are unstable even using the same package with different run times. The performance of neural networks are closely related to the features and data size. Recently, Kontrimas and Verikas empirically studied several different models on selected 12 dimensional features, e.g. type of the house, size, and construction year. Their results show that linear regression outperforms neural network on their selected 100 houses. More recent studies in propose a ranking objective, which takes geographical individual, peer and zone dependencies into consideration. Their method is able to use various estate related data, which helps improve their ranking results based on properties' investment values. Furthermore, the work in studied online user's reviews and mobile users' moving behaviors on the problem of real estate ranking. Their proposed sparsity regularized learning model demonstrated competitive performance. In particular, our model does not use the meta data of a house (e.g. size, number of rooms, and construction year). We intend to utilize the location information in a novel way such that our model is able to use the state-of-the-art deep learning for feature extraction (Convolutional Neural Network) and model learning (Recurrent Neural Network).

2.1 REFERENCE PAPERS

2.1.1 PAPER – 1

Authors	Yanjie Fu, Hui Xiong, Yong Ge, Zijun Yao, Yu Zheng, Zhi-Hua Zhou.
Paper Title & Year of Publication	Exploiting Geographic dependencies for Real Estate Appraisal. 2014.
Methodology	Cluster Ranking was proposed for leveraging the mutual enforcement of ranking and clustering power.
Drawback	It exploit geographic individual, peer, and zone dependencies in a probabilistic ranking model.

2.1.2 PAPER - 2

Authors	A. Shinde, N. Dange, N. Patane, S. Gholap, V. Beera.
Paper Title & Year of Publication	Real Estate Properties Assessment using Deep Neural Network. 2019
Methodology	Systematic method to derive a layered knowledge graph and design a structured Deep Neural Network (DNN).
Drawback	The network time requires fewer data points for training which slows the effective assessing of real estate values.

2.1.3 PAPER - 3

Authors	Junchi Bin, Bryan Gardiner, Eric Li, Zheng Liu.
Paper Title & Year of Publication	Peer-Dependence Valuation Model for Real Estate Appraisal. 2019.
Methodology	A peer-dependence valuation model (PDVM) is proposed.
Drawback	Model outperforms the other state-of-the-art machine learning models.

2.1.4 PAPER - 4

Authors	S. Elnagar, Manoj A. Thomas.
Paper Title & Year of Publication	Image-Based Appraisal for Real Estate Using Mask Region Convolutional Networks. 2019.
Methodology	Enhanced R-CNN network called Mask R-CNN to evaluate the condition of each property image.
Drawback	The system is expected to be an integral module to existing real estate appraisal systems to enhance the appraisal process.

3. PROPOSED SYSTEM

3. PROPOSED SYSTEM

We intend to employ the pictures for the task of real estate price estimation. We want to know whether visual features, which are a reflection of a real estate property, can help estimate the real estate price. Intuitively, if visual features can characterize a property in a way similar to human beings, we should be able to quantify the house features using those visual responses. Real estate properties are closely related to the neighborhood. In this work, we develop algorithms which only rely on 1) the neighbor information and 2) the attributes from pictures to estimate real estate property price. To preserve the local relation among properties we employ a novel approach, which employs random walks to generate house sequences. In building the random walk graph, only the locations of houses are utilized. In this way, the problem of real estate appraisal has been transformed into a sequence learning problem. Recurrent Neural Network (RNN) is particularly designed to solve sequence related problems. We deploy RNN to learn regression models on the transformed problem. The main contributions of our work are as follows: To the best of our knowledge, we are the first to quantify the impact of visual content on real estate price estimation. We attribute the possibility of our work to the newly designed computer vision algorithms, in particular Convolutional Neural Networks (CNNs). We employ random walks to generate house sequences according to the locations of each house. In this way, we are able to transform the problem into a novel sequence prediction problem, which is able to preserve the relation among houses. We employ the novel Recurrent Neural Networks (RNNs) to predict real estate properties and achieve accurate results.

3.1 PROPOSED OUTCOMES AND OBJECTIVES

- Incorporating the pictures for the task of real estate price estimation.
- Using the visual features in the image, which are a reflection of a real estate property, can help estimate the real estate price.

- The Algorithms used rely on.
    1. The neighbor information.
    2. The attributes from pictures to estimate real estate property price.
- Quantify and analyzing the impact of visual content on real estate price estimation.
- Assessing the attributes for the possibility of our work to the Convolutional Neural Networks (CNNs).
- Random walks to generate house sequences according to the locations of each house.
- Recurrent Neural Networks (RNNs) to predict real estate properties and achieve accurate results.

4. SYSTEM ANALYSIS

4. SYSTEM ANALYSIS

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvements on the system. System analysis is a problem solving activity that requires intensive communication between the system users and system developers. System analysis of study is an important phase of any system development process. The system is studied to the minimum unit test detail and analyzed. The system analyst plays the role of an interrogator and wells deep into the working of the present system. The system is viewed as a whole and the inputs to the system are identified. The outputs from the organization are traced through the various processing that the inputs through in the organization. A detailed study of these processes must be made by various techniques like Interviews, Questionnaires etc.

The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now, the existing system is subjected to close study and the problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as a proposal. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. Analysis was done by keeping in mind the two modules of the project. The Analysis part of the project was the user module. Users of this application may or may not have much computer knowledge, so we mainly focused on our design, which had to be as user friendly as possible. The next important thing was to provide user level security. It was necessary to provide privacy to community members. Another thing was the appearance of the application; it had to be made pleasant and decent enough to attract the user. Last but not the least, was to provide the authorities to the administrators. Proper validations where to be implemented of the registration-form.

4.1 EXISTING SYSTEM

Current research from both estate industry and academia has reached the conclusion that real estate value is closely related to property infrastructure, traffic, online user reviews and so on. There are several different types of appraisal values. In particular, we are interested in the market value, which refers to the trade price in a competitive walrasian auction setting Traditionally, both real estate industry professionals and researchers have relied on a number of factors, such as economic index, house age, history trade and neighborhood environment and so on to estimate the price. The current algorithms are 1). Regression Models and 2). Deep Walk. Regression model has been employed to analyze real estate price index. We choose to use LASSO which is an l1-constrained regression model, as one of our baseline algorithms. Deep Walk is another way of employing random walks for unsupervised feature learning of graphs. We also use neighborhood graph with the same settings with the graph we built for generating sequences for B-LSTM. The learned features are also fed into a LASSO model for learning the regression weights.

4.2 LIMITATIONS OF EXISTING SYSTEM

- The existing system is quite difficult to estimate and sensitive to many different human activities.
- There are lot of difficult works have been done with the existing systems to measure the number of factors such as economic index, house age, history trade and neighborhood environment.
- Current research from both estate industry and academia has reached the conclusion that real estate value is closely related to property infrastructure, traffic online user Reviews and so on.
- Outdated
- Resource consuming.

4.3 PROPOSED SYSTEM

We intend to employ the pictures for the task of real estate price estimation. We want to know whether visual features, which are a reflection of a real estate property, can help estimate the real estate price. We should be able to quantify the house features using those visual responses. Real estate properties are closely related to the neighborhood. We develop algorithms which only rely on 1) the neighbor information and 2) the attributes from pictures to estimate real estate property price. To preserve the local relation among properties we employ a novel approach, which employs random walks to generate house sequences. In building the random walk graph, only the locations of houses are utilized. Recurrent Neural Network (RNN) is particularly designed to solve sequence related problems. Recently, RNNs have been successfully applied to challenging tasks including machine translation, image captioning, and speech recognition. We deploy RNN to learn regression models on the transformed problem. The main contributions of our work are as follows:

To the best of our knowledge, we are the first to quantify the impact of visual content on real estate price estimation. We attribute the possibility of our work to the newly designed computer vision algorithms, in particular Convolutional Neural Networks (CNNs). We employ random walks to generate house sequences according to the locations of each house. In this way, we are able to transform the problem into a novel sequence prediction problem, which is able to preserve the relation among houses. We employ the novel Recurrent Neural Networks (RNNs) to predict real estate properties and achieve accurate results.

4.4 ADVANTAGES OF PROPOSED SYSTEM

- A picture is worth a thousand words. One advantage with images and videos is that they act like universal languages. For the given house pictures, people can easily have an overall feeling of the house, e.g. what is the overall construction style, how the neighboring environment looks like. These high-level attributes are difficult to be quantitatively described**.**

- Map Based Location information are most commonly effective than the viewing in raw details. The most accurate details can be viewed in simple steps.
- The proposed algorithms are very effective than the existing algorithms such as LASSO and Deep Walk.

4.5 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the user.

Three key considerations involved in the feasibility analysis are

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

4.6 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on a project, which will give the user the best quality of life possible. One of the factors, which affect the development of a new system, is the cost it would require. The following are some of the important financial questions asked during preliminary investigation

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced or fewer costly errors.

4.7 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

4.8 BEHAVIORAL FEASIBILITY

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

4.9 SOFTWARE REQUIRMENTS

- WINDOWS 7 or high.
- Python Version 3.6 or more.
- Django
- MySQL (WAMP Server).
- Html, css, javascript.

4.10 HARDWARE REQUIRMENTS

- PIV 2.8 GHz Processor and Above.
- RAM 4 GB and Above.
- HDD 20 GB Hard Disk Space and Above.
- Intel i3 processor and above.

5. MODELLING

5. MODELLING

5.1 ALGORITHMS USED

5.1.1 RECURRENT NEURAL NETWORK

The idea behind RNNs is to make use of sequential information. In a traditional neural network we assume that all inputs (and outputs) are independent of each other. But for many tasks that's a very bad idea. If you want to predict the next word in a sentence you better know which words came before it. RNNs are called *recurrent* because they perform the same task for every element of a sequence, with the output being depended on the previous computations. Another way to think about RNNs is that they have a "memory" which captures information about what has been calculated so far. In theory RNNs can make use of information in arbitrarily long sequences, but in practice they are limited to looking back only a few steps (more on this later). Here is what a typical RNN looks like:

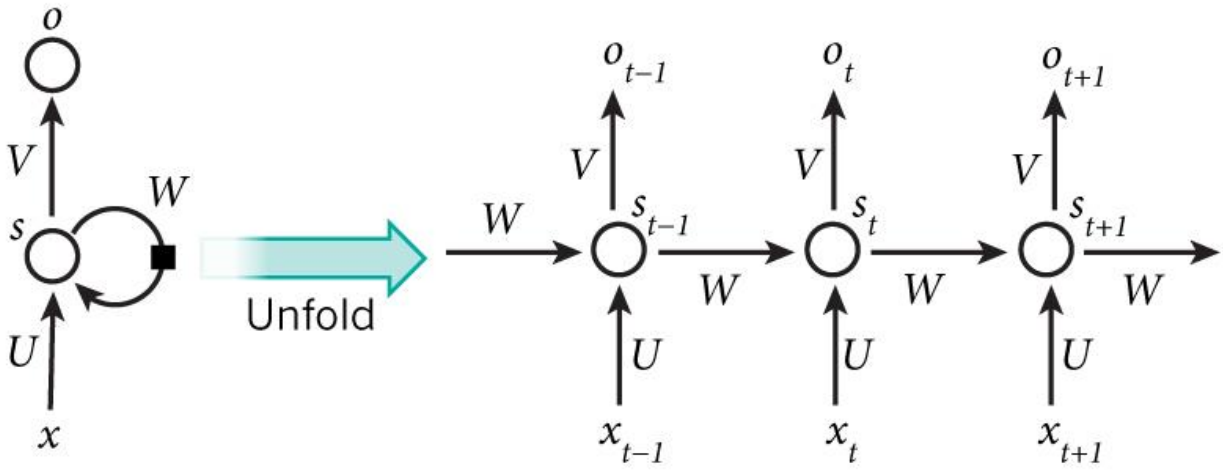


FIGURE 1: Recurrent Neural Network

The above diagram shows a RNN being *unrolled* (or unfolded) into a full network. By unrolling we simply mean that we write out the network for the complete sequence. For example, if the sequence we care about is a sentence of 5 words, the network would be unrolled into a 5-layer neural network, one layer for each word.

5.1.2 RANDOM WALK

One main feature of real estate properties is its location. In particular, for houses in the same neighborhood, they tend to have similar extrinsic features including traffic, schools and so on. We build an undirected graph G for all the houses collected, where each node vi represent the i-th house in our data set. The similarity sij between house hi and house hj is defined using the Gaussian kernel function, which is a widely used similarity measure,

$$sij = \exp(dist(hi, hj))/ 2\sigma 2$$

where dist(hi, hj) is the geodesic distance between house hi and hj . σ is the hyper-parameter, which controls the similarity decaying velocity with the increase of distance.

5.2 FRAMEWORK

We compare the proposed framework with the following algorithms.

