# Table of Contents

# List of Tables

# List of Figures

# 1. Introduction

# 1. Introduction

## 1.1 Definition

This is very useful website for customer as well as broker who deals with real estate market. This website provides the facility to add new properties for the broker so that customer can see the properties and contact directly to the owner of the property. This website contains main three types of properties. Residential and Commercial. This website also provides the facilities like featured properties, premium properties etc. Featured properties are display on the home page so user can easily see these properties. Website also provides the great search facility to search properties as per customer requirements. In this website customer can directly connect with the broker who has post the property. Customer can also search properties on Google map so customer can easily identified the location of property. Admin panel is very powerful where admin can do everything like manage users, properties, etc.

## 1.2 Purpose

The main purpose of this website is to provide a platform for the people to search as well as list the information about any real estate property on the internet. This website will provide a platform to search all the properties in its database, with various authentication levels.

## 1.3 Objective

The main objective of the system is to make the work of finding appropriate properties and maintain the direct relationship between Seller of the property and Buyer of the Property and to make a system which reduce the work of industry.

## 1.4 Scope

This web-site will basically perform two tasks that is searching and listing. In searching basically three kinds of searching techniques are provided in which the first is the normal search and the second is the advance search and the third one is the searching on the map. But there is one unique feature in this web-site that it provides the, spatial search which provides the facility to search any property based on the current location of the user. It means that if a user wants to find the properties which are nearer to his/her current location then also they can find the appropriate properties

## 1.5 Frontend and Backend

Front-end development creates the visual presentation to the end user. For web development this includes the HTML and JavaScript that creates what the user sees and what can run in the user's web browser. All of the front end presentation is generated by the middle-tier and back-end components. Back-end development includes all of the processing required to create the front-end for the user. For web development this includes the middle tier development on the web or application server, and back-end components such as the database and Search engine. Back-end development handles security through user authentication - login, and authorization - customizations based upon what a user is authorized to see. Any action requested by a user goes through a security check, and information is typically looked up in the database, and then a web page is generated and sent to the user using a web server.

- **Front End:** PHP
- **Back End :** My SQL

## 1.6 Technology and Literature Review

- **Technological review**
  - Coding: PHP with Dream Viewer.
  - Database: MS SQL.
  - Client side: JavaScript, Mortgage , Loan Calculator and Google maps APIs
- **Literature review**
  - To study the PHP.
  - To study the various DBMS concepts which are to be implemented to create the database. Also study of MS SQL and working of Xamp Server Details.

- To thoroughly study various websites based on the same concept. To study the domain and creating an application overcoming the major flaws of the studied websites.
- These should not only be useful at present but should be capable of change pertaining to the future as well. By doing so one will be able to accomplish the goals set for the website.

# 2. Project Management

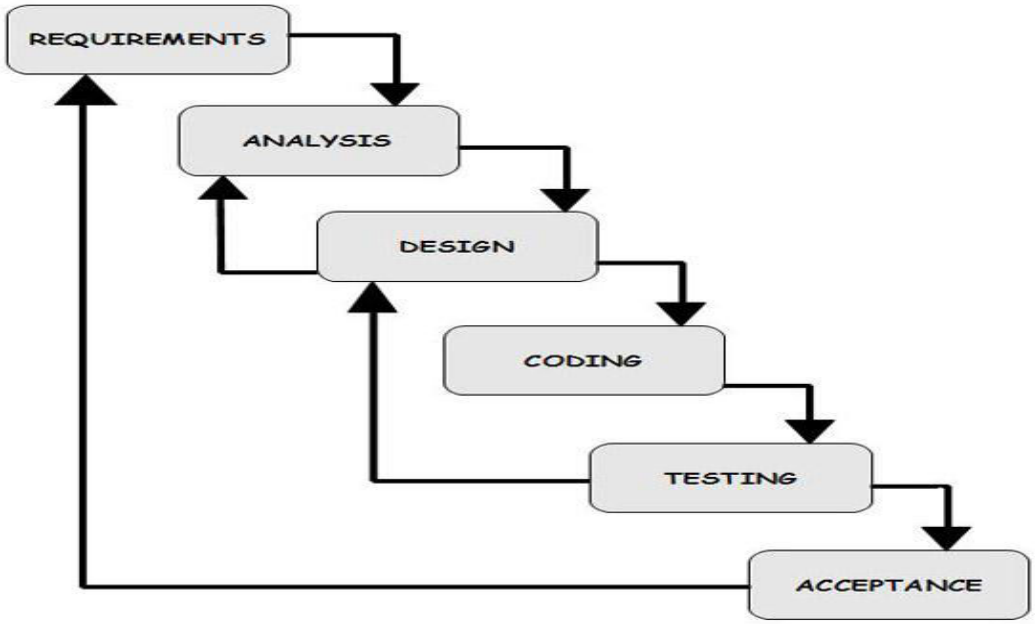
# 2. Project Management

## 2.1 Project Plan:

The planning of this project has followed the approach of Waterfall Development. The first phase being of Analysis and Requirement gathering deals with analysing the project and preparing a system requirement specification along with software project management and planning. Then comes the designing phase in which the various UML diagrams are made for our better understanding. The basic prototype of our database design is also prepared.

- **Development Strategy : Iterative Waterfall Model**
  Waterfall Model has been adopted as an approach for development of project. Most of the requirements of the project are fixed and already thought of, very less functionality updating is expected in future. So Waterfall model is the right approach for our project.

Fig 2.1 Waterfall model




**Justification:**

As our application is a Real Estate Property Listing and Searching website, the requirements are much clear as we have number of reference Real Estate Property Listing and Searching websites which can be used as reference so by looking that our requirements and functionalities that are to be implemented are clear.
As waterfall model is used for project whose requirements and functionalities are already known and as our project falls in this category we have chosen waterfall model as an approach to develop our project.

## 2.2 Product Perspective:

This web-site handles most of its works on its system only. Though there is some other Websites which provide the same service such as property listing and property searching but they do not provide the spatial search facility which is covered in this website. Also there is one other special feature that the current web sites which are already present in the market are not having which is open date feature of this web-site. These two special features are illustrated in the scope part of the document early

## 2.3 Product Functionality:

The major functions that this web-site will perform are:

(1) Post the property information in normal list on the website's database.
(2) Post the property information in featured list on the website's database by doing payment online.
(3) Search the property on the basic information provided by the user such as property price, state, city, property type.
(4) Search the property with some advance options such as no. of rooms, extra indoor facilities, etc.
(5) Searching the property on the map directly.
(6) Additional modules such as
    a. Vastu Tips.
    b. Area Conversion Calculators.
    c. EMI Calculators.
    d. Taxation Advice.
    e. Loan Eligibility.
(7) Opinion Poll: Public opinion for a particular property by conducting a series of questions, which would even be helpful to other clients referring to the website.
(8) Banking Options-This module will provide several banking recommendation for the property dealing.
(9) Feedback/ Customer Reviews: It mainly works as a tool for communication between the the two Clients and between Client and Administrator on the regular basis.

## 2.4 Milestones and Deliverables

- **Milestones**
    - Analysis & requirement gathering.
    - Learning curve.

- Designing – GUI.
- Designing – Database.
- Coding.
- Testing.

- **Deliverables**

- Software Requirement Specification.

  The SRS document states in precise in explicit language those functions and capabilities a software system must provide. Also states any required constraints by which the system must abide. It contains all the necessary functional and Non Functional requirements of the application. SRS accomplishes four major goals:

  i. It provides feedback to the customers
  ii. It decomposes the problems into different component parts
  iii. It serves as an input to the design specification.
  iv. It serves as a product validation check.
  v. The SRS document contains all diagrams (use cases, sequence, data flow and class), functional and Non Functional requirements for the website.

- Software Project Management Plan.

  Software project management is the art and science of planning and leading software projects. It is a sub-discipline of project management in which software projects are planned, monitored and controlled. Listing the details of the major functions, project resources, staff representations and scheduling.

- Analysis and design of the Website.

  Containing the design issues and the GUI diagrams. The design issues for any web application are:

  i. Page load efficiency.
  ii. Simplicity.
  iii. Use the space wisely.

- Project Report.

  A full project report, enlightening about the details of the project.

Working Website.

A fully functional website at the end of the year.

## 2.5 Roles and Responsibilities

| ROLES | RESPONSIBILITY |
|---|---|
| Analysis and Requirement Gathering | Jay Sheth<br>Smit Vasha |
| Design | Jay Sheth<br>Smit Vasha |
| Coding | Jay Sheth<br>Smit Vasha |
| Testing | Jay Sheth<br>Smit Vasha |

Table 2.3 Roles and Responsibility

## 2.6 Project Scheduling

**Start date: 01 / 10 / 2013**

**Project: REAL ESTATE PROPERTY LISTING AND SEARCHING**

| Current Week | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Weeks<br>**Tasks** | 01/10 to 01/17 | 01/17 to 01/24 | 01/24 to 01/31 | 01/31 to 02/07 | 02/07 to 02/14 | 02/14 to 02/21 | 02/21 to 02/28 | 02/28 to 03/07 | 03/07 to 03/14 | 03/14 to 03/21 | 03/21 to 03/28 | 03/28 to 04/04 | 04/04 to 04/11 | 04/11 to 04/18 |
| Themes and Desigining | | | | | | | | | | | | | | |
| Admin Login module and UI | | | | | | | | | | | | | | |
| User Module and user login | | | | | | | | | | | | | | |
| Registration and Validation | | | | | | | | | | | | | | |
| Search Module | | | | | | | | | | | | | | |
| API and Forget Password | | | | | | | | | | | | | | |
| Browse Listing and Pagination | | | | | | | | | | | | | | |
| Additional Features and Optimization | | | | | | | | | | | | | | |
| Testing | | | | | | | | | | | | | | |
| Documentation | | | | | | | | | | | | | | |
| **Tasks**<br>Weeks | 01/10 to 01/17 | 01/17 to 01/24 | 01/24 to 01/31 | 01/31 to 02/07 | 02/07 to 02/14 | 02/14 to 02/21 | 02/21 to 02/28 | 02/28 to 03/07 | 03/07 to 03/14 | 03/14 to 03/21 | 03/21 to 03/28 | 03/28 to 04/04 | 04/04 to 04/11 | 04/11 to 04/18 |
| Current Week | | | | | | | | | | | | | | |

Table 2.4 Time Line chart

# 3. System Requirement Study

# 3. System Requirements Study

## 3.1 User Characteristics

The requirements presented in this section describe the interfaces for users. The requirements do not assume a particular interface; however, the requirements are grouped according to the main features (as defined by the use cases) provided by the system.

1. Home Page
   The Homepage will display featured property listings, have quick search, and other various links such as Listing Alerts, Contact Us, Site Map that will provide the user with more relevant type of real estate information. There will be and min Control Panel link allowing the Administrator to login to their control panel and edit the database and user permissions.
2. Search Listings
   The Search Listings page will allow the user to perform a detailed search of any available properties within the web site. Some of the search features will include listing
   Type, bathrooms, bedrooms, price range, and location, among many others.
3. Password Security and Retrieval
   The Password Reminder page will be linked out of the login page. It will allow the user to enter the E-Mail address they registered to use with the Real Estate web site. The user will then receive an email containing their username and password they used during registration.
4. Contact Us
   The Contact Us page will provide the user the means to send an email to the websites administrator. The user will be required to enter First Name, Last Name, Address, Phone.
   Number, Email, Message, and a Security Code. In order to contact the Realtor, there is an Agents Profile section that the user can send a message to the selected realtor.
5. Add Listings
   The Add Listing page will allow the realtor to add images, listing type, status, MLS, Property Type, Style, Description, and various other features such as size, bathrooms, bedrooms, location, address, year built, appliances included, price, etc. There will also be a
   Calendar to set up showings or open houses. After the required fields and optional fields are entered, the realtor will hit the submit button to add that listing. Any required fields not filled will require information before proceeding.

6. Searching
7. In This module the client will be allowed to search through the Website for the property according to its need.
8. Administrator's Domain
   The Administrator's Domain is the module of the website where only the ones who have Administrator's privileges are allowed. In this module they are allowed to edit the database and the user privileges.

## 3.2 Hardware and Software Requirements.

### 3.2.1 Hardware Requirements

The system will interface with the following hardware:

1. Pentium p4 or higher processor.
2. 256 MB or more RAM.
3. Internet connectivity
4. For handling the website the server side computer uses the static IP machine.

### 3.2.2 Software Requirements

#### 3.2.1.1 Server Environment:

| | |
|---|---|
| Operating System: | Windows XP with Service Pack 3 or Windows Vista (Home Premium, Business, Ultimate, or Enterprise) or Windows 7 minimum<br>Mac 10.4 minimum<br>Ubuntu 10.6 minimum |
| Web Browser: | Internet Explorer 6 minimum, Firefox 3.0 minimum, Safari 3 minimum |

Table 3.2.1.1 Server Requirement

#### 3.2.1.2 Client Environment:

| | |
|---|---|
| Operating System: | Windows XP with Service Pack 2 or Windows Vista (Home Premium, Business, Ultimate, or Enterprise) or Windows 7<br>Mac 10.4 minimum<br>Ubuntu 10.6 minimum |
| Web Browser: | Internet Explorer 6 minimum, Firefox 3.0 minimum, Safari 3 minimum |

Table 3.2.1.2 Client Requirement

## 3.3 Communication Interface

The system will be developed as a client-server application with the server providing data access service only
This web-site uses the MVC architecture for the communication interface.
The MVC design pattern defines a way to organize our code according to its nature. This pattern separates the code into three layers:
• The Model layer defines the business logic (the database belongs to this layer).
• The View is what the user interacts with (a template engine is part of this layer)

## 3.4 Design and Implementations Constraints

The general constraints on the development of the system are as follows:

1. All data transmitted to the central database will be will be accessed only by the authorized users.
2. Member will not be able to access the administrator's data.
3. The system must not be hanged when many of the requests are sent at a time to the server
4. The system should be reliable in terms of validation and storing of the information to the database.
5. The hardware constraint for this website is minimum 128 MB RAM memory which should be available in the end-user's computer.
6. PHP Sessions will be used to secure the data.

## 3.5 Overall Constraints:

3.5.1 Regulatory Policies:

All the access rights to access the project is provided to all the users depending on their user type that is given by the administrator

3.5.2 Hardware Limitations:

Limitation of server due to high traffic.

### 3.5.3 Parallel Operations:

The site can be used by administrator as well as user at the same time. Administrator provides different category for buy product. And user can buy product which are available on the site.

### 3.5.4 Higher Order Language Requirements:

We have made project in PHP but it has a problem of post back while the registration of the form. If we use the JavaScript then post back won't occur and speed will also increase.

### 3.5.5 Reliability Requirements:

The main reliability requirement is the validation used. Without proper validation the system does not allow to enter that value into the database. For eg. In the email id, the user cannot Enter any dummy value, the validation checks that whether there is '@' or '.' symbol in that. Also any null value is not allowed in place of compulsory fields.

### 3.5.6 Safety and Security Requirements:

If any user wants to buy product then he must be a register member otherwise he can't buy the product. We have stored the encrypted password in the database so none can see although if he got the database. We have done all payment checking so none dummy cardholder can buy any product and if no balance or credit card expire then also he can not buy the product.

# 4. System Analysis

# 4. System Analysis

## 4.1 Study of Current System:

Our system is a new system and so the study of the current system or the old system is not applicable.

## 4.2 Problem and weakness of current system:

- The purpose of creating this Real Estate Web Application is to outcast the discrepancies in hundreds of such existing systems on the World Wide Web.

- One of the basic problems with the existing systems is the non-interactive environment they provide to the users.

- Most of the applications involved in Real Estate business use some web template to put the content specific to their company and make it communicate with the database to search the listings.

- These templates simply use basic web controls to do this task making the web page non-interactive.

- On the other hand, the motive of this Real Estate Web Application is to allow the user to play with the search tool and create different combinatorial search criterion to perform exhaustive search.

- Another problem in such applications designed so far is the use of traditional user interfaces which make continuous post backs to the server; each post back makes a call to the server, gets the response and then refreshes the entire web form to display the result.

- This scenario adds an extra trade off causing a delay in displaying the results.

## 4.3 Requirements of new system:

As described above to overcome all the problem faced in existing system this new system will be develop.

### 4.3.1 User Requirements:

1) Any user can surf on the site. Means any user can search on the site. But when he click any item to add to cart then message will displayed that you must logged in.

2) When user login in the site then he can buy product add that in to the shopping cart and if he forgot its password then we mail him his password so he can login easily.

### 4.3.2 System Requirements:

System Requirement Specification (SRS)
End User Side:
R1.1 Login

R1.1.1 For Registered user:
I/P: Enter Id and Password.
O/P: Successfully login or retype password.

R1.1.2 Registration for new User:
I/P: Fill the Registration form.
O/P: You will get your id.

Admin Side**:**
R1**:** Login

In this if you have admin id and password then only you can login and make any change in website. Like in user if you don't have id you can register and get your id but hare you can't do anything if you don't have admin id and password.
I/P**:** Enter your id and password
O/P**:** You will redirect to admin page.

## 4.4 Feasibility Study:

Feasibility study is the likelihood the system will be useful to organization. After studying the requirements, whether the proposed project is feasible or not, is determined by checking the various feasibilities.

The three aspects in the feasibility study portion of preliminary investigation are:

## 1) Technical Feasibility:

- Technical feasibility corresponds to determination of whether it is technically feasible to develop the software.
- Necessary technology exists to do what is suggested and required by the organization.
- The proposed Equipment's have the technical capacity to hold the data required to use the new system.
- The proposed system will provide adequate response to inquiries regardless of the location if users.
- The hardware needed to develop and implement the system is adequate.
- The software guarantees accuracy, reliability and ease of access and data security.

## 2) Economic Feasibility:

A system that can be developed and that will be used if installed must still be a good investment for the organization. Financial benefits must equal or exceed the costs.

The financial and Economic issues raised are as under:

- No extra cost is incurred for developing the system. As required software are already used by the department.

- No extra cost for the modification or addition of software and hardware will require in case of future expansion of the current system.

- As the project is to be developed by trainees the cost incurred by the company is in the form of resource allocation rather than monetary. The cost on the company is indirect in the form of resources utilization.

- The company will be at profit if they implement this system because of the cost of implementation is nominal as compared to the profit they will be earning in terms of efficiency.

## 3) Operational Feasibility:

Operational feasibility focuses on whether the system will work when it is developed and installed. Operationally the system is feasible because:

- There is sufficient support for the project from management and user. The system is well liked and used to the extent that persons will not be able to see reasons for change.

- The current business methods are not acceptable because the manual system is time consuming. The users though initially repressive worked along with the development team once the initial doubts were cleared.

- The users have been involved in the planning and development of the project. This reduces the chances of resistance to the system.

- The proposed system will not cost any harm to the existing system and its users.

- No special training required for the user as it has a Self-explanatory interface. Validation of data input is taken care of by the system and not by the user.

- Since the most trivial of issues assumes a major problematic state later in the development cycle, every possible aspect of operational feasibility was checked. The proposed project passed all the feasibility tests and hence was declared feasible to organization and its functioning.

4.4.1 Does the system contribute to the overall objectives of the organization?

Yes

4.4.2 Can the system be implemented using the current technology and within the given cost and schedule constraints?

Yes

4.4.3 Can the system be integrated with other system which are already in place?

No

## 4.5 Requirements Validation:

- User id and password validation.
- All fields should be non-empty and satisfy all constraints.

## 4.6 Activity/Process in the System:

- Login page
  - Validation for login
- Properties record result page
- Detail information about individual properties record page.
- Property selection.
- Logout

## 4.7 Features of New System:

Features of new system are:

- It provide facility of Google mapping and 3d view.
- We add Color Box for slideshow of images of property.

# 5. System Design

# 5. System Design

## 5.1 Flow of Control

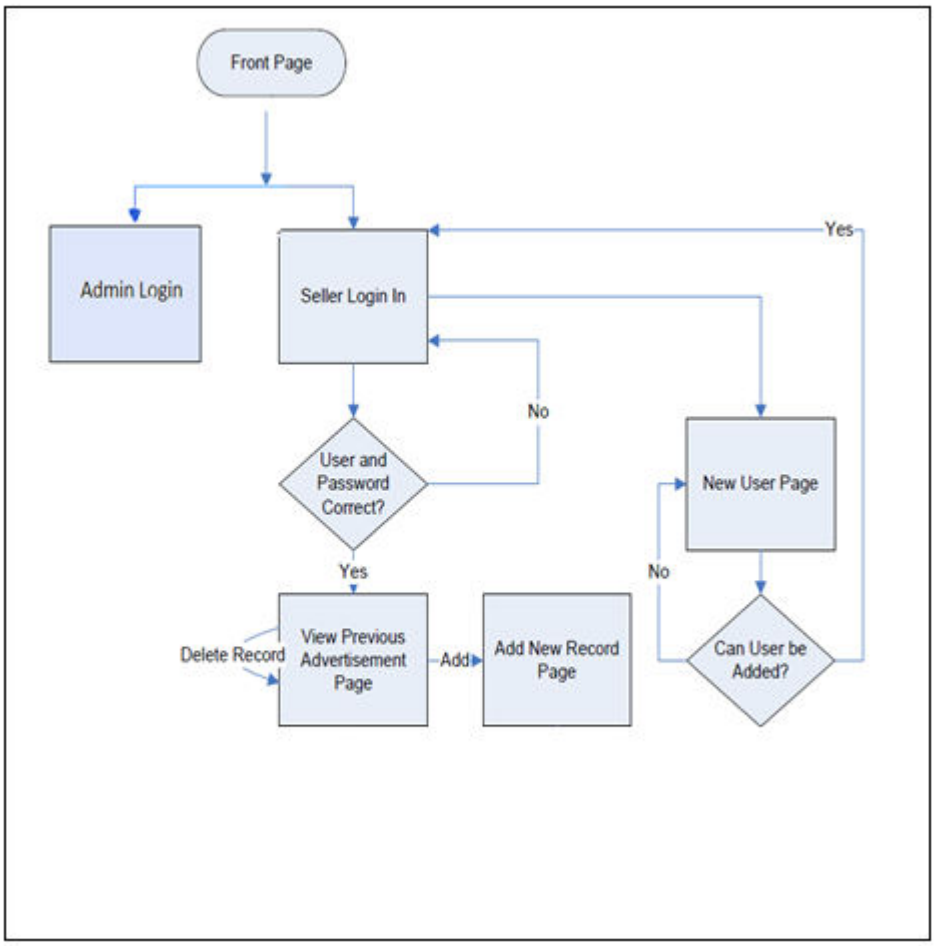



Fig 5.1 Flow of the System (Primary)