1) Regression Model (LASSO): Regression model has been employed to analyze real estate price index [6]. Recently, the results in show that sparse regularization can obtain better performance in real estate ranking. Thus, we choose to use LASSO which is a l1-constrained regression model, as one of our baseline algorithms.

2) DeepWalk- Deepwalk is another way of employing random walks for unsupervised feature learning of graphs. The main approach is inspired by distributed word representation learning. In using DeepWalk, we also use neighborhood graph with the same settings with the graph we built for generating sequences for B-LSTM. The learned features are also fed into a LASSO model for learning the regression weights. Indeed, deepwalk can be thought as a simpler version of our algorithm, where only the graph structure are employed to learn features. Our framework can employ both the graph regression model.

3) Training a Multi-layer B-LSTM Model with the above mentioned similarity graph, we are able to generate sequences using random walks following the steps described in Algorithm 1. For each city, we randomly split the houses into training (80%) and testing set (20%). Next, we generate sequences using random walks on the training houses only to build our training sequences for Multi-layer B-LSTM.

5.3 INPUT AND OUTPUT DESIGN

5.3.1 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES

- Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
- It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
- When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant, Thus the objective of input design is to create an input layout that is easy to follow.

5.3.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

- Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.
- Select methods for presenting information.
- Create document, report, or other formats that contain information produced by the system.

OBJECTIVES

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

6. ARCHITECTURE

6. ARCHITECTURE

The system after careful analysis has been identified to be presented with the following

- Python.
- Django.

PYTHON

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespaces indentation to delimit code block rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer line of code than might be used in languages such as C++ or java. It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems, Python the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. Python is managed by the non-profit Python Software Foundation. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

DJANGO

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source. Django's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes re usability and "pluggability" of components, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings files and data models.

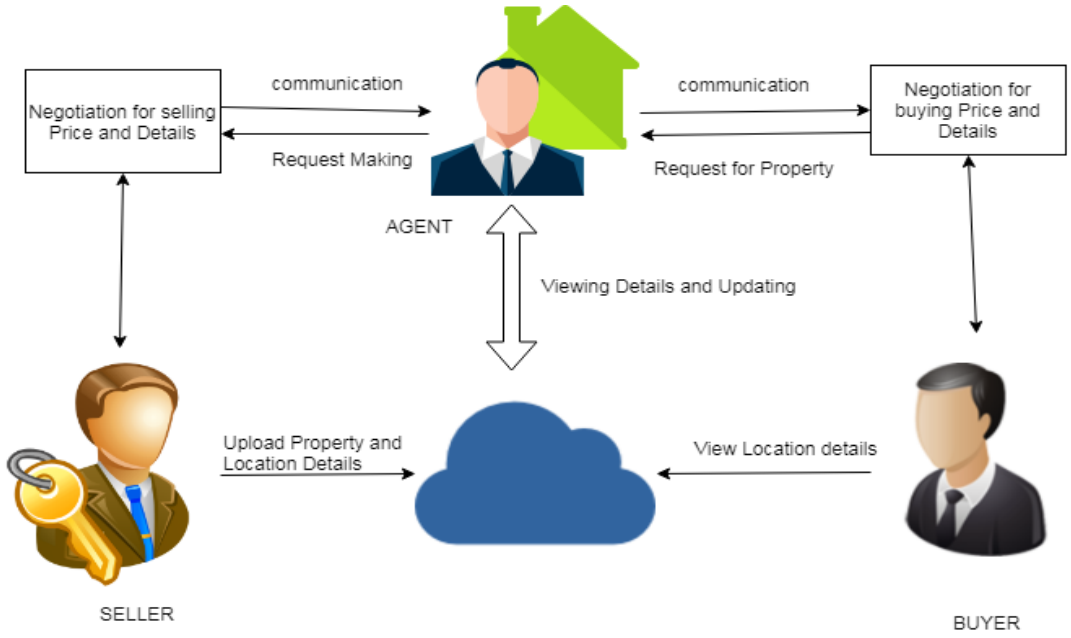


FIGURE 2: Architecture.

The project Architecture describes how the Seller, Buyer and Agent interact and all the components the seller, Buyer and agent can access.

6.1 MODULES

In this project there are four modules present as listed in the below

- Property Addition.
- Adding Location Details.
- Price Negotiation.
- Geometrical Analysis.

MODULES DESCRIPTION:

6.1.1 PROPERTY ADDITION

The property addition is the main initiative module for the project. Once authorized user login into the system, they can perform their activity as per their wish. In this module, User must have interested in selling the property which they own.

The Property details such as Location, Address, and Facilities that the households are need to add to the cloud where everything that seller uploads can viewable to buyer and agent.

6.1.2 ADDING LOCATION DETAILS

In this module user that is seller need to upload the details of their location as well as their neighboring facility location such as schools, colleges and medical etc.
In previous modules also user need to add the location that are into the raw typed format but here in this module we can upload the location details in maps and map formats. Spotting these locations can be very handy for agents or users to get to know about the details of property and neighboring details.

6.1.3 PRICE NEGOTIATION

This module is mainly designed for buyers and agents. Firstly, buyer sends the request to agents along with the cost of expectations and other query details about property. Once agents view the request from the buyer, Agent can decide the price according to the merit of location and both the buyer and seller. This module designed like chat. Dual way communication can be accomplished among the various users.

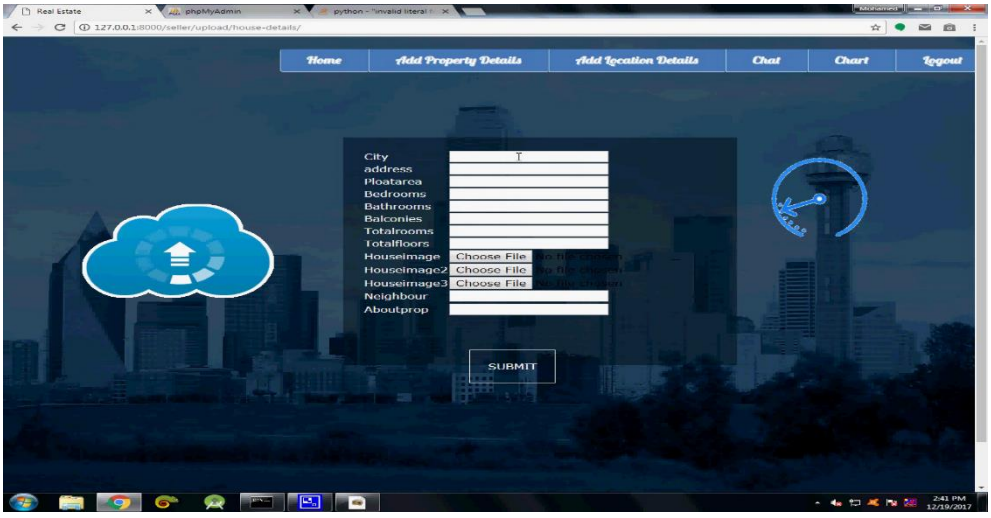


FIGURE 3: Property Addition.

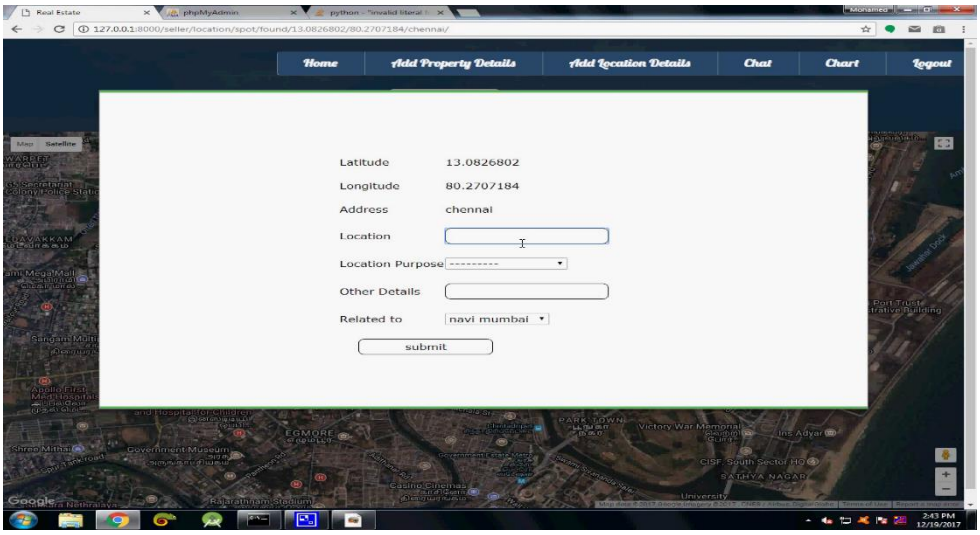


FIGURE 4: Adding Location Details

6.1.4 GEOMETRICAL ANALYSIS

The Geometrical analysis of given data set is done by charts. Here in this project there are two graphs have been plot between numbers of locations versus city. The pie chart and line charts are established in this project in order to analysis the data effectively.

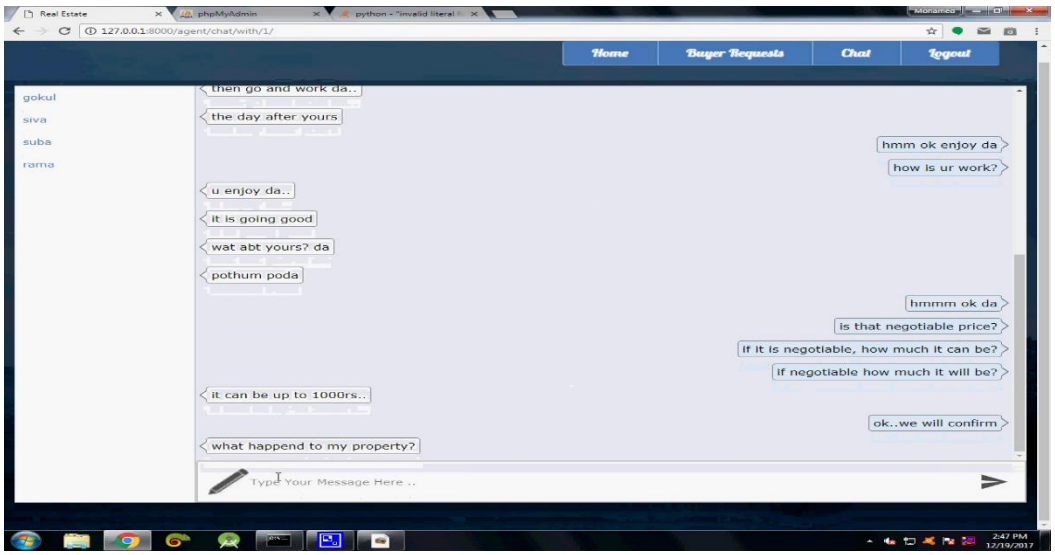


FIGURE 5: Price Negotiation.

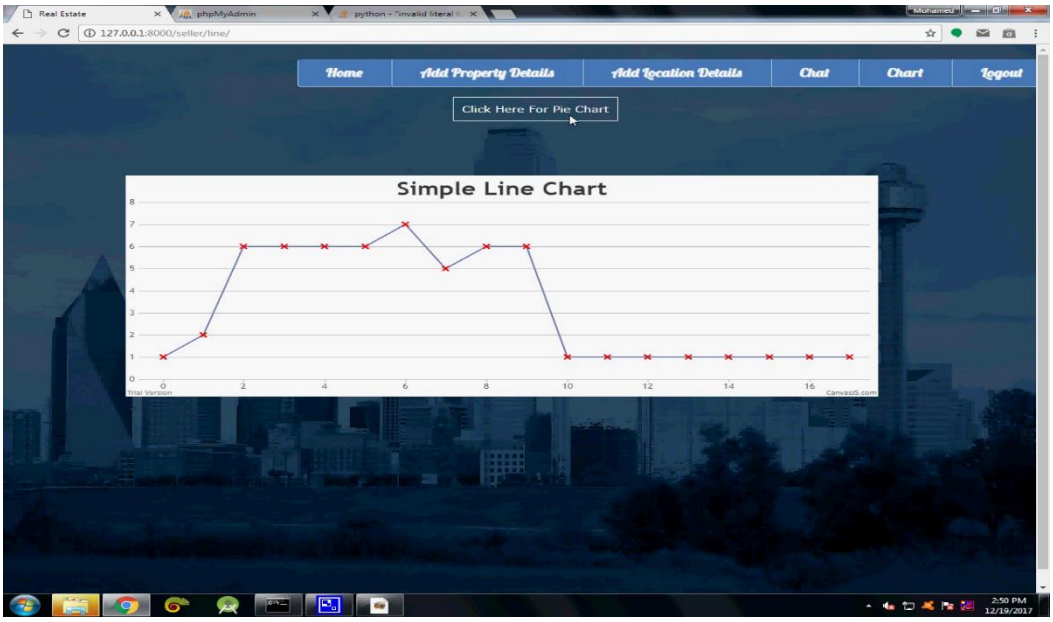


FIGURE 6: Geometric Line Chart.

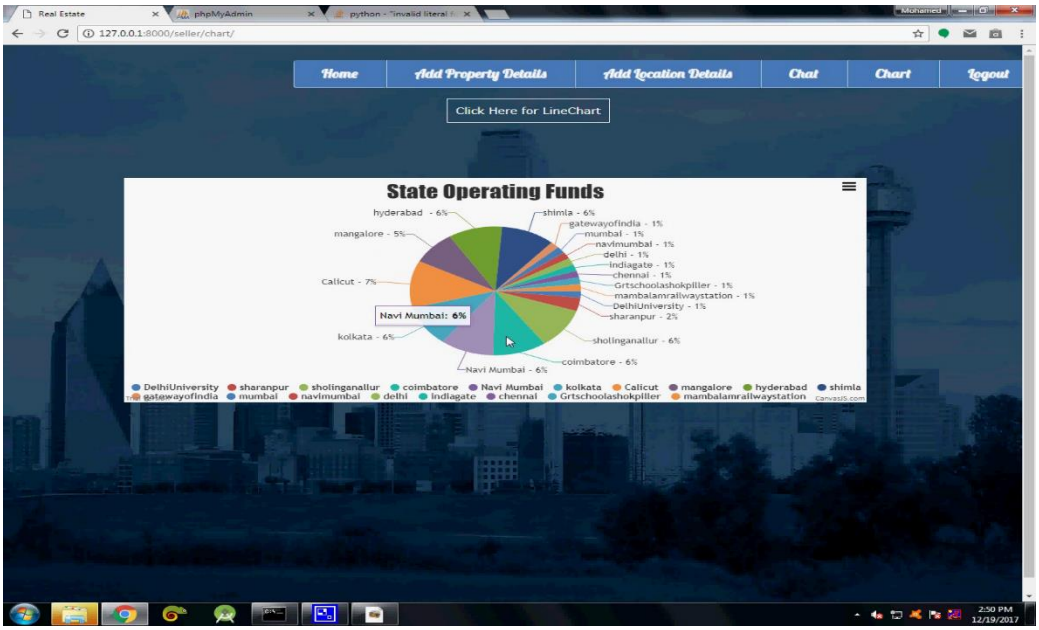


FIGURE 7: Geometric Pie Chart.

6.2 USE CASE DIAGRAM

A Usecase Diagram is a graphical depiction of a user's possible interactions with a system.

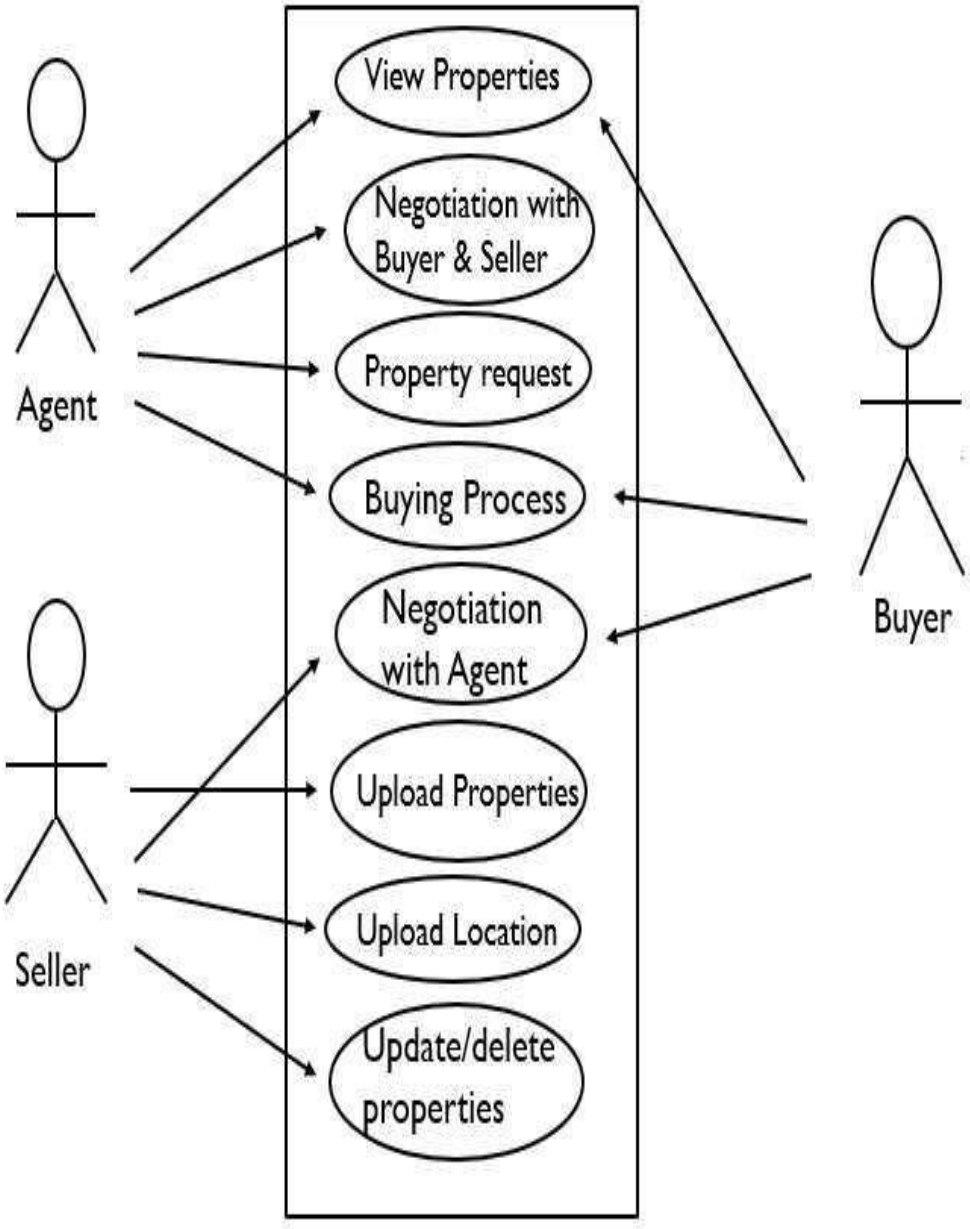


FIGURE 8: Usecase Diagram for Buyer, Seller and Agent component.

6.3 CLASS DIAGRAM

Class Diagram is a collection of classes and objects.

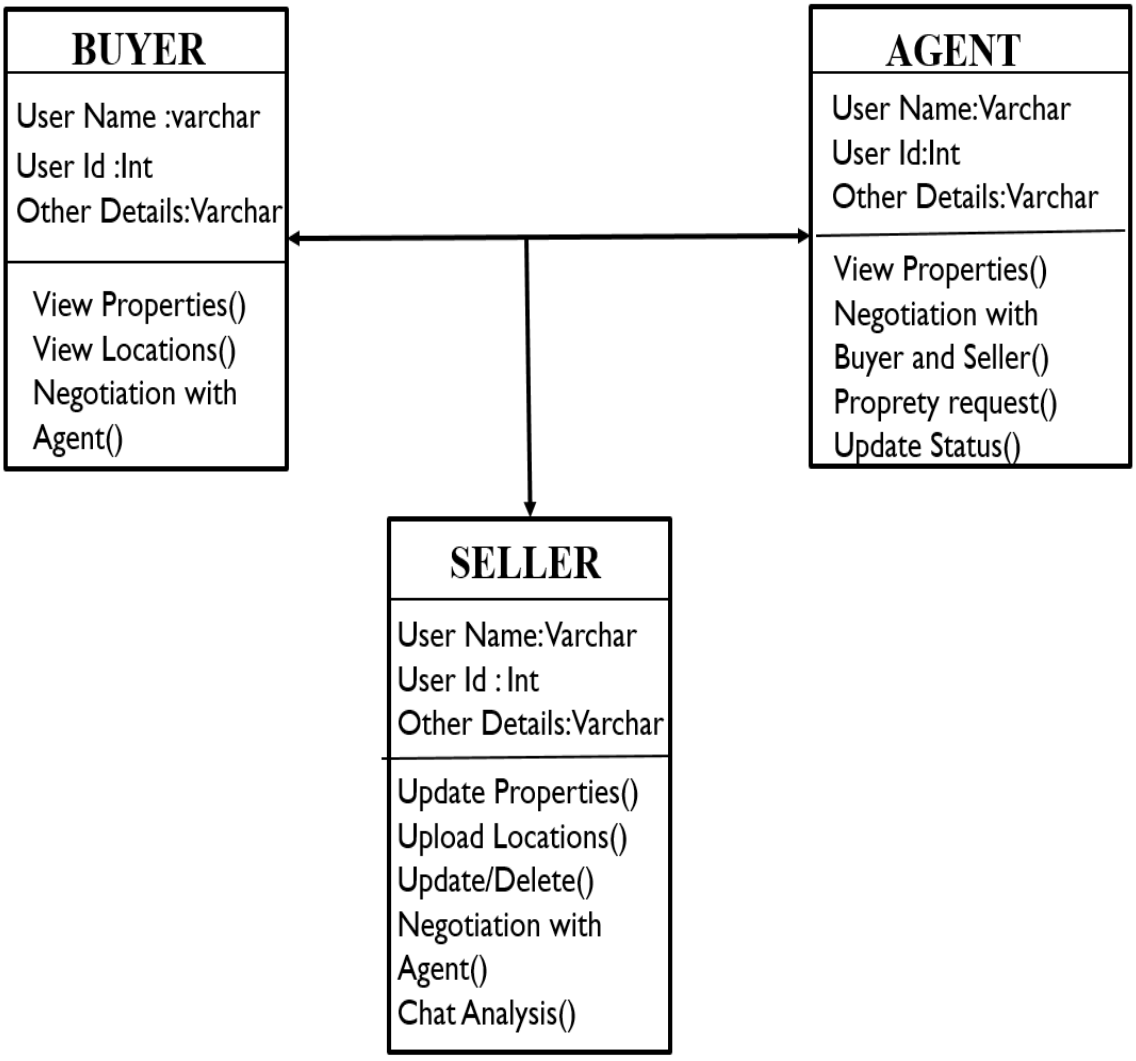


FIGURE 9: Class Diagram for Buyer, Seller and Agent component.

6.4 SEQUENCE DIAGRAM

The sequence diagram shows the sequence in which different tasks are being carried out by the actors.

AGENT

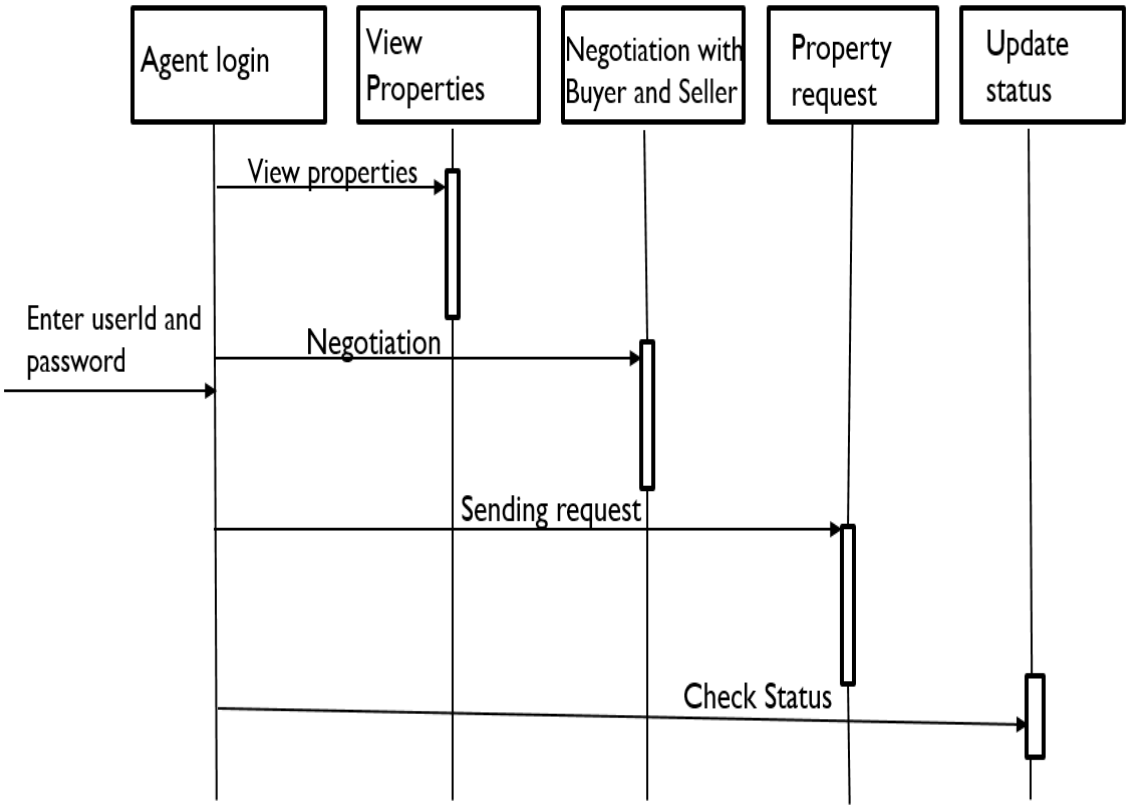


FIGURE 10: Sequence Diagram for Agent component.