## 5.2 Sequence Diagram (General )

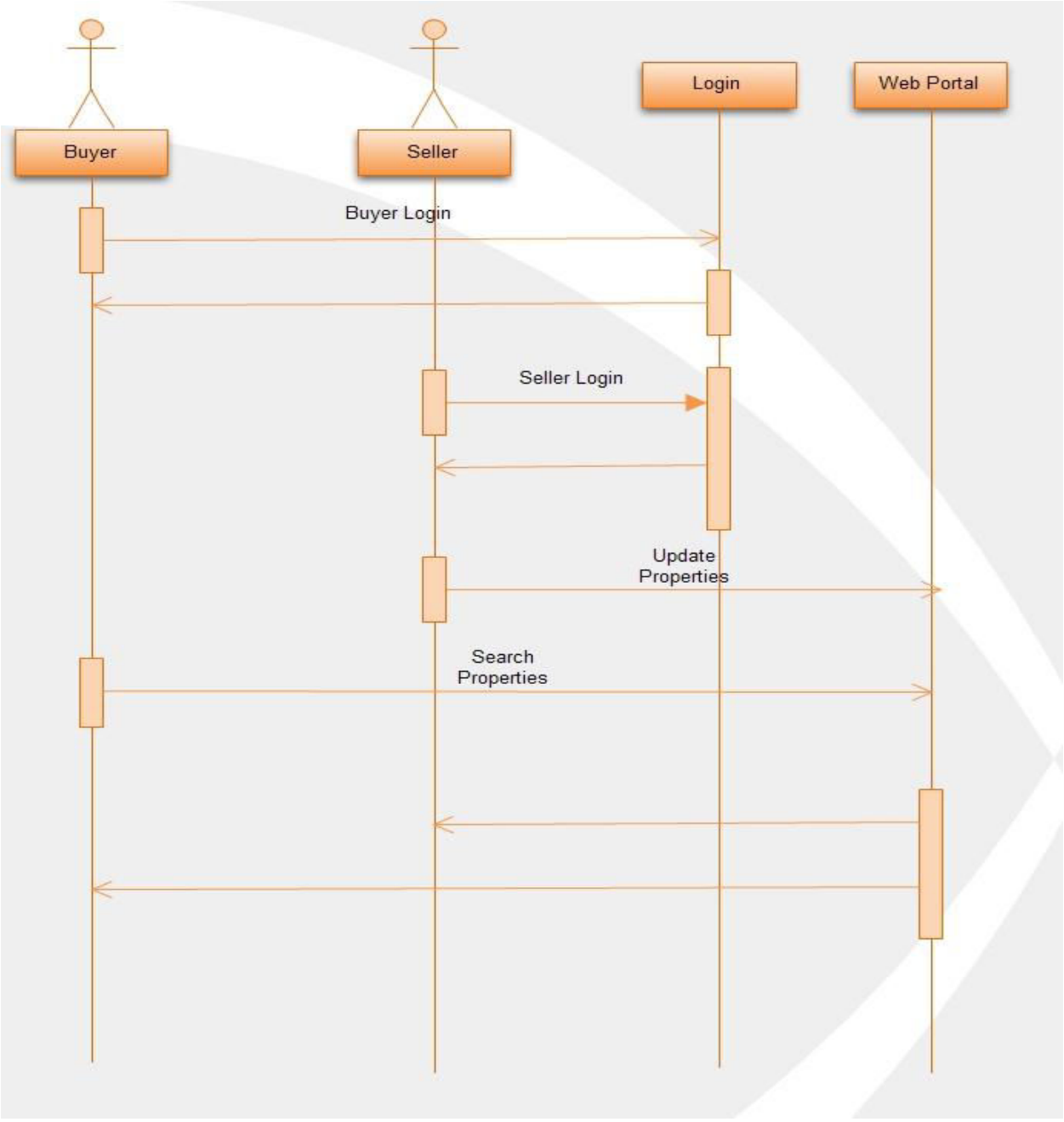



Fig 5.2 Sequence Diagram for the whole system

### 5.3 Context Diagram

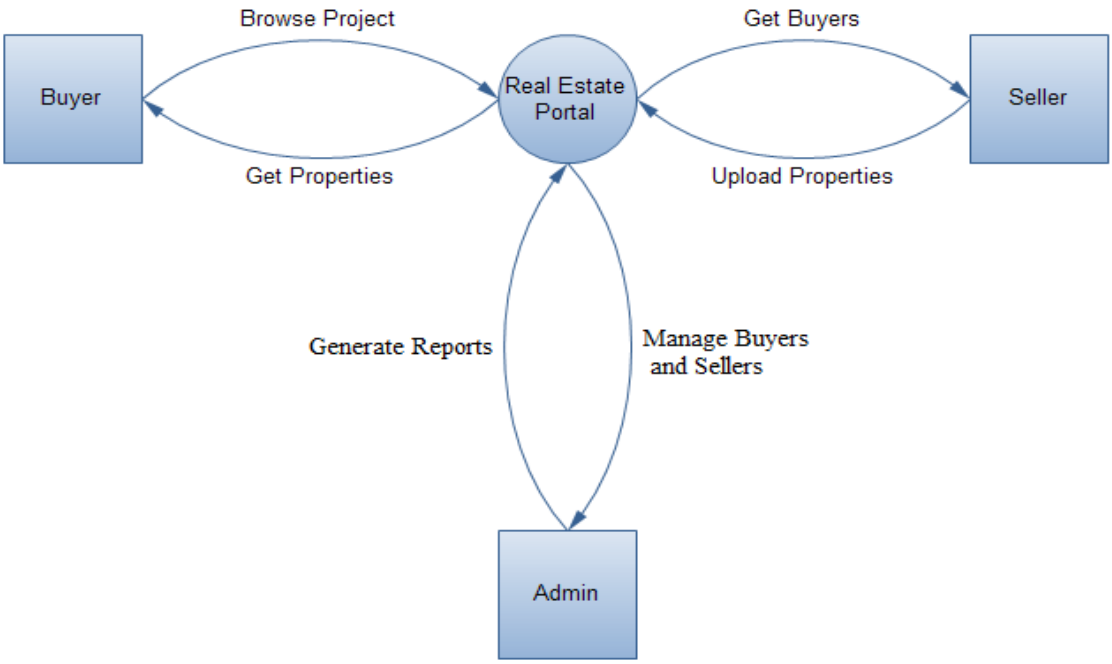



Fig 5.3 Data Flow Diagram

### 5.4 Data flow Diagram (zero level)

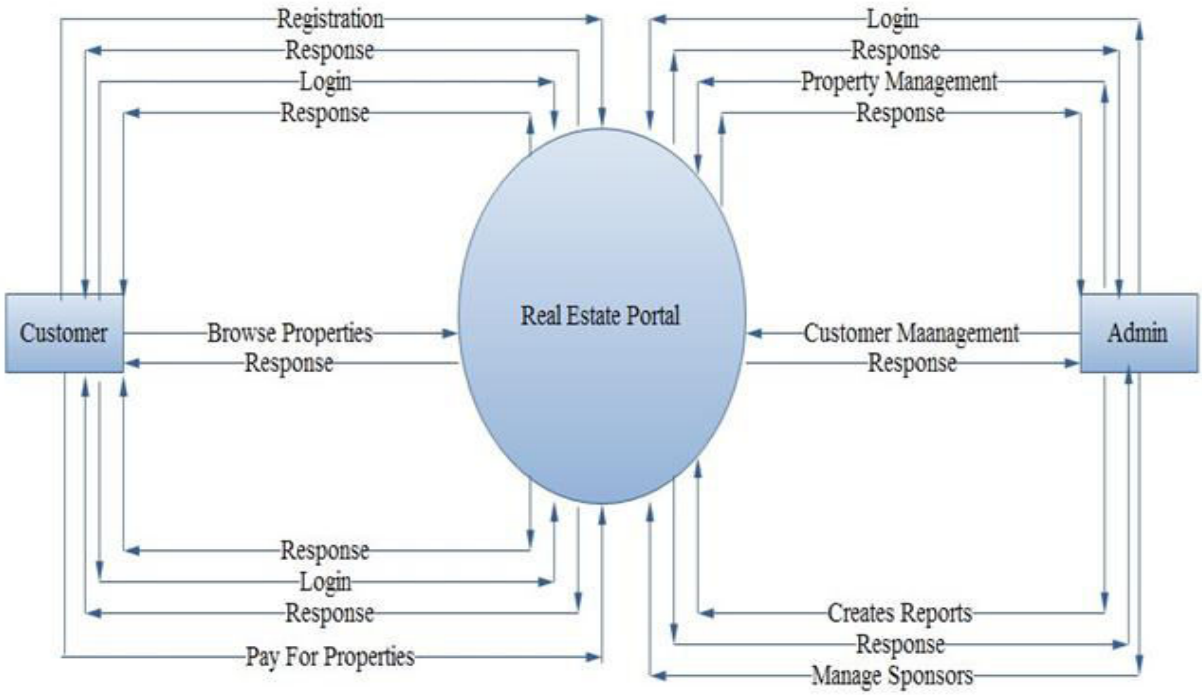



Fig 5.4 Data Flow Diagram

### 5.4.1 Data flow diagram(third level)

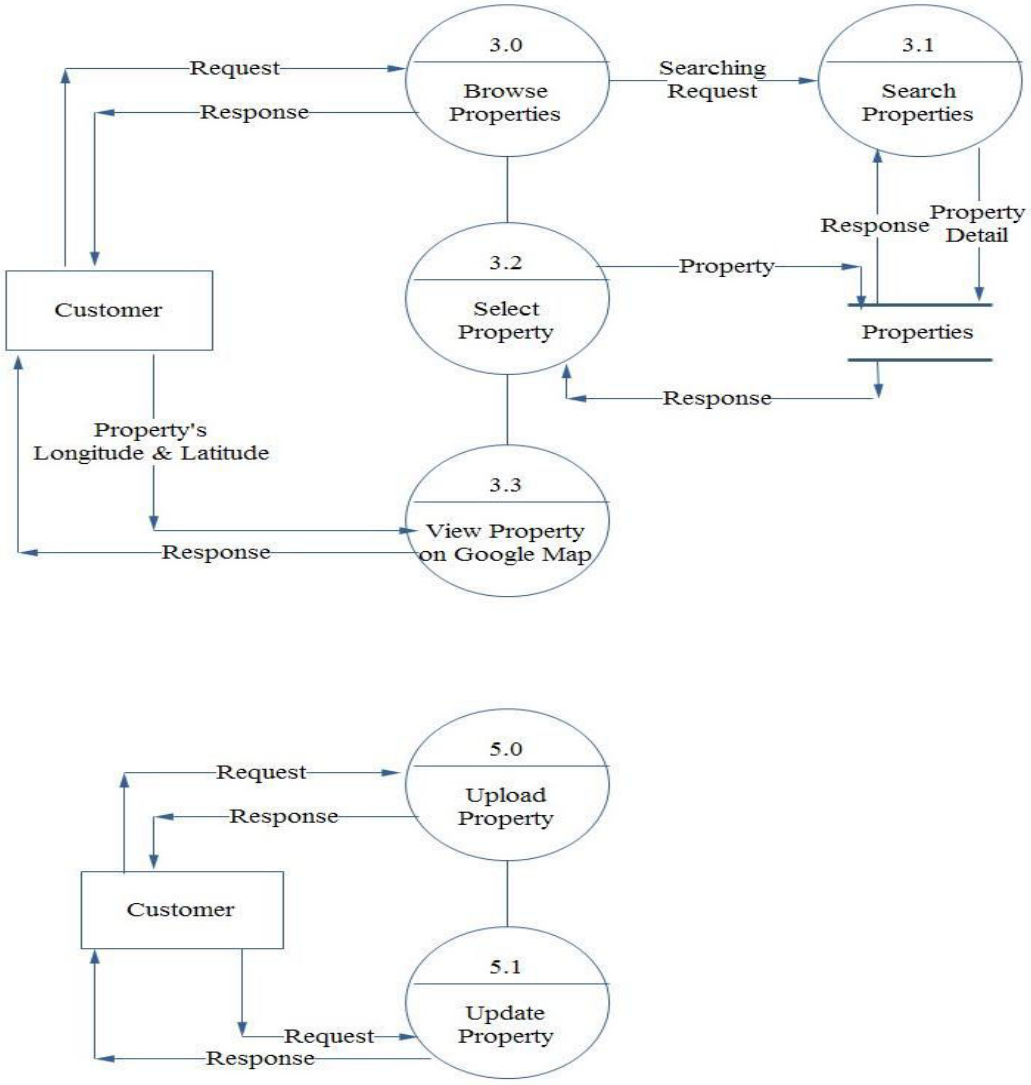



Fig 5.4.1 Data Flow Diagram

### 5.4.2 Data flow Diagram for Admin

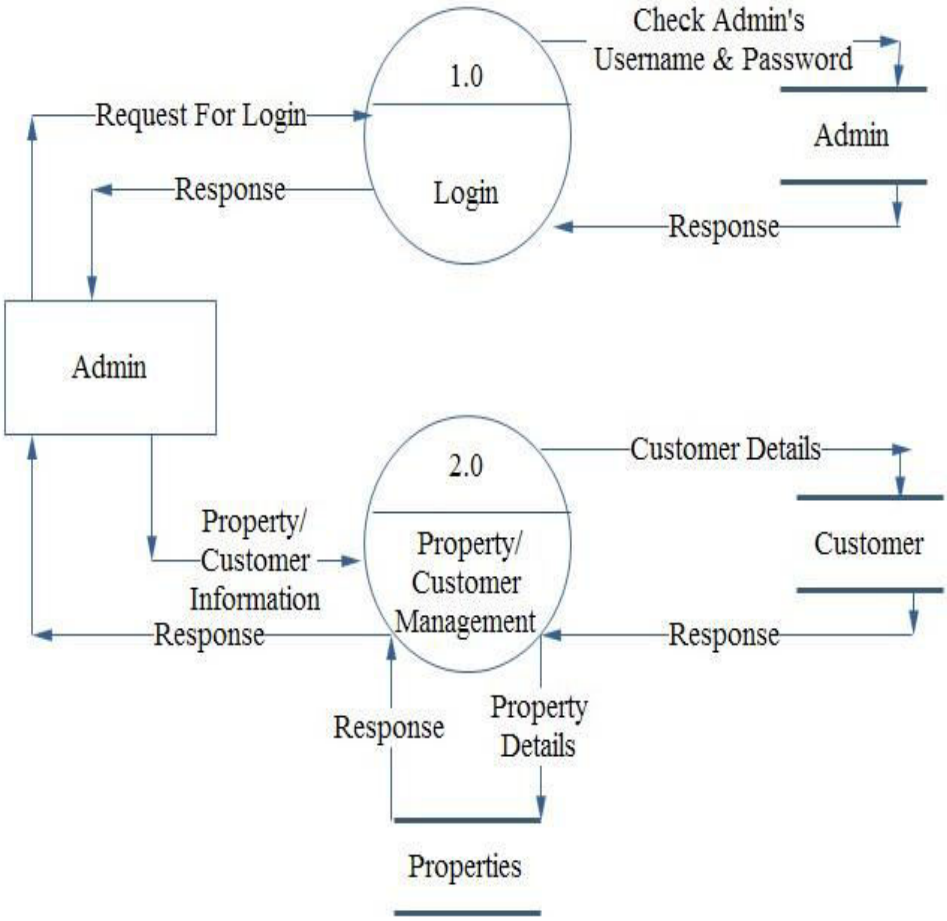



Fig 5.4.2 Data flow diagram for Admin

## 5.5 Use Case Diagram

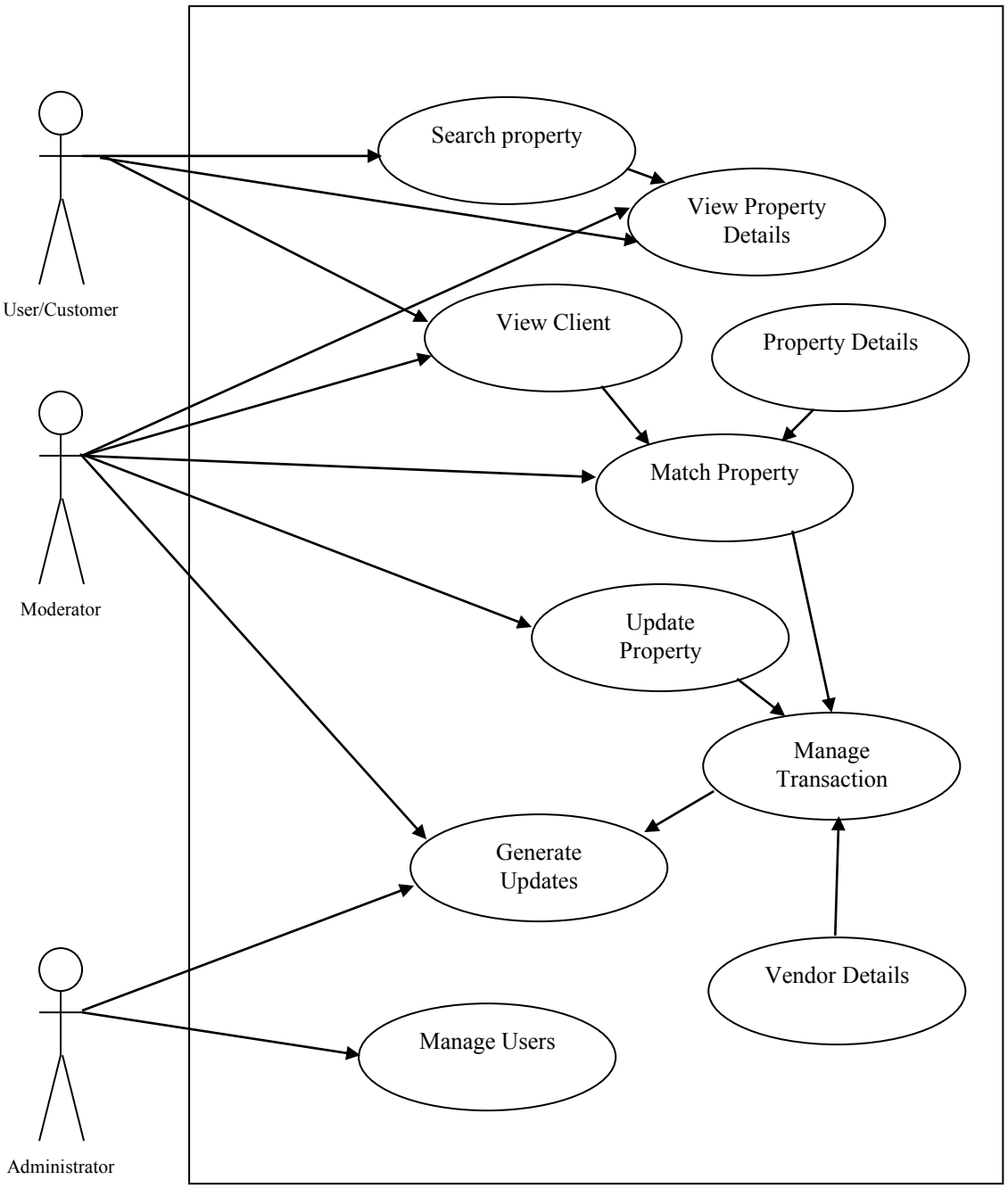



Fig 5.5 Use Case Diagram

## 5.6 ER Diagram:

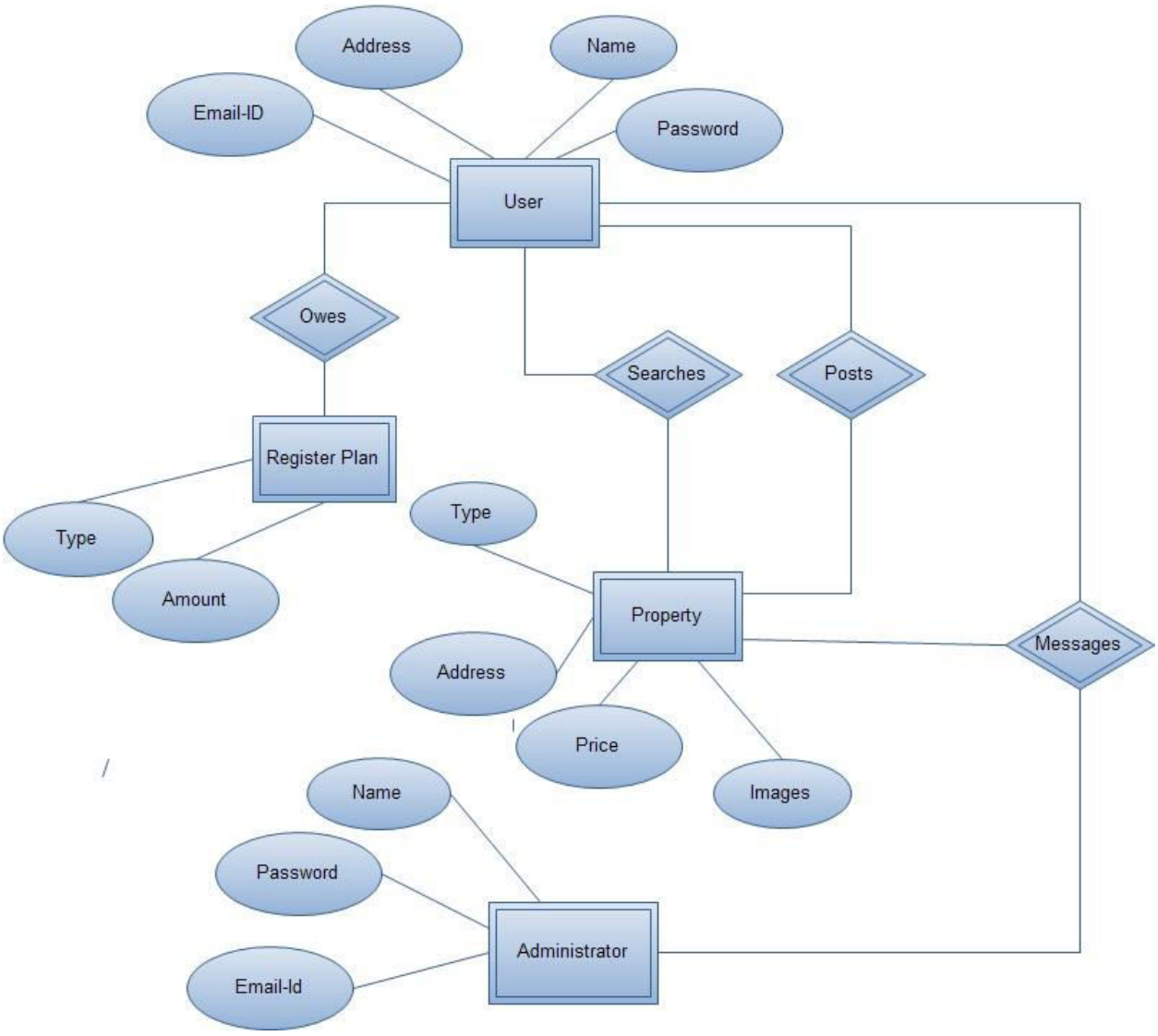



Fig 5.6 ER Diagram

## 5.7 Activity Diagram

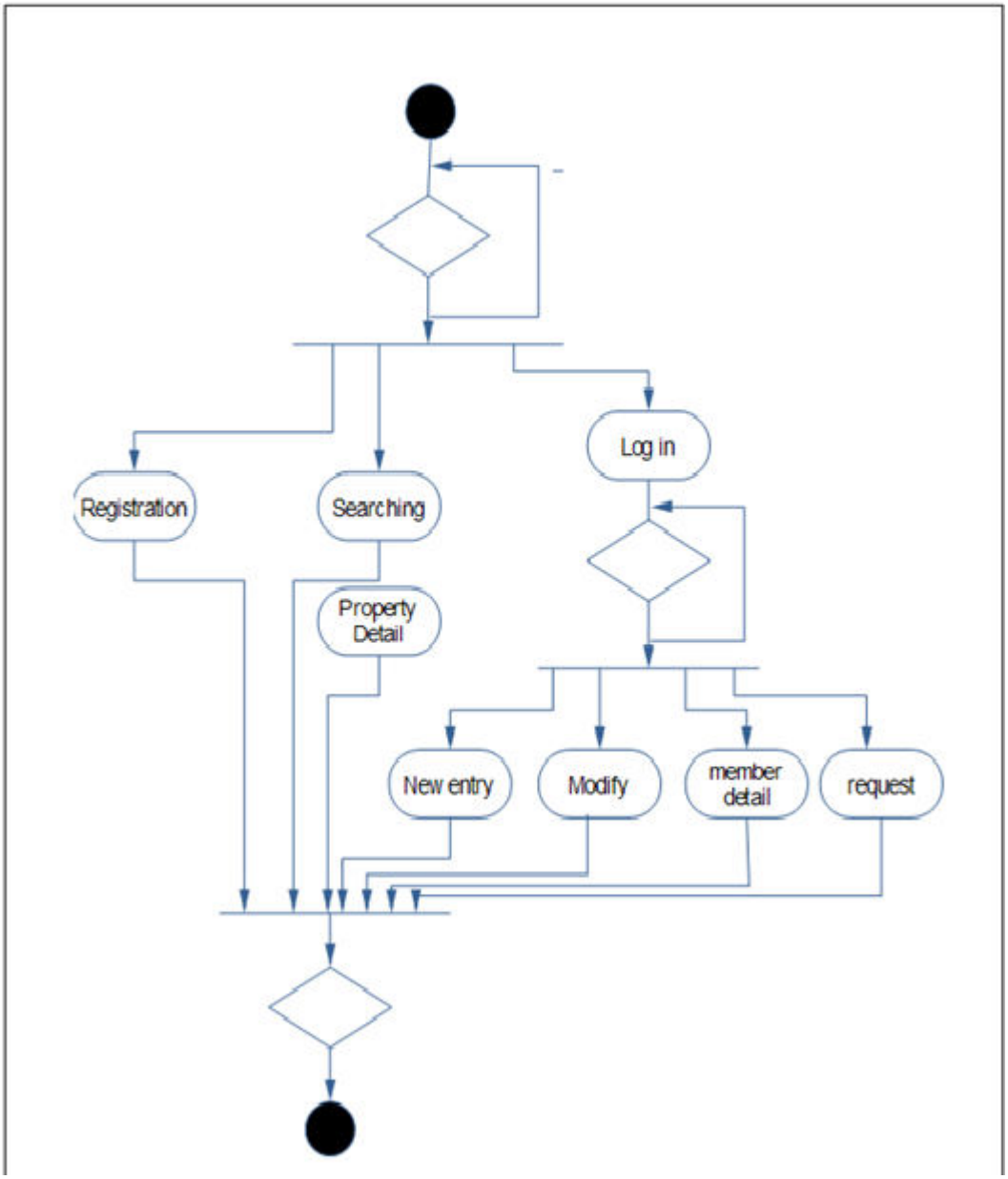



Fig 5.7 Activity Diagram

## 5.8 State Chart Diagram

### 5.8.1 State Chart for registration

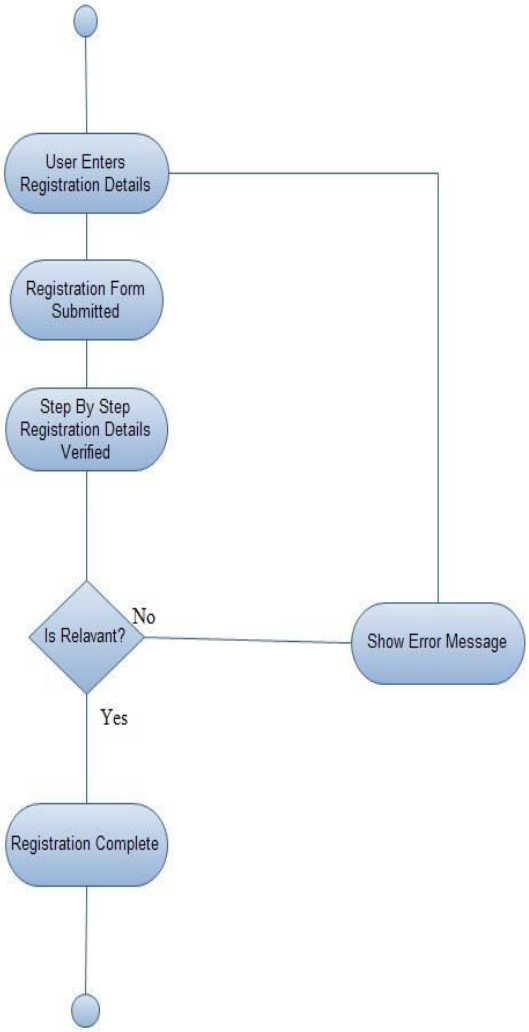


Fig 5.8.1 State Diagram for Registration

**5.8.2 State Chart Diagram of Login:**

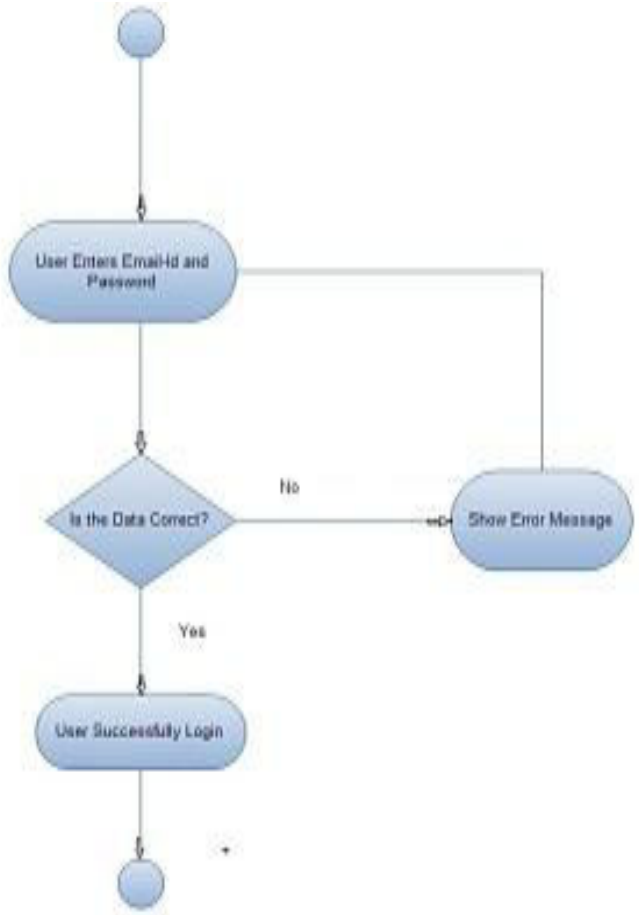



Fig 5.8.2 State Chart Diagram for Login

# 6. System Architecture

# 6. System Architecture

## 6.1 Types of Architecture

The software architecture of a program or computing system is a depiction of the system that aids in the understanding of how the system will behave.