BUYER

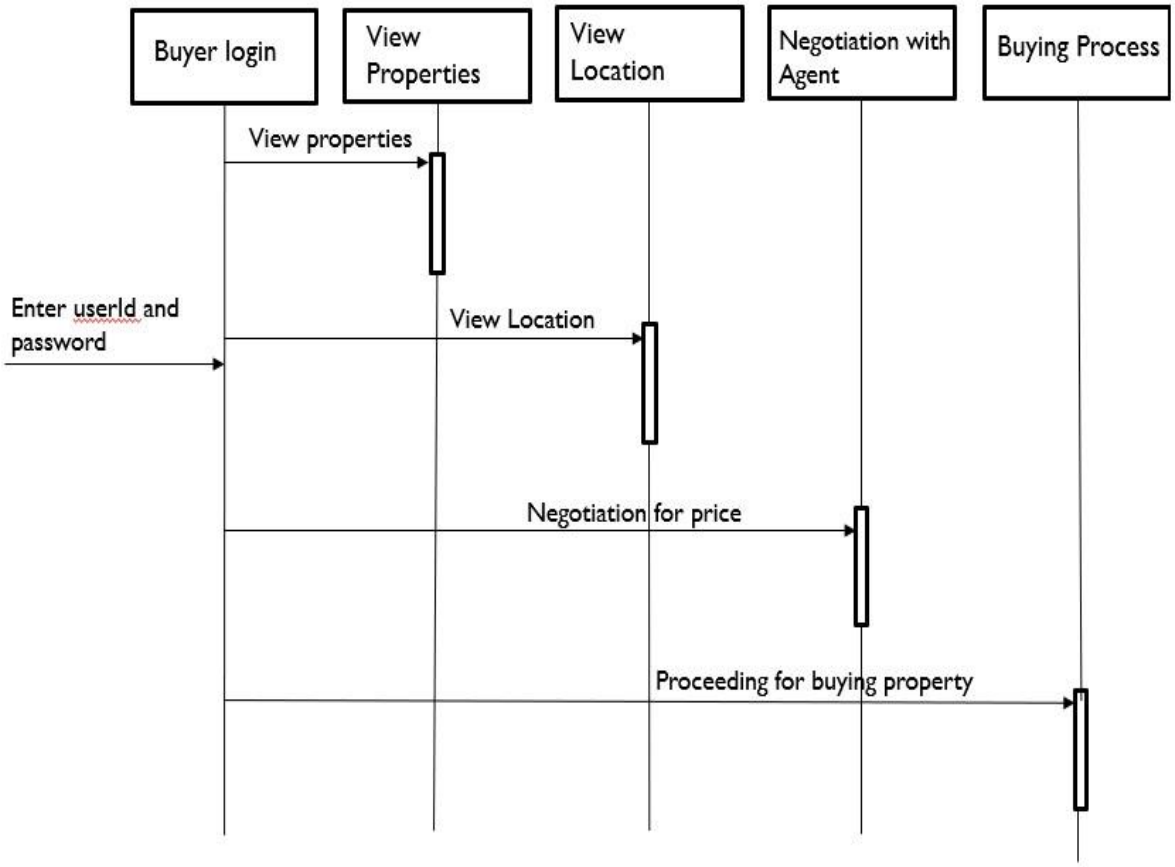


FIGURE 11: Sequence Diagram for Buyer component.

SELLER

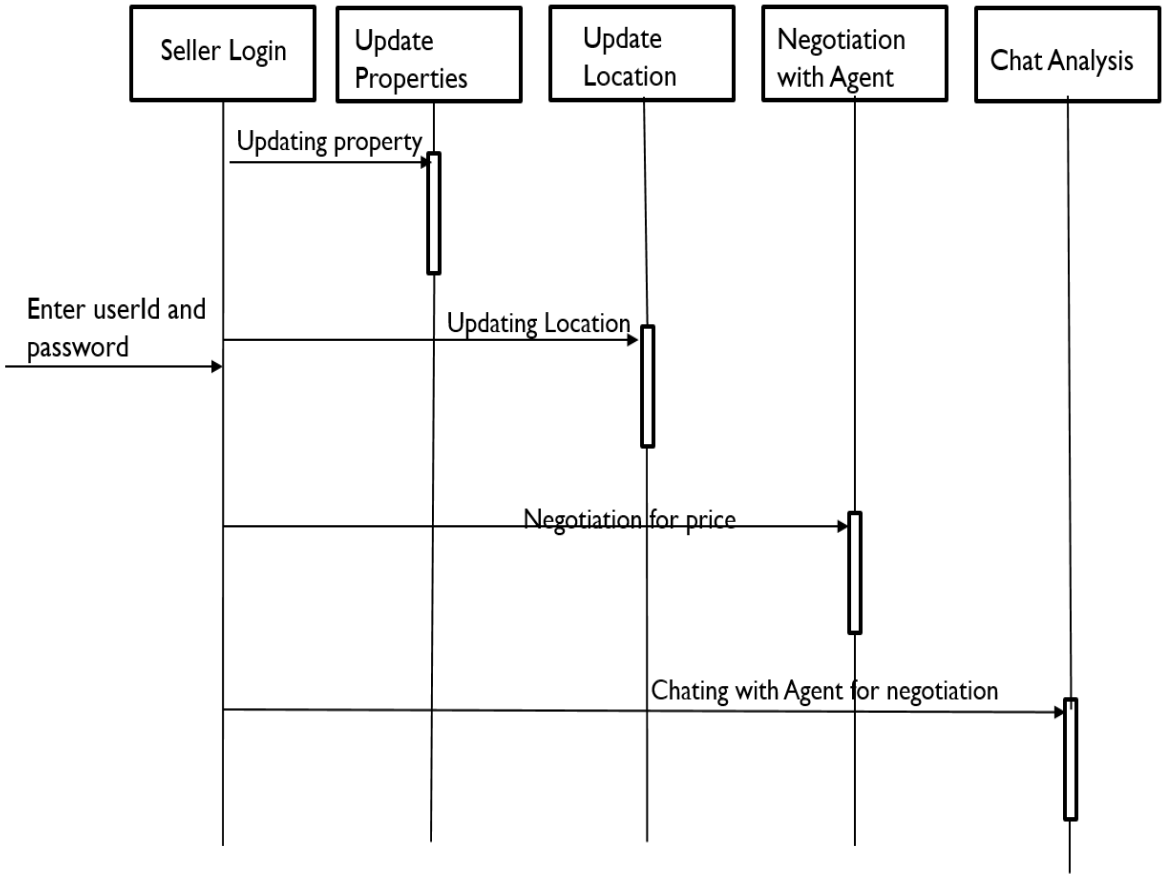


FIGURE 12: Sequence Diagram for Seller component.

.

6.5 ACTIVITY DIAGRAM

It describes the flow of activity states.

AGENT

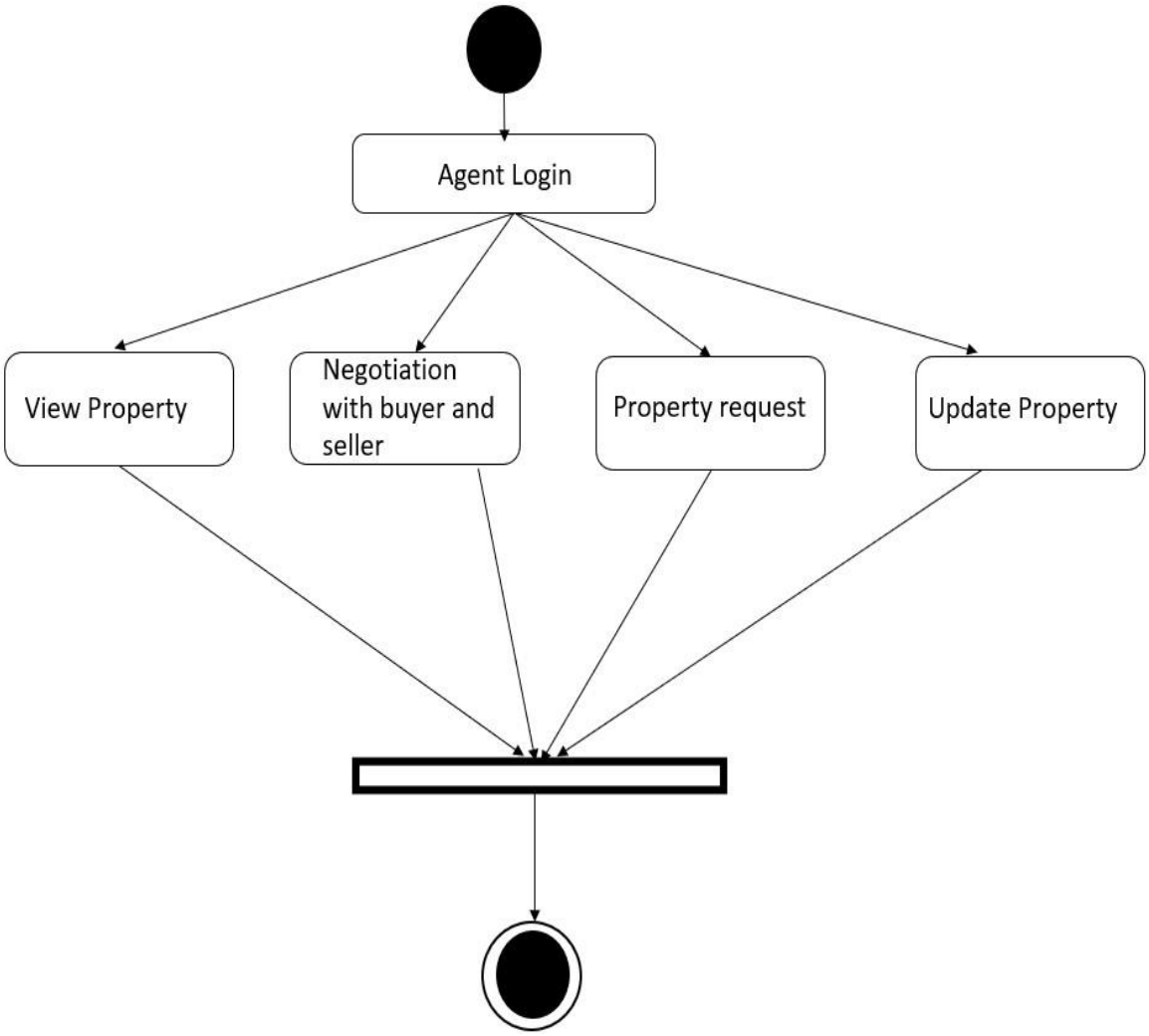


FIGURE 13: Activity Diagram for Agent component.

BUYER

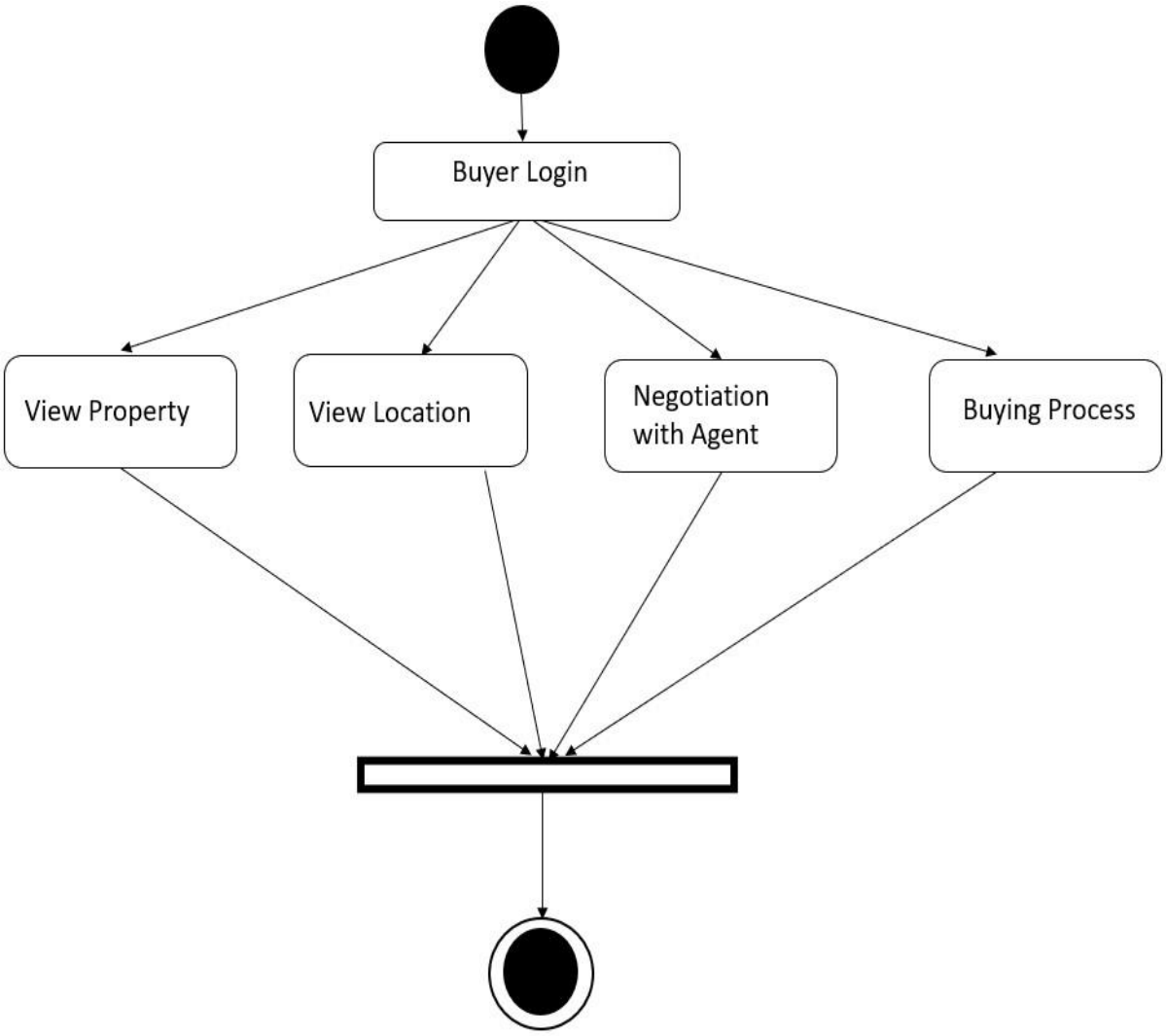


FIGURE 14: Activity Diagram for Buyer component.

SELLER

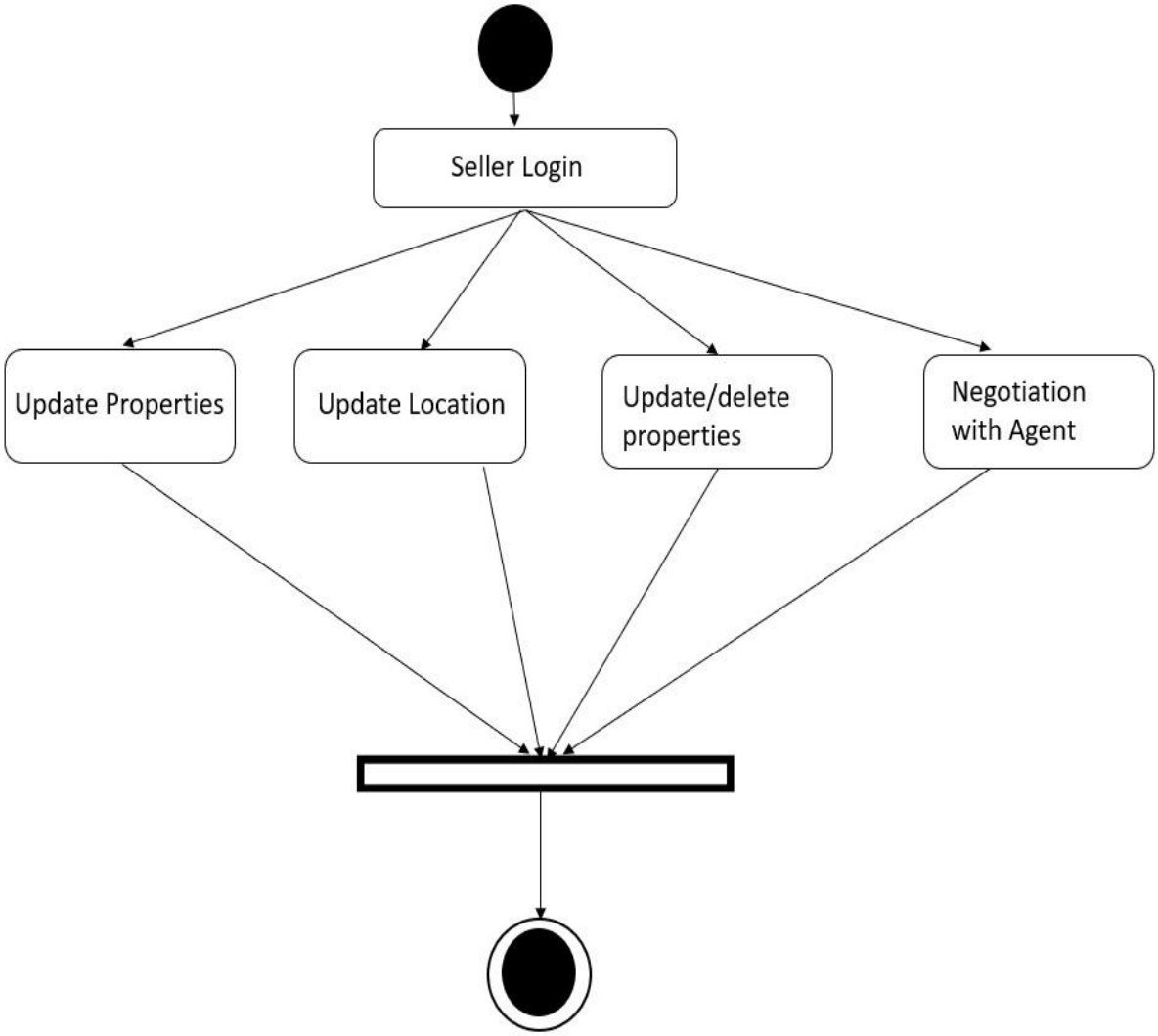


FIGURE 15: Activity Diagram for Seller component.

7. IMPLEMENTATION

7. IMPLEMENTATION

CODE:

```
from django.shortcuts import render, redirect, get_object_or_404
from django.contrib.auth.decorators import login_required
from forms import GetPropertyDetailsForm, GetLocationForm, LocationMapForm
from.models import PropertyDetails, LocationDetails
from geopy.geocoders import Nominatim
# from django.core.urlresolvers import reverse
from django.urls import reverse
import requests
import urllib
from django.db.models import Count
from agent.models import Chat
from agent.forms import ChatForm
from general.models import Profile
@login_required(login_url='/login/')
def home(request):
 proplist = PropertyDetails.objects.filter(user=request.user)
 data={}
 data['prop_list'] = proplist
    return render(request, 'seller/index.html', data)
@login_required(login_url='/login/')
def chart(request):
loc=LocationDetails.objects.all().values('propid','locationname').annotate(total=Count('prop
id')).order_by('propid')
    return render(request, 'seller/chart.html',{'loc':loc})
@login_required(login_url='/login/')
def line(request):
```

```
lc1=LocationDetails.objects.all().values('propid','locationname').annotate(total=Count('prop
id')).order_by('propid')
    return render(request, 'seller/linechart.html',{'loc':lc1})

@login_required(login_url='/login/')
def uploadPropertyDetails(request):
    form = GetPropertyDetailsForm(request.POST or None, request.FILES)
    if form.is_valid():
        propdet = form.save(commit=False)
        propdet.user = request.user
        propdet.save()
        return redirect('sellers:home')
    return render(request, 'seller/newproperty.html', {'form':form})

@login_required(login_url='/login/')
def locationspot(request):
    form = GetLocationForm(request.POST)
    if form.is_valid():
        address = form.cleaned_data.get('locaddress')
        """locationt = geolocator.geocode(addr)
        lati = locationt.latitude
        loni = locationt.longitude
        gmaps = GoogleMaps(AIzaSyCthblLTE___47JWZjTjKjEfCxq0ofxBCU)
        lat, lng = gmaps.address_to_latlng(address)"""
        api_key="AIzaSyCthblLTE___47JWZjTjKjEfCxq0ofxBCU"

api_response=requests.get('https://maps.googleapis.com/maps/api/geocode/json?address={
0}&key={1}'.format(address, api_key))
        api_response_dict = api_response.json()
        if api_response_dict['status'] == 'OK':
```

```
 latitude = api_response_dict['results'][0]['geometry']['location']['lat']
            longitude = api_response_dict['results'][0]['geometry']['location']['lng']
        return                              redirect('sellers:locationfound',lati=latitude,loni=longitude,
address=address.replace(" ",""))
    return render(request, 'seller/mylocation.html',{'form':form})
@login_required(login_url='/login/')
def locationspotfound(request, lati,loni,address):
    form = LocationMapForm(request.POST)
    item = PropertyDetails.objects.filter(user=request.user)
    data={}
    data['lati']=lati
    data['loni']=loni
    data['address']=address
    data['forms']=form
    data['items']=item
    if form.is_valid():
        locpdet = form.save(commit=False)
        locpdet.user = request.user
        locpdet.propid = get_object_or_404(PropertyDetails, pk=request.POST.get('item_id'))
        locpdet.locationname=address
        locpdet.lati=lati
        locpdet.longi=loni
        locpdet.save()
        return redirect('sellers:home')
    return render(request, 'seller/mylocation.html',data)


@login_required(login_url='/login/')
def nearestlocation(request):
    return render(request, 'seller/nearestlocation.html', {'form':form})
```

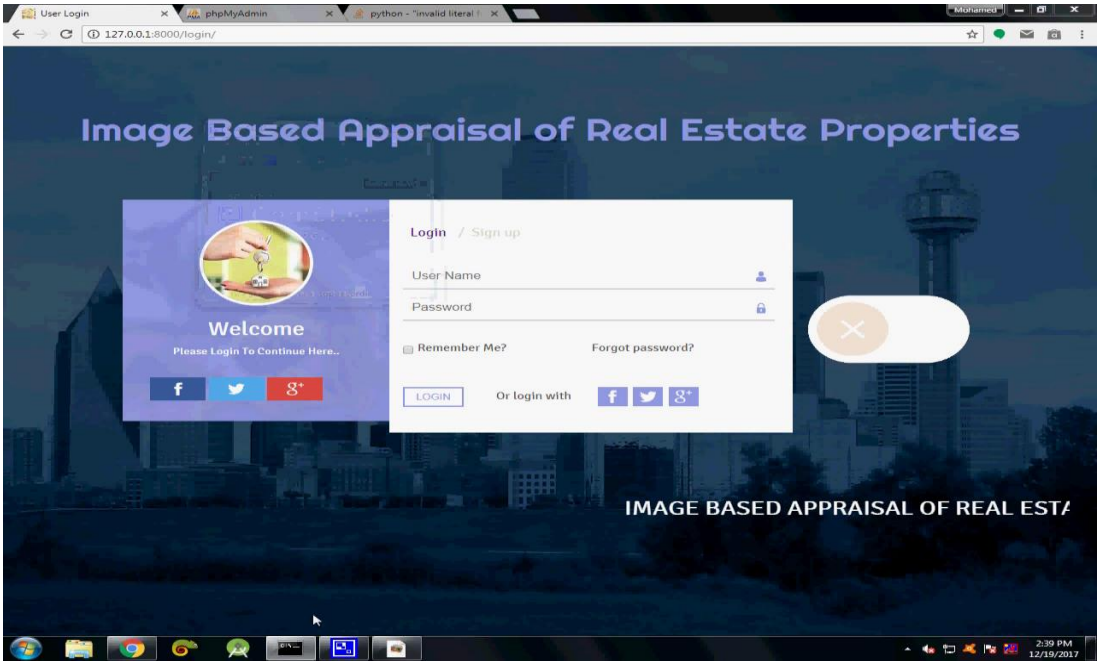
```
@login_required(login_url='/login/')
def aboutprop(request,pk):
    propdet=get_object_or_404(PropertyDetails,pk=pk)
    form =GetPropertyDetailsForm(request.POST or None, instance=propdet)
    if 'updatedet' in request.POST:
        if form.is_valid():
            form.save()
            return redirect('sellers:home')
    elif 'deletedet' in request.POST:
        propdet.delete()
        return redirect('sellers:home')
    return render(request, 'seller/viewproprtydetails.html',{'form':form,'propdet':propdet})

def aboutproploc(request,pk):
    locet = LocationDetails.objects.filter(user=request.user,propid=pk)
    lt=0.0
    lg=0.0
    cnt = 0
    for lo in locet:
        lt = lt+lo.lati
        lg = lg+lo.longi
        cnt = cnt+1
    if cnt != 0:
        latt=lt/cnt
        lonn=lg/cnt
    else:
        latt=0.0
        long=0.0
    data={}
    data['prop_list'] = locet
```