Software architecture serves as the blueprint for both the system and the project developing it, defining the work assignments that must be carried out by design and implementation teams. The architecture is the primary carrier of system qualities such as performance, modifiability, and security, none of which can be achieved without a unifying architectural vision. Architecture is an artefact for early analysis to make sure that a design approach will yield an acceptable system. By building effective architecture, you can identify design risks and mitigate them early in the development process.

In our project we are using 2-tier architecture. The application logic is either buried inside the User Interface on the client or within the database on the server (or both). With two tier client/server architectures, the user system interface is usually located in the user's desktop environment and the database management services are usually in a server that is a more powerful machine that services many clients. Processing management is split between the user system interface environment and the database management server environment. The database management server provides stored procedures and triggers

## 6.2 Description of Roles of Each Tier

The two-tier is based on Client Server architecture. The two-tier architecture is like client server application. The direct communication takes place between client and server. There is no intermediate between client and server. Because of tight coupling a 2 tiered application will run faster.

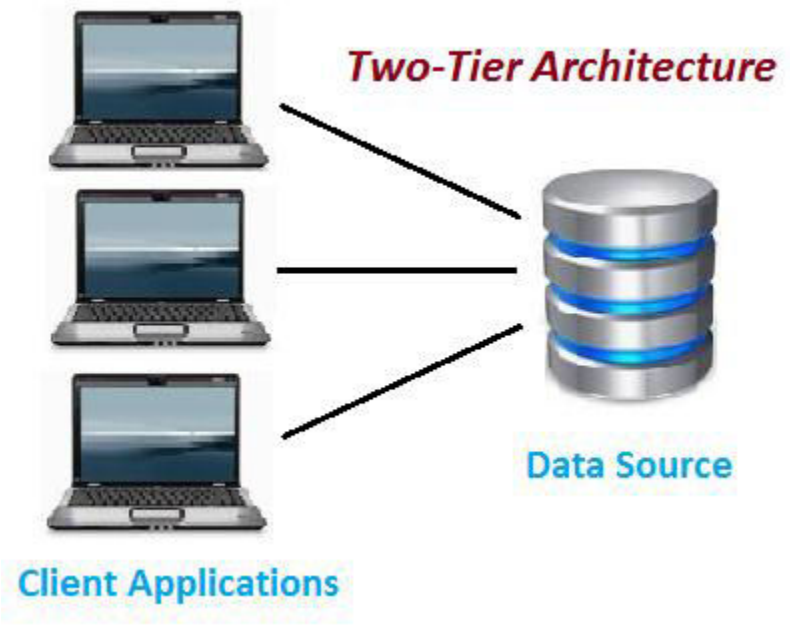



Fig 6.2 Tier Architecture

The above figure shows the architecture of two-tier. Here the direct communication between client and server, there is no intermediate between client and server.