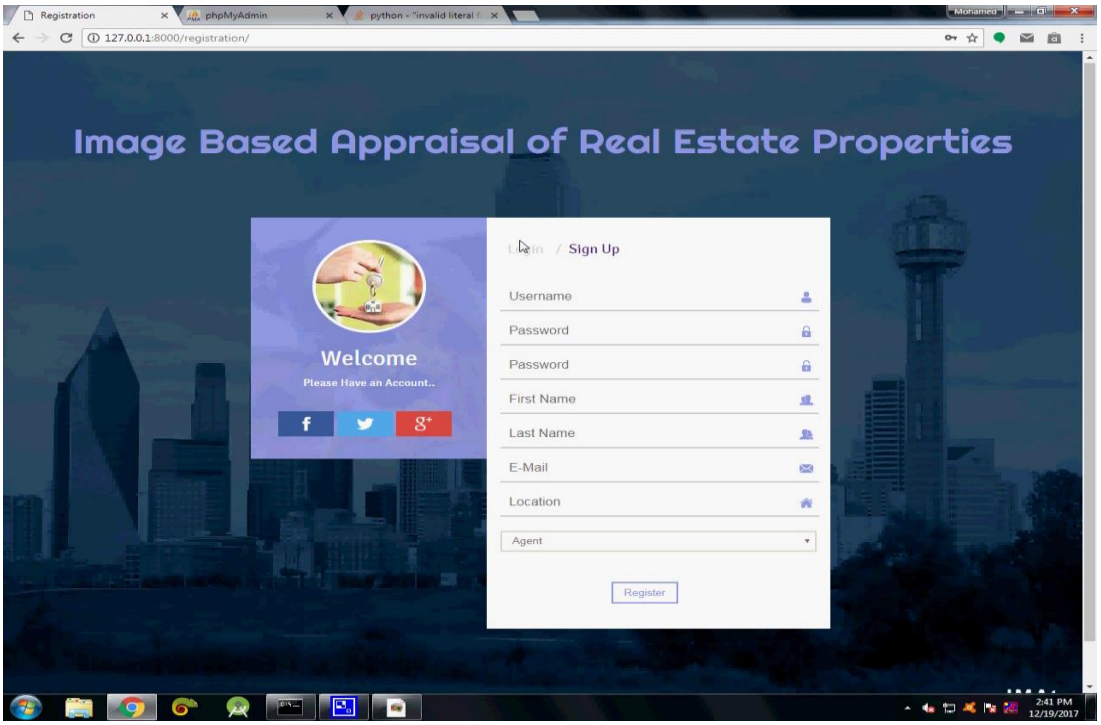
```
    data['lonn'] = lonn
    return render(request, 'seller/viewproprtyloc.html',data)
@login_required(login_url='/login/')
def chat(request):
    prof=Profile.objects.filter(user_type='AG')
    chatcheck=False
    data={}
    data['prof'] = prof
    data['chatcheck'] = chatcheck
    return render(request, 'seller/chats.html',data)
@login_required(login_url='/login/')
def chatbegin(request,uid):
    prof=Profile.objects.filter(user_type='AG')
    form = ChatForm(request.POST)
    msgs=Chat.objects.filter(receiver__in=[uid,request.user.id])
    chatcheck=True
    if form.is_valid():
        chatdet=form.save(commit=False)
        chatdet.user=request.user
        chatdet.receiver=uid
        chatdet.save()
        return redirect('sellers:chatbegin',uid)
    data={}
    data['form'] = form
    data['prof'] = prof
    data['msgs'] = msgs
    data['uid'] = uid
    data['chatcheck'] = chatcheck
    return render(request,'seller/chats.html',data)
```

8. SCREENSHOTS

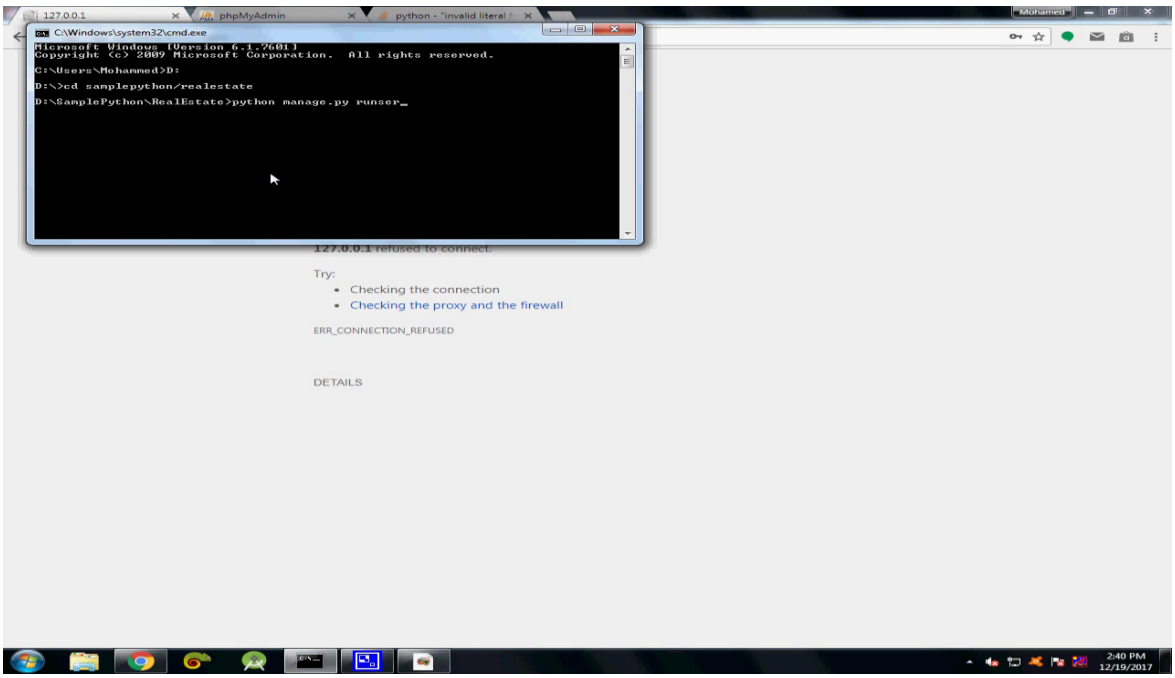
8. SCREENSHOTS



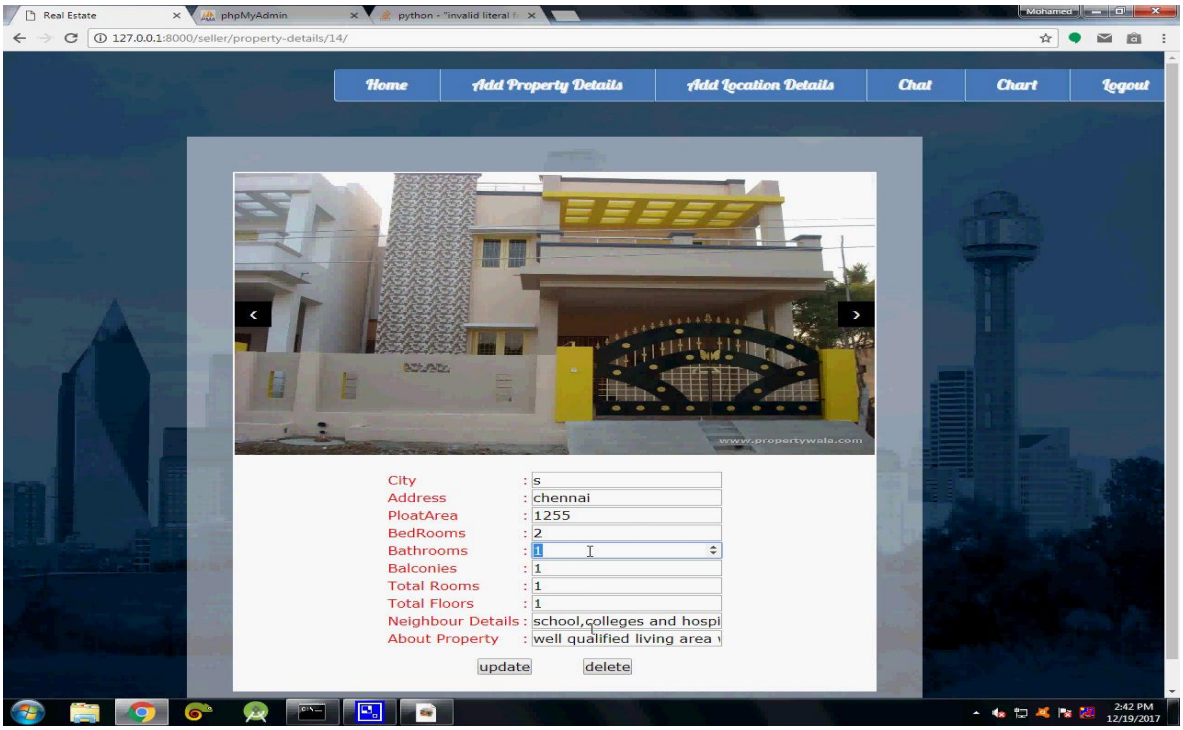
Screenshot 8.1: Login Page for Buyer, Seller and Agent Component.



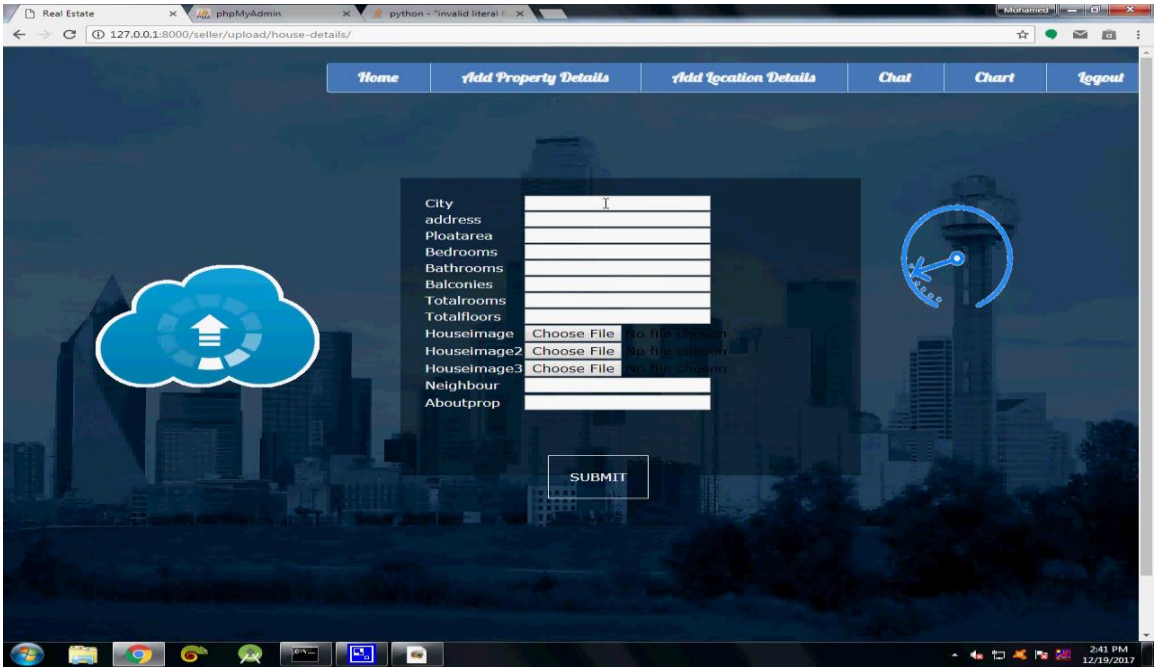
Screenshot 8.2: Signup Page for Buyer, Seller and Agent Component.