Let's take a look of real life example of Railway Reservation two-tier architecture:

Let's consider that first Person is making Railway Reservation for Mumbai to Delhi by Mumbai Express at Counter No. 1 and at same time second Person is also try to make Railway reservation of Mumbai to Delhi from Counter No. 2

If staff from Counter No. 1 is searching for availability into system & at the same staff from Counter No. 2 is also looking for availability of ticket for same day then in this case there is might be good change of confusion and chaos occurs. There might be chance of lock the Railway reservation that reserves the first.

But reservations can be making anywhere from the India, then how it is handled?

So here if there is difference of micro seconds for making reservation by staff from Counter No. 1 & 2 then second request is added into queue. So in this case the Staff is entering data to Client Application and reservation request is sent to the database. The database sends back the information/data to the client.

In this application the user is an end user who is using Railway reservation application software. He gives inputs to the application software and it sends requests to Server. So here both Database and Server are incorporated with each other, so this technology is called as “Client-Server Technology“.

The Two-tier architecture is divided into two parts:

1) Client Application (Client Tier)

2) Database (Data Tier)

On client application side the code is written for saving the data in the SQL server database. Client sends the request to server and it process the request & send back with data. The main problem of two tier architecture is the server cannot respond multiple request same time, as a result it cause a data integrity issue.

**Advantages:**

- Easy to maintain and modification is bit easy
- Communication is faster

## 6.3.Blockdia

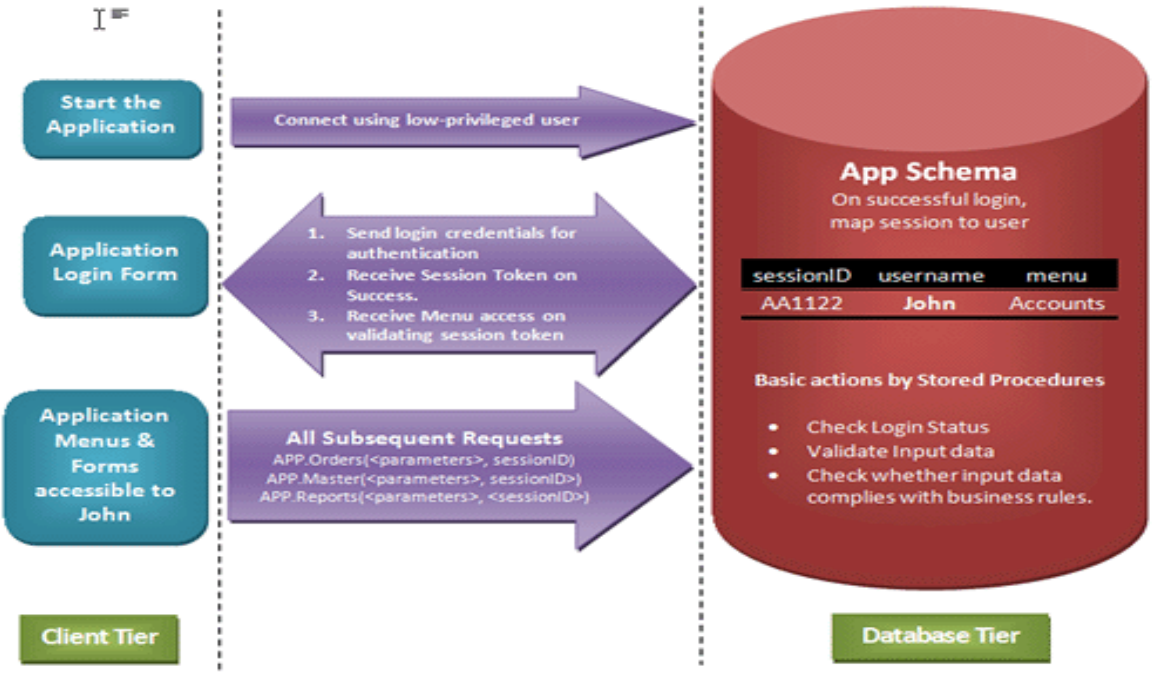



Fig 6.3 Tier Block Diagram

# 7. Implementation

# 7. Implementation

## 7.1 Security Features

- Only authorized users are allow to access the system.
- System shall provide a way to assign various rights to the various users. Like users of only COMPANY are allowed to login the system and view details, update details, and apply for information.
- Visitors are allowed entering into the system just to view the properties..
- Physical security is always a main concern to any system, but in our system it is not issue because we have purchased amazons server.
- All the password are stored in encryption form. So it is not easy to hack the user's password.

## 7.2 Coding Standards

The C# Language Specification does not define a coding standard. However, we have followed the standard as per the Microsoft's documentation.

- **Layout Convention**
    - Write only one statement per line.
    - Write only one declaration per line.
    - If continuation lines are not indented automatically, indent them one tab stop (four spaces).
    - Add at least one blank line between method definitions and property definitions.
    - Use parentheses to make clauses in an expression apparent, as shown in the following code.
- **Commenting Convention**
    - Place the comment on a separate line, not at the end of a line of code.
    - Begin comment text with an uppercase letter.
    - End comment text with a period.
    - Insert one space between the comment delimiter (///) and the comment text, as shown in the following example.

- **Data type convention**
    - Use the + operator to concatenate short strings, as shown in the following code.
    - In general, use int rather than unsigned types. The use of int is common throughout C#, and it is easier to interact with other libraries when you use int.

## 7.3 Functionalities

**7.3.1 Home Page:** User can view the Home Page

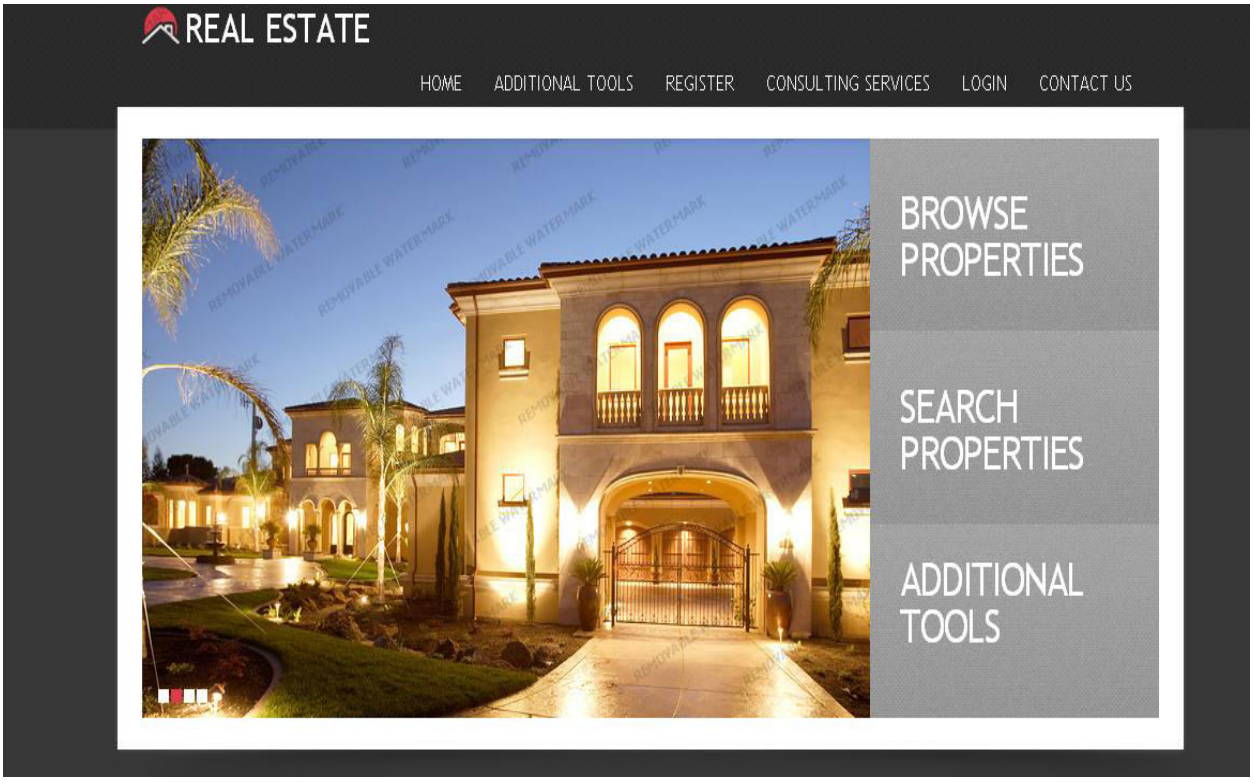



Fig 7.3.1 Home Page

### 7.3.2 Login Panel:

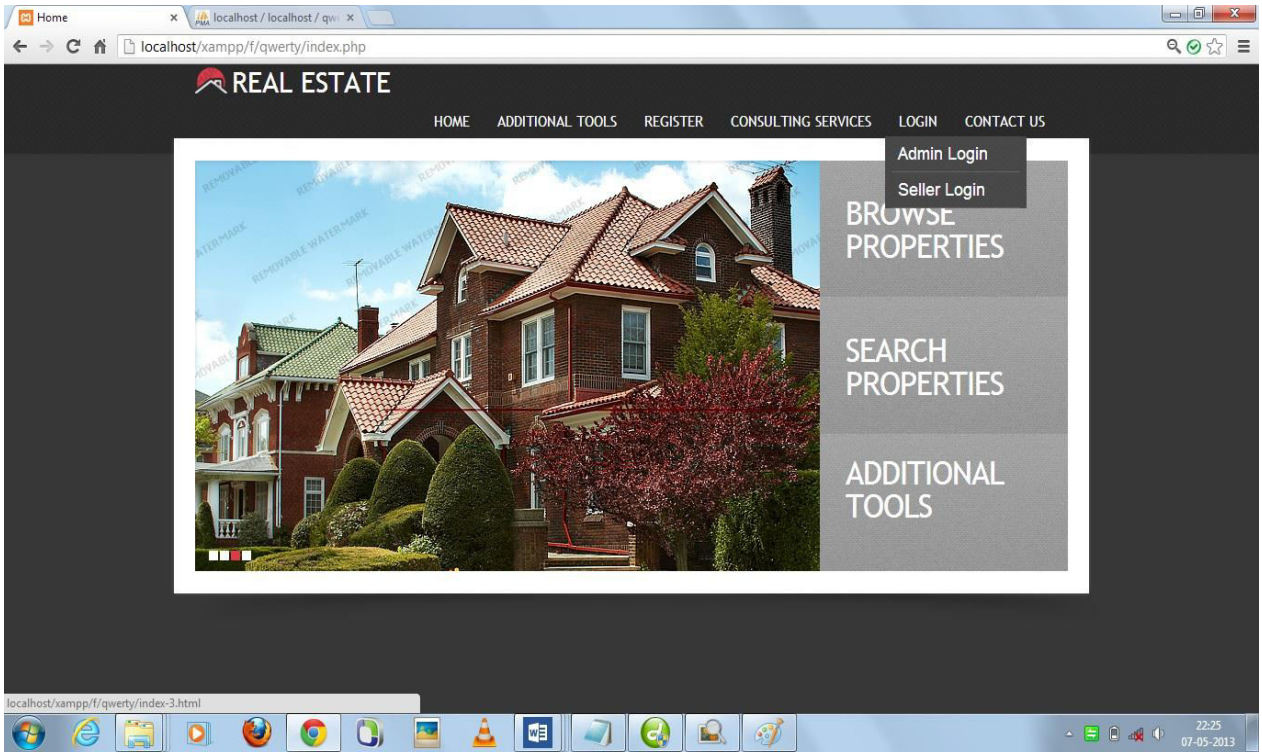



Fig 7.3.2- Login Panel

7.3.3 **Registration Page:** Registration Form for New User to get registered

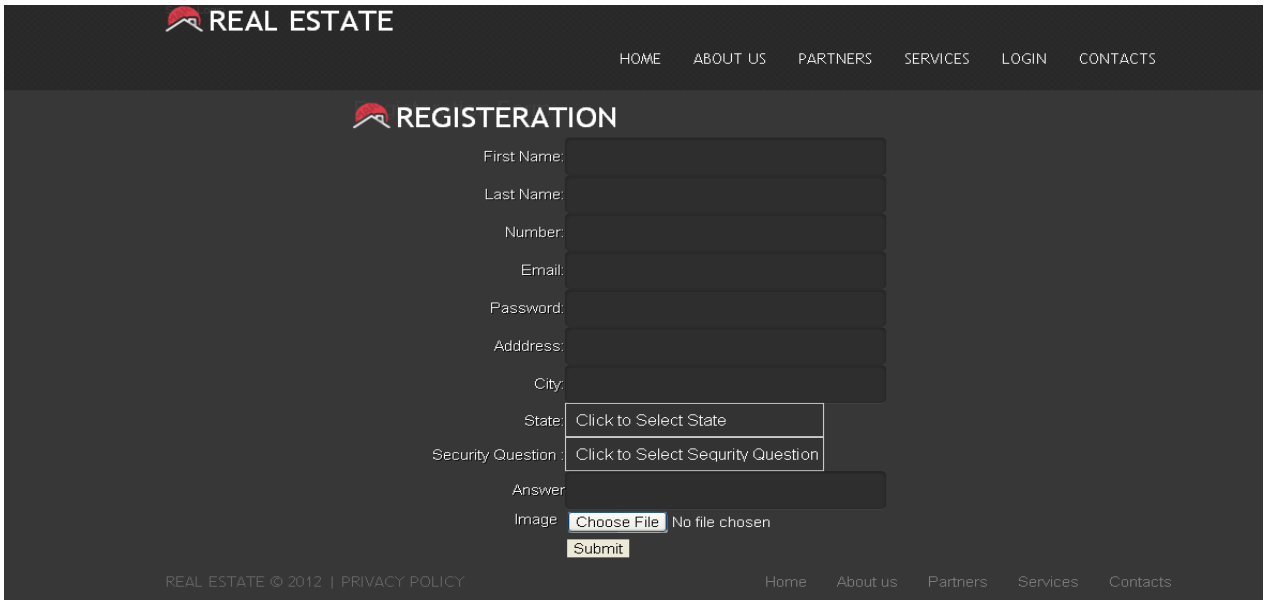



Fig 7.3.3 Registration Form

### 7.3.4 **Registration Validation:** Validation Added

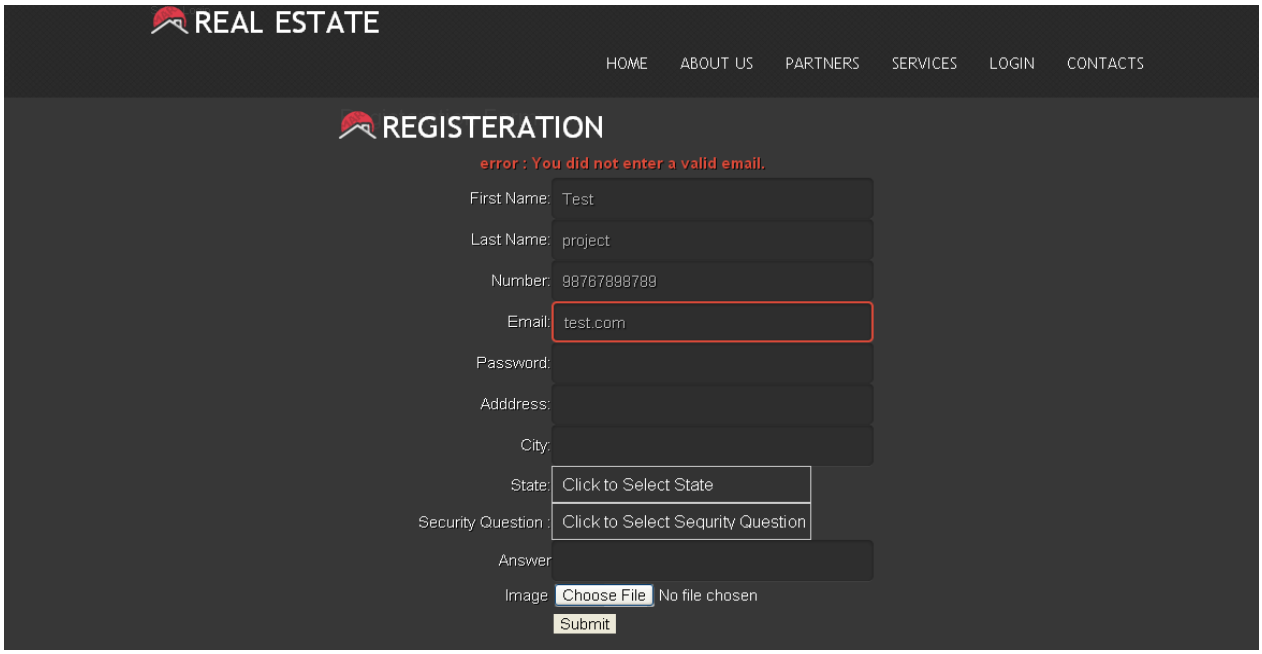



Fig 7.3.4 Validation

### 7.3.5 **Admin:** Admin can login to the system.

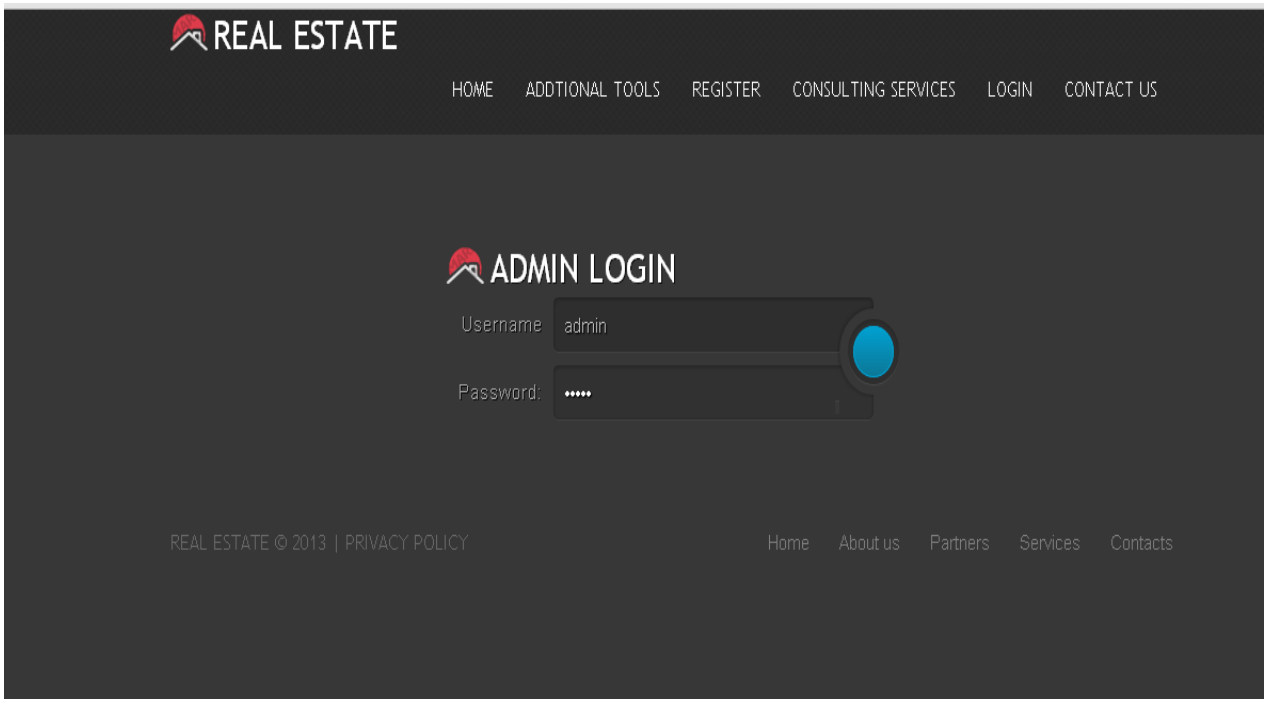



Fig 7.3.5 Admin

### 7.3.6 **Admin Validates :** Admin can Validate the User Requests

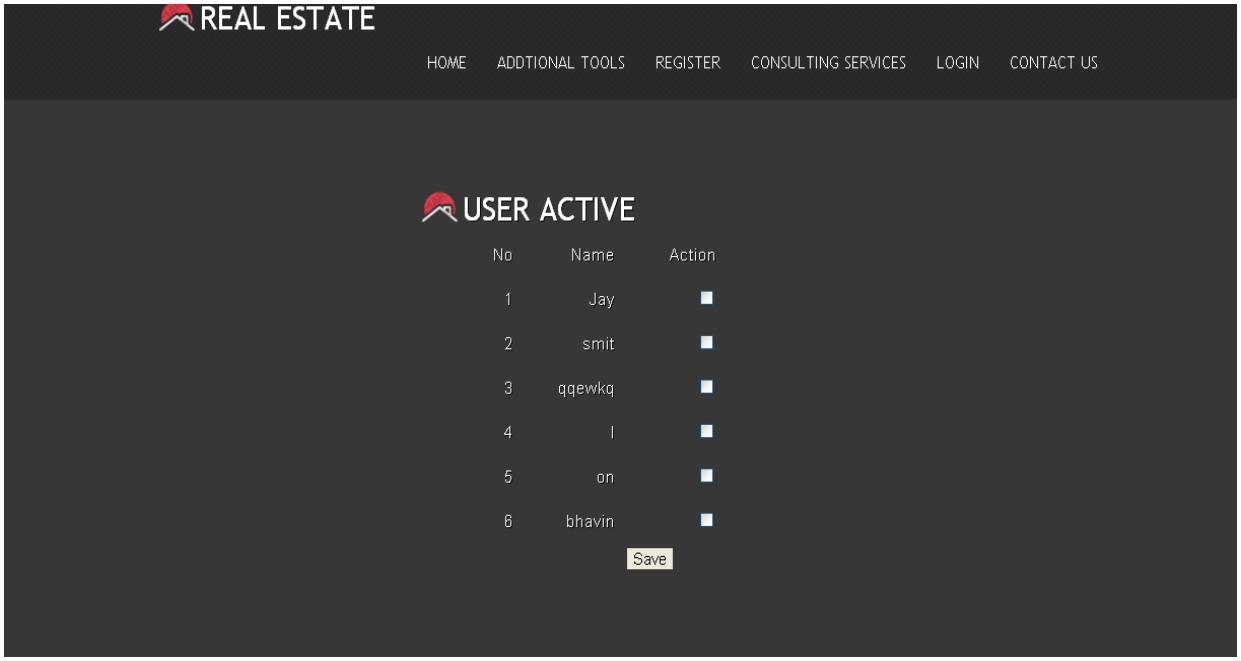



Fig 7.3.6 Admin Validates

### 7.3.7 **Seller Login:** Registered Seller Login

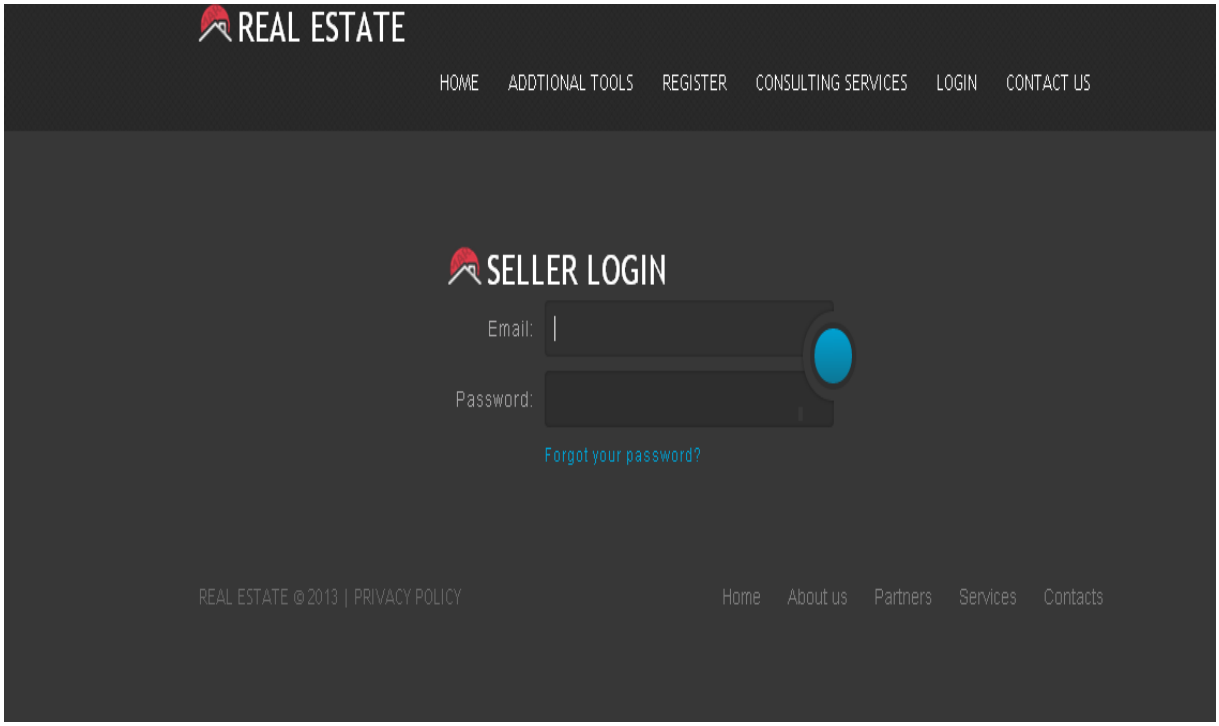



Fig 7.3.7 Seller Login

**7.3.8 User Profile:** User Profile wherein he/she can add listing and see existing listings

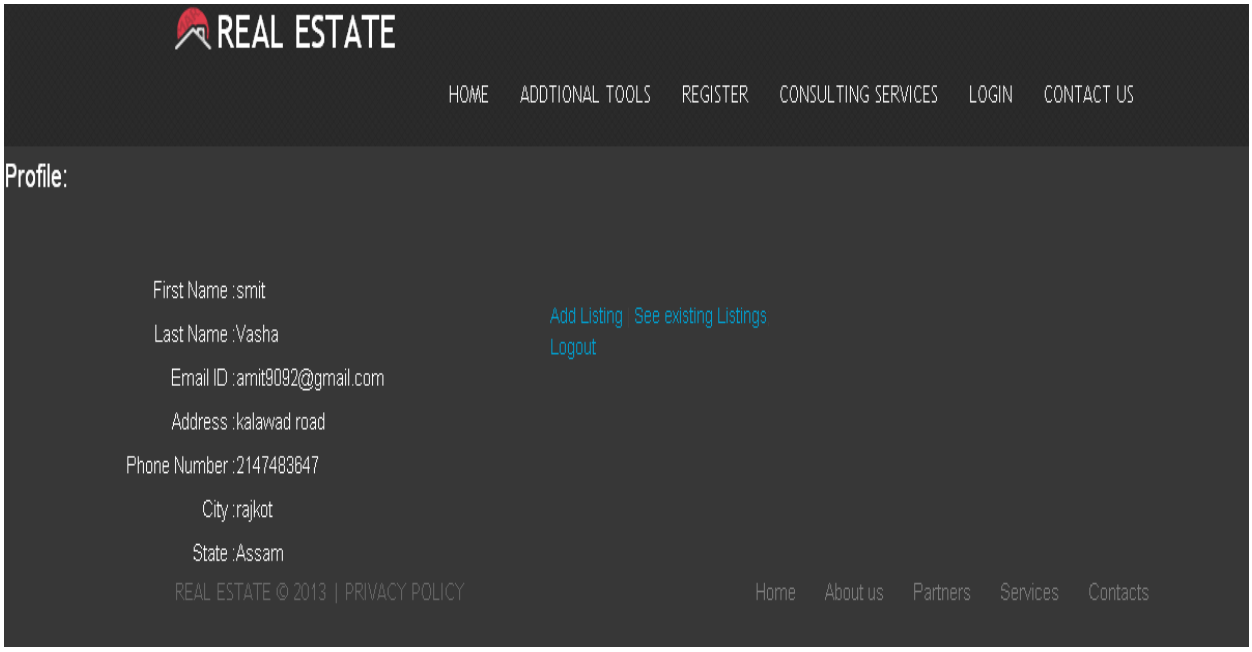