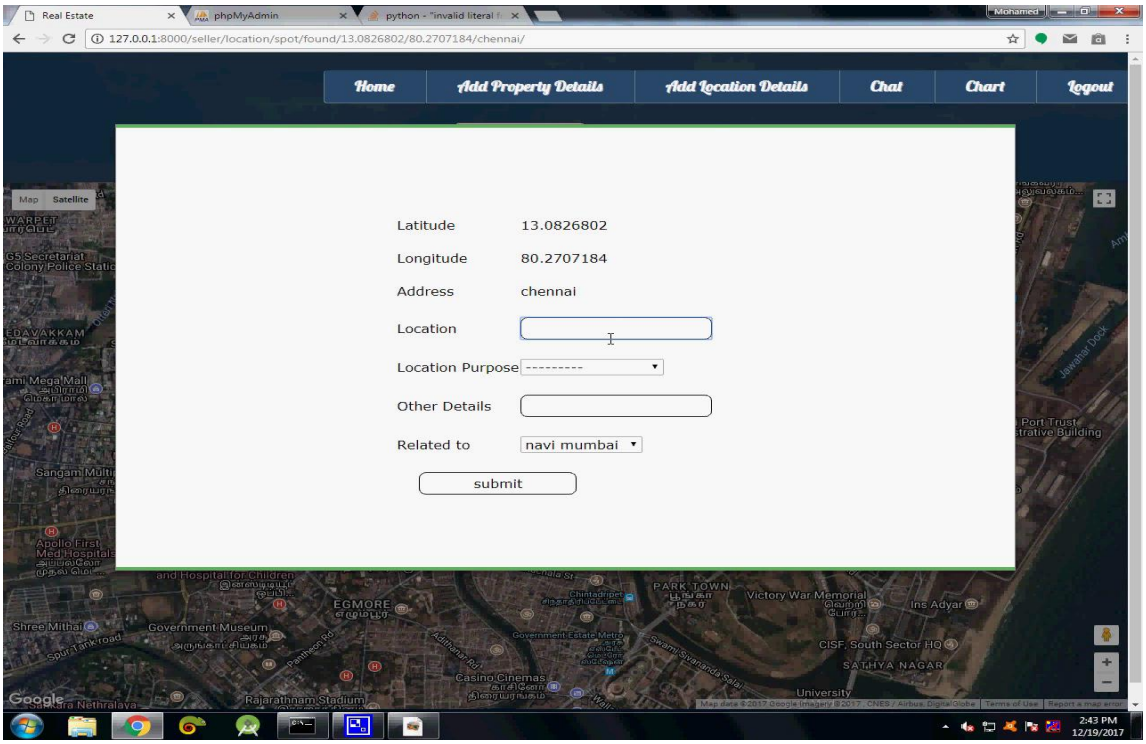
Screenshot 8.3: Execution of manage.py in command prompt.



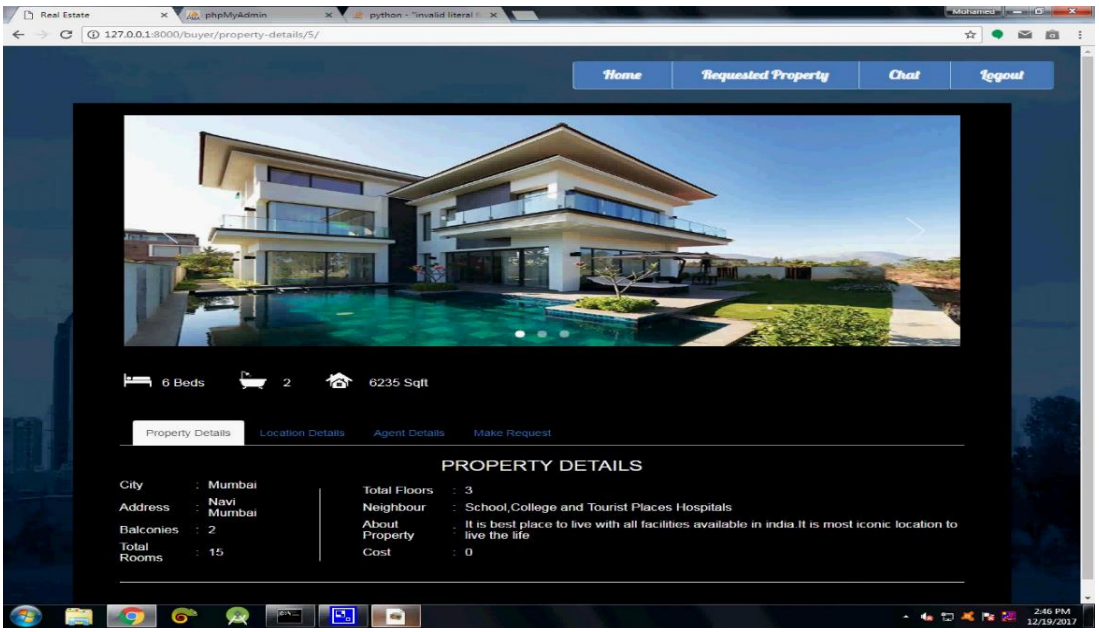
Screenshot 8.4: Adding of property Details.



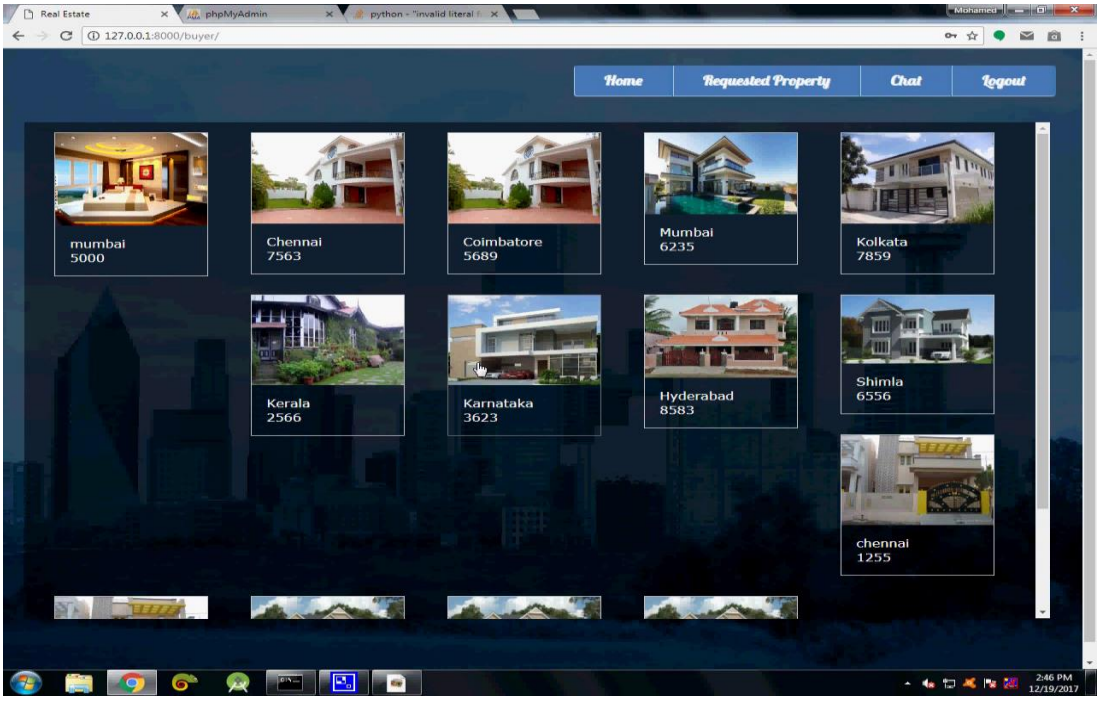
Screenshot 8.5: Adding Features of the property.



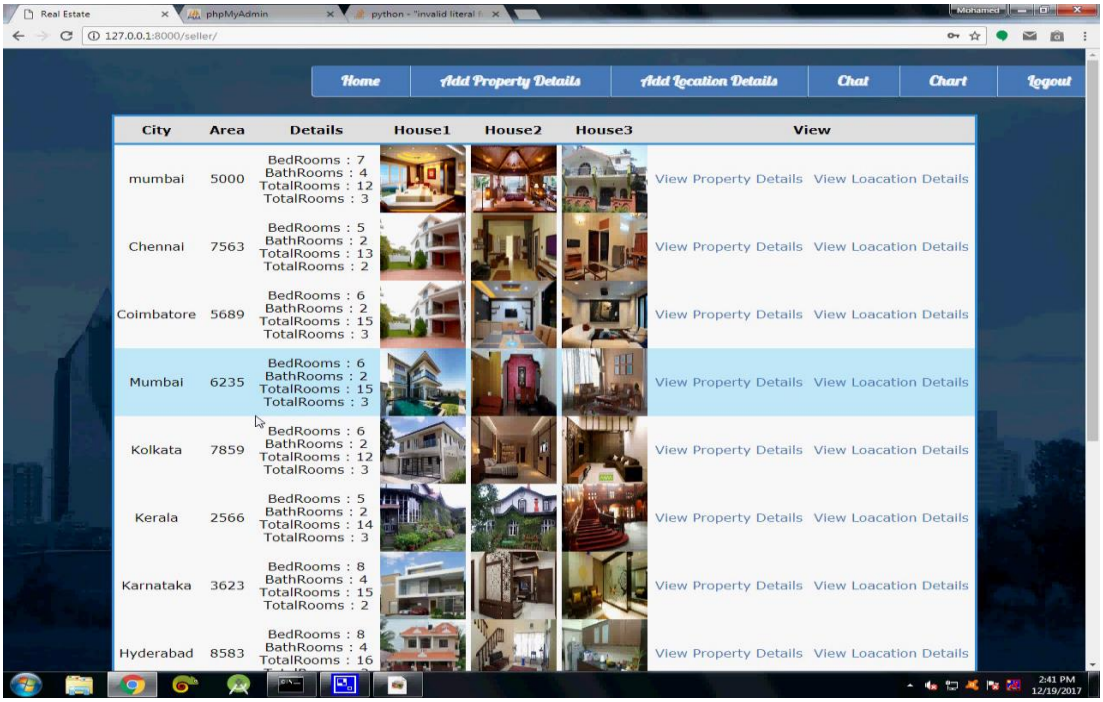
Screenshot 8.6: Adding location details for property.



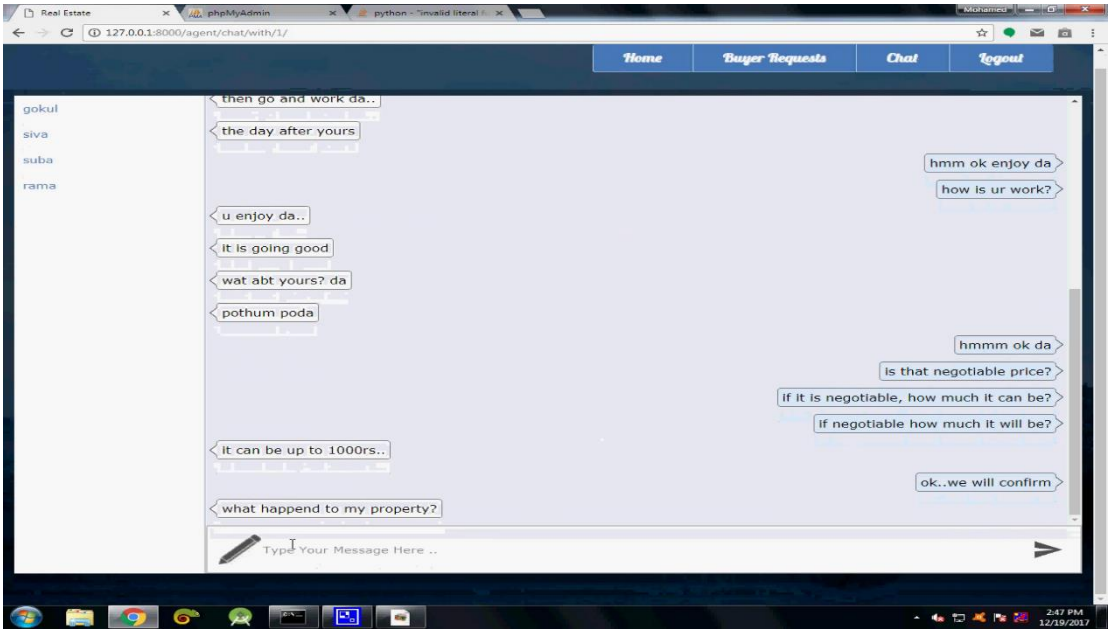
Screenshot 8.7: Property Viewed by the Buyer and Agent.