Fig 7.3.8 User Profile

**7.3.9 Existing Listing :** User can watch the existing listing

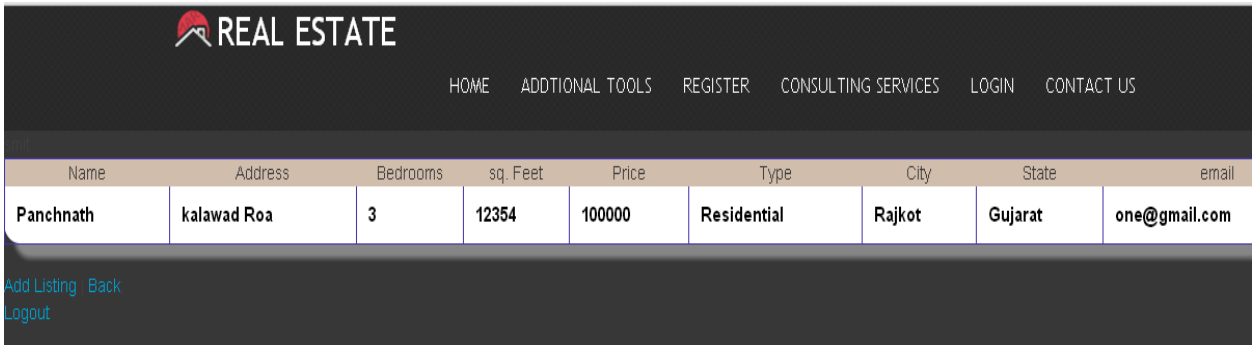



Fig 7.3.9 User's Existing Listing

**7.3.10 Add Listing :** User can add Listing i.e. add property along with Validation.

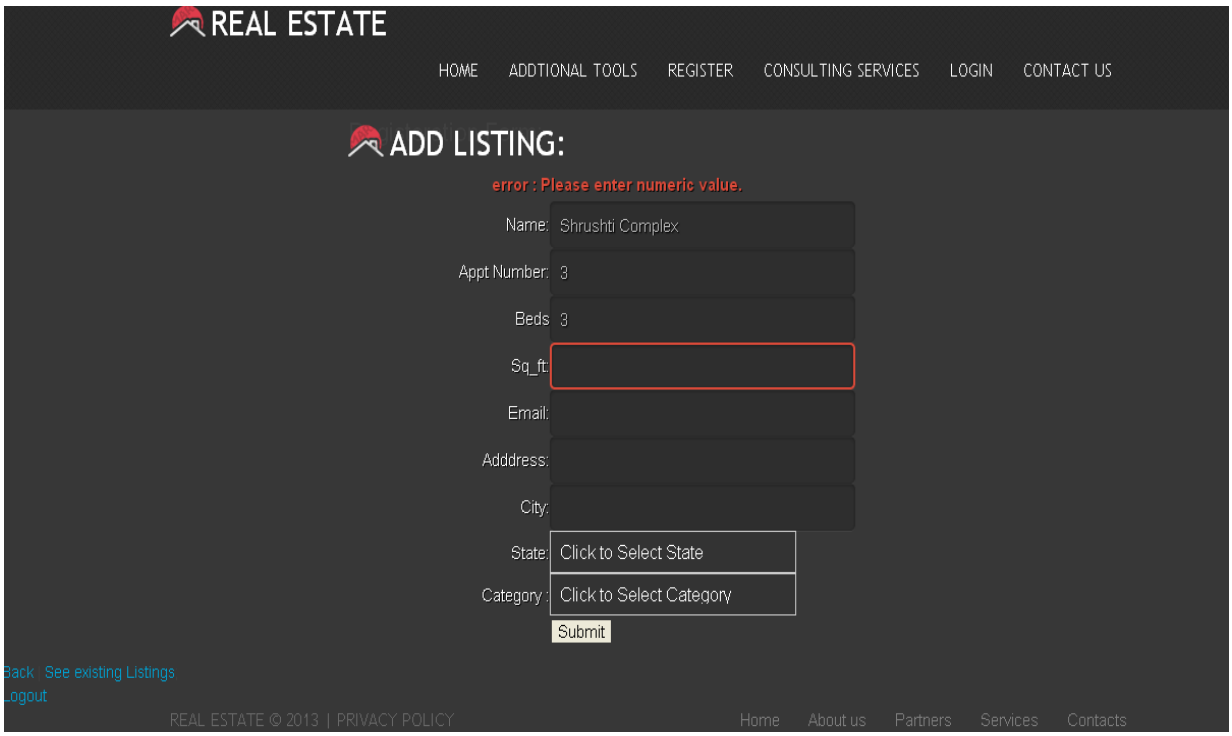



Fig 7.3.10 Add Listing

**7.3.11 Additional Tools:** Additional tools added.

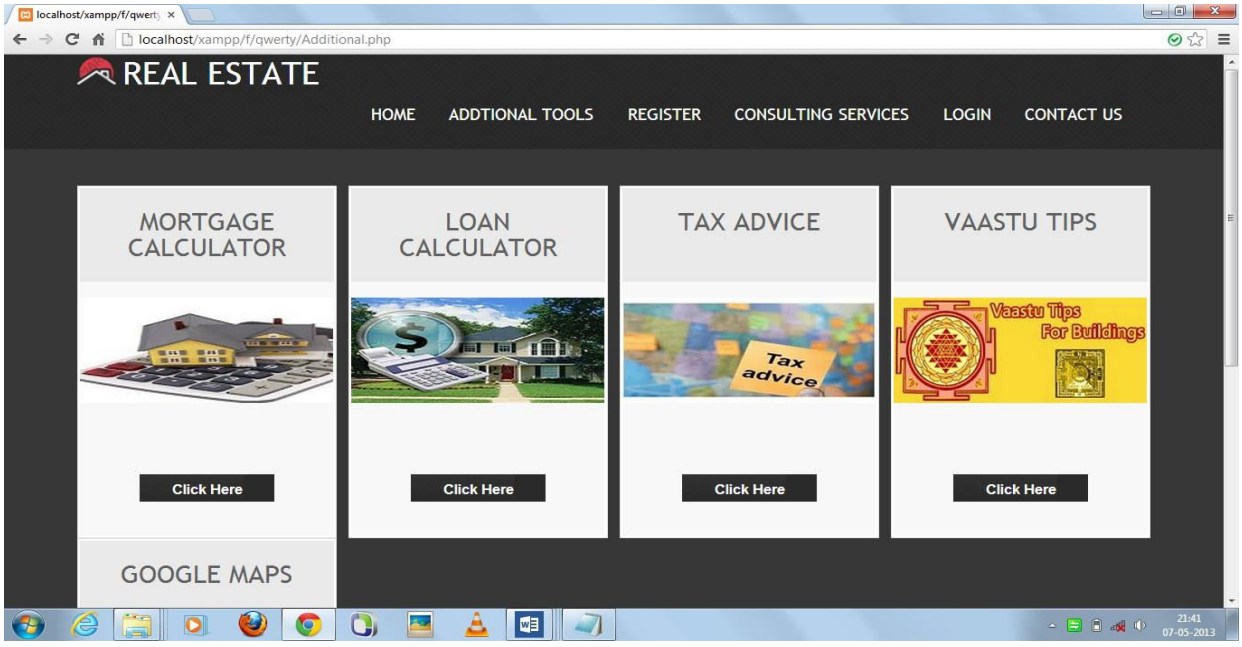



Fig 7.3.11 Additional tools menu

### 7.3.12 **Mortgage Calculator:** User can use mortgage calculator

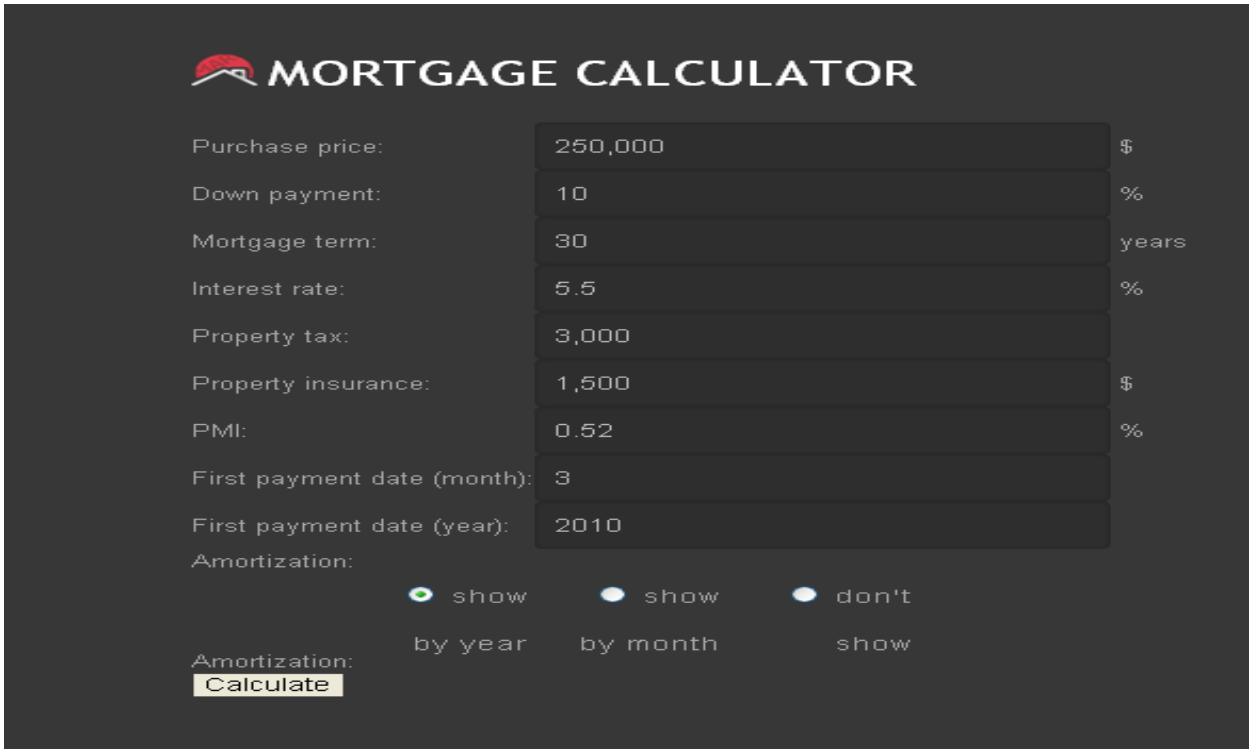



Fig 7.3.12 Mortgage Calculator

### 7.3.13 **Mortgage Results:** Graph Representation of Results of Mortgage Calculator.

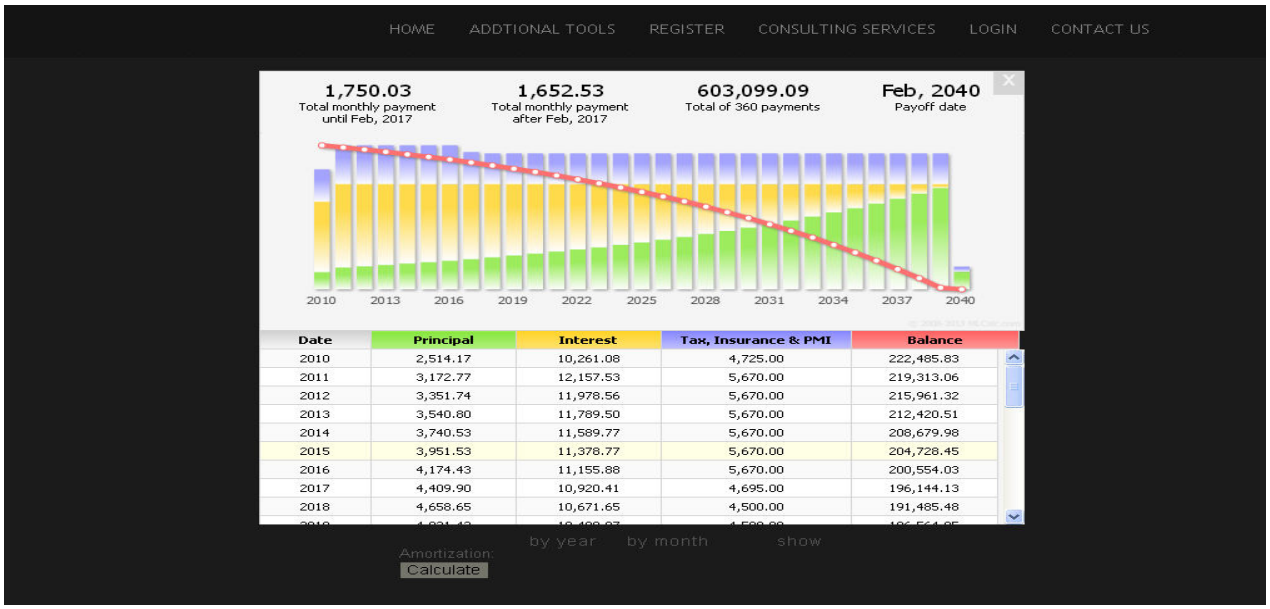



Fig 7.3.13 Mortgage Calculator Results

### 7.3.14 **Loan Calculator:** User can use Loan Tools

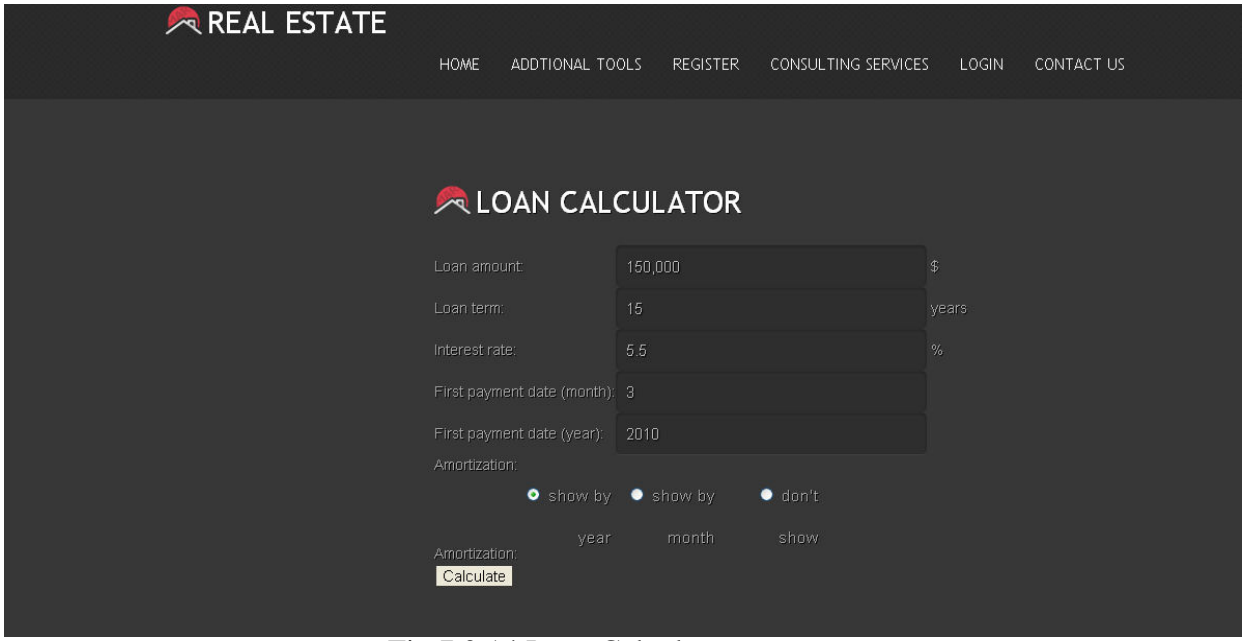
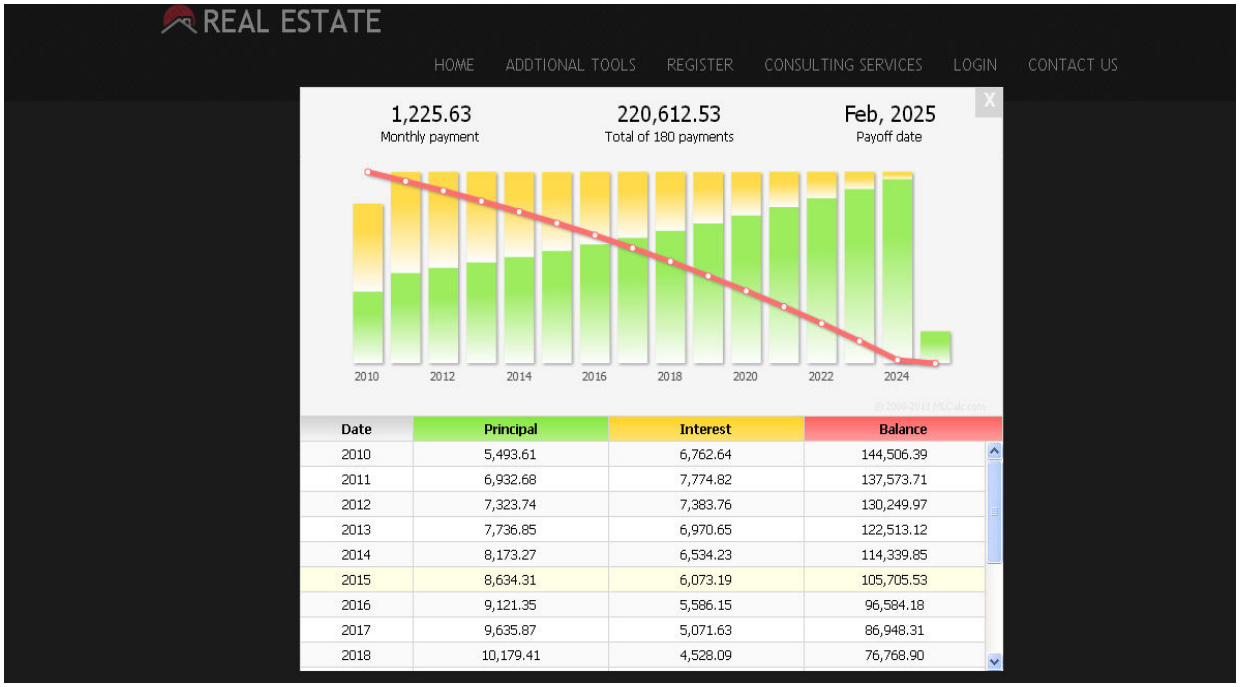



Fig 7.3.14 Loan Calculator