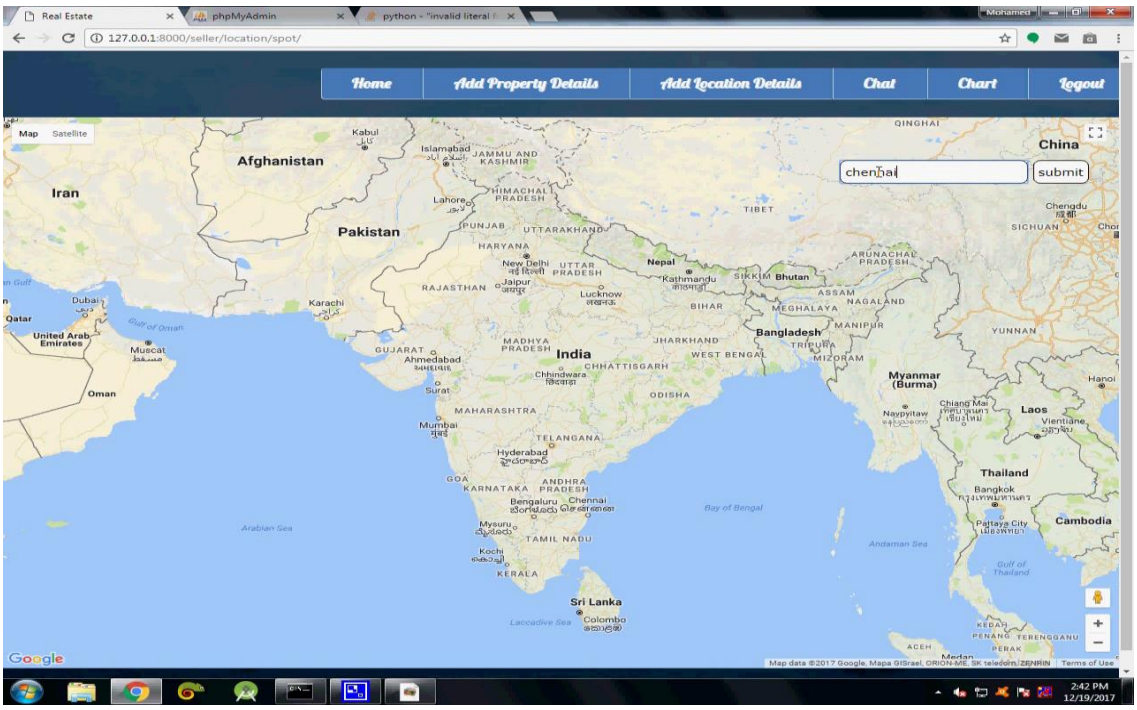
Screenshot 8.8: Property available at different loactions.



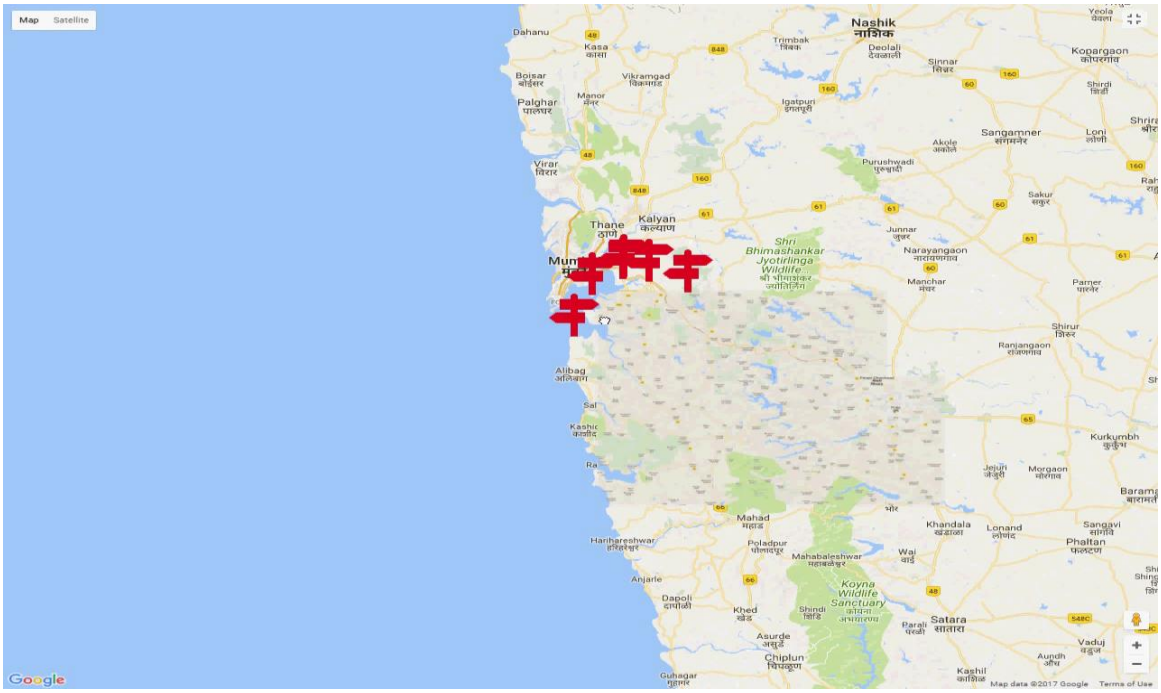
Screenshot 8.9: Property features at Different cities.



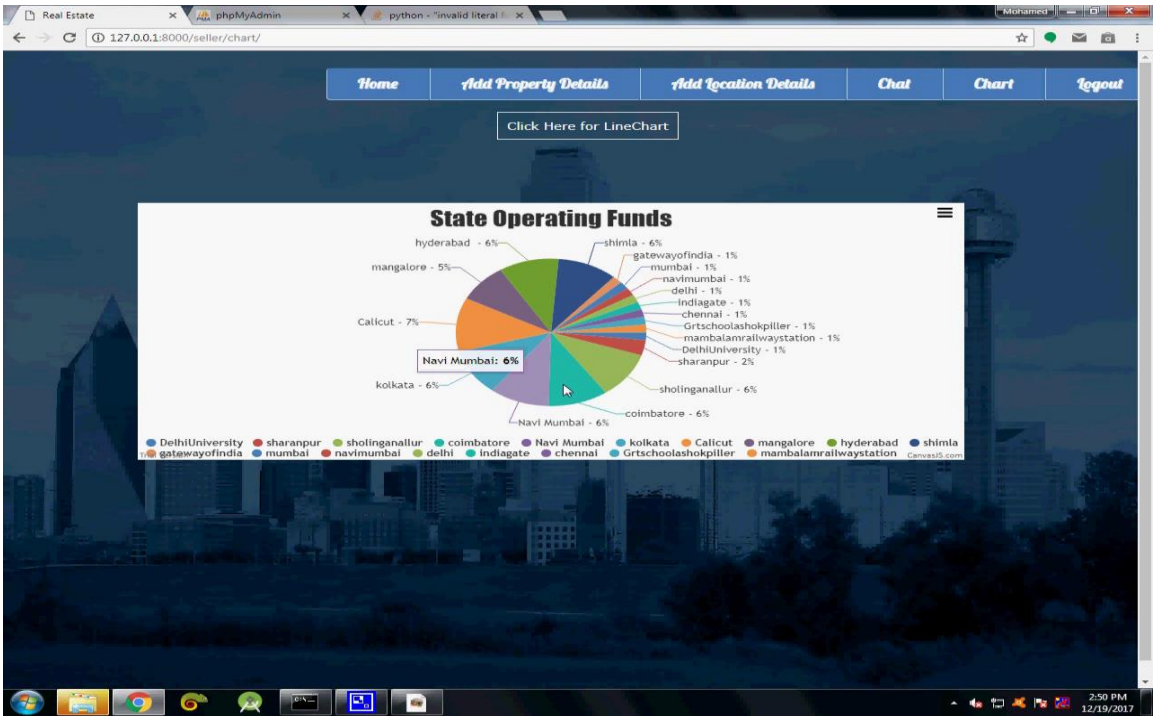
Screenshot 8.10: Chat between agent and buyer.



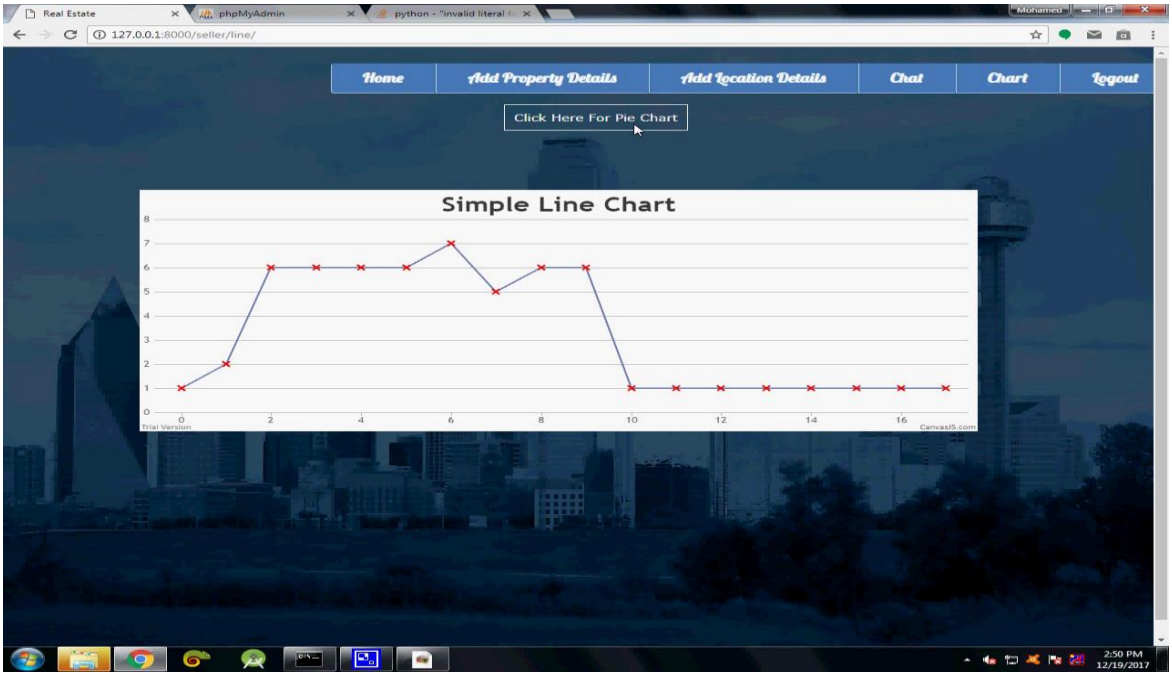
Screenshot 8.11: Satellite view of real estate in different cities.



Screenshot 8.12: Satellite of properties represented by tags.



Screenshot 8.13: Pie chart representing cities with most properties.



Screenshot 8.14: Line chart representing graph of sales.

9. TESTING

9. TESTING

9.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

- Valid Input : identified classes of valid input must be accepted.
- Invalid Input : identified classes of invalid input must be rejected.
- Functions : identified functions must be exercised.
- Output : identified classes of application outputs must be exercised.
- Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format.
- No duplicate entries should be allowed.
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company.

level – interact without error.

Test Results

All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

10. CONCLUSION

10. CONCLUSION

10 PROJECT CONCLUSION

In this work, we propose a novel framework for real estate appraisal. In particular, the proposed framework is able to take both the location and the visual attributes into consideration. The evaluation of the proposed model on two selected cities suggests the effectiveness and flexibility of the model. Indeed, our work has also offered new approaches of applying deep neural networks on graph structured data. We hope our model can not only give insights on real estate appraisal, but also can inspire others on employing deep neural networks on graph structured data.

11. BIBLIOGRAPHY

11. BIBLIOGRAPHY

REFERENCE

[1] Y. Fu, H. Xiong, Y. Ge, Z. Yao, Y. Zheng, and Z.-H. Zhou, "Exploiting geographic dependencies for real estate appraisal: a mutual perspective of ranking and clustering," in SIGKDD. ACM, 2014, pp. 1047–1056.

[2] K. Wardrip, "Public transits impact on housing costs: a review of the literature," 2011.

[3] Y. Fu, Y. Ge, Y. Zheng, Z. Yao, Y. Liu, H. Xiong, and N. Yuan, "Sparse real estate ranking with online user reviews and offline moving behaviors," p. 120129, 2014.

[4] A. Beja and M. B. Goldman, "On the dynamic behavior of prices in disequilibrium," The Journal of Finance, vol. 35, no. 2, pp. 235–248, 1980.

[5] E. L'Eplattenier, "How to run a comparative market analysis (cma) the right way," http://fitsmallbusiness.com/comparative-market-analysis/, 2016.

[6] M. J. Bailey, R. F. Muth, and H. O. Nourse, "A regression method for real estate price index construction," Journal of the American Statistical Association, vol. 58, no. 304, pp. 933–942, 1963.

WEBSITES

https://www.ijitee.org/wpcontent/uploads/papers/v8i12S/L110310812S19.pdf

https://ieeexplore.ieee.org/abstract/document/7960069

https://viso.ai/deep-learning/object-detection/

GITHUB LINK