### 7.3.15 **Result:** Loan Calculator's Result




| Date | Principal | Interest | Balance |
|---|---|---|---|
| 2010 | 5,493.61 | 6,762.64 | 144,506.39 |
| 2011 | 6,932.68 | 7,774.82 | 137,573.71 |
| 2012 | 7,323.74 | 7,383.76 | 130,249.97 |
| 2013 | 7,736.85 | 6,970.65 | 122,513.12 |
| 2014 | 8,173.27 | 6,534.23 | 114,339.85 |
| 2015 | 8,634.31 | 6,073.19 | 105,705.53 |
| 2016 | 9,121.35 | 5,586.15 | 96,584.18 |
| 2017 | 9,635.87 | 5,071.63 | 86,948.31 |
| 2018 | 10,179.41 | 4,528.09 | 76,768.90 |

Fig 7.3.15 Loan Calculator Results

7.3.16 **Vastu Tips:** User can use Vastu Tips for residential as well as commercial

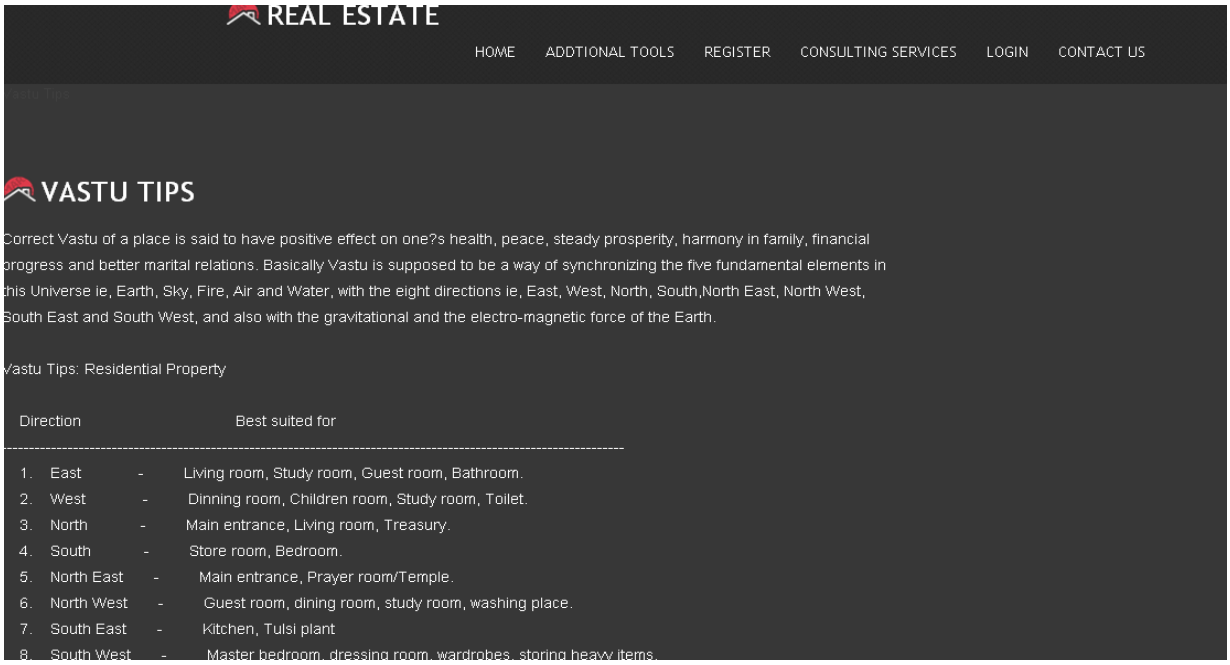



Fig 7.3.16 Vastu Tips

**7.3.17 Consulting Advices:** User can use the consulting advices module

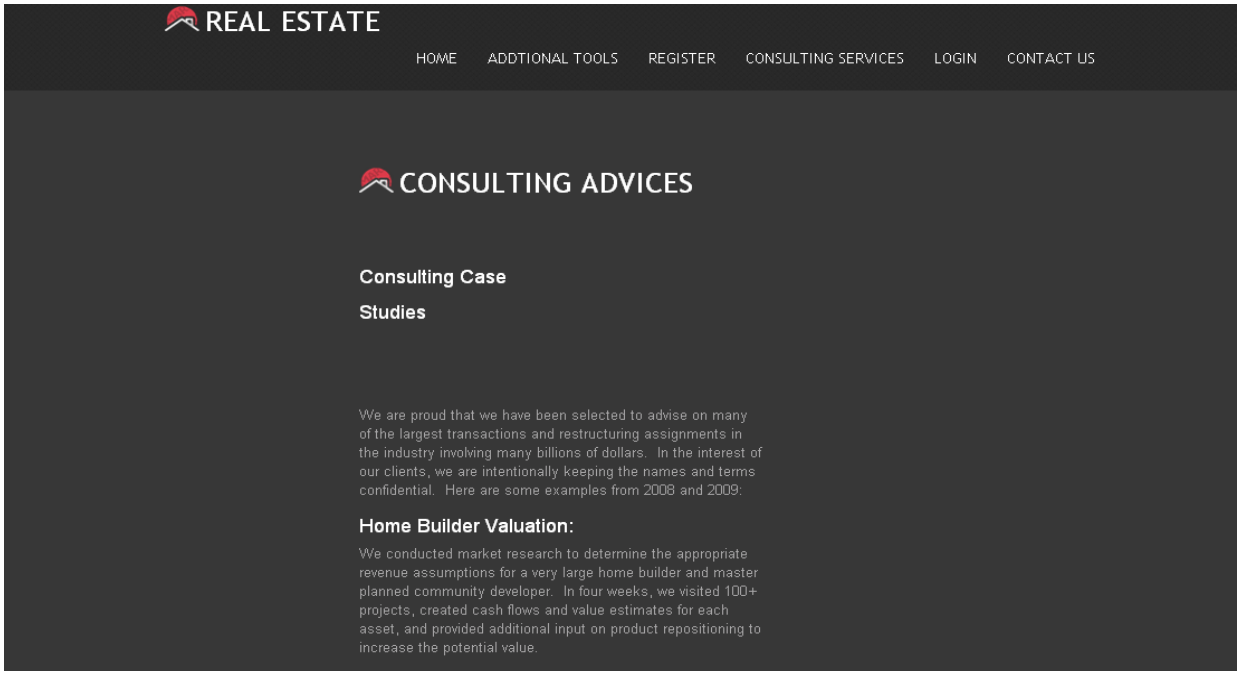



Fig 7.3.17 Consulting Advices

**7.3.18 Searching Module:** In order to search for the property

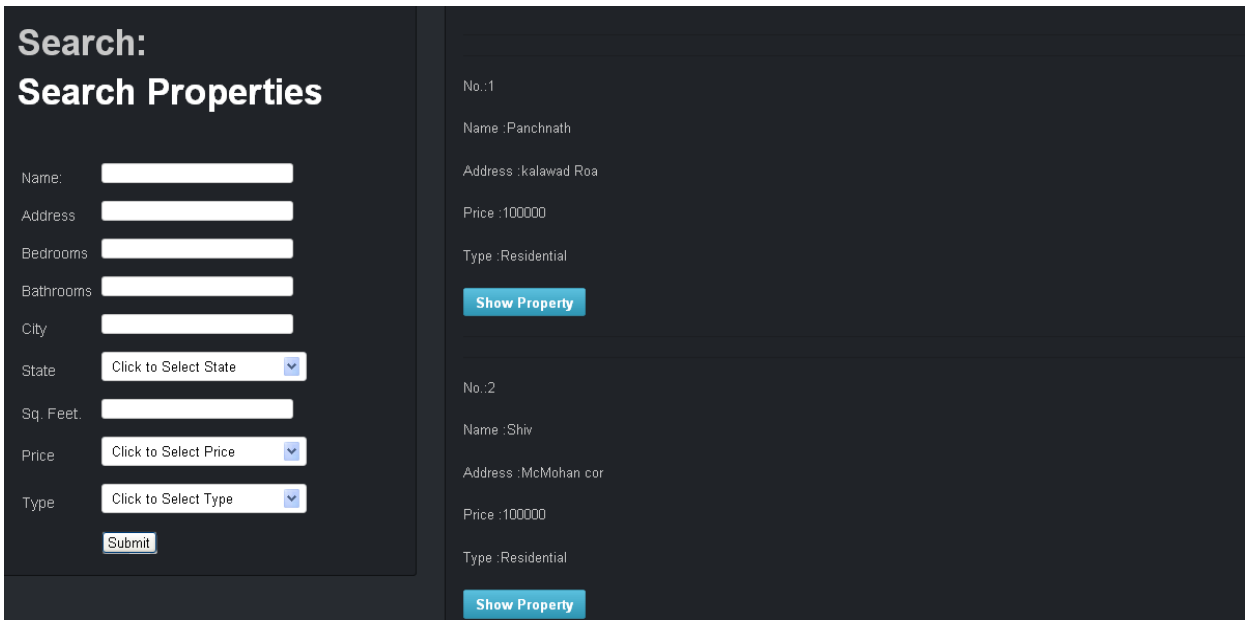



Fig 7.3.18 Searching Module

**7.3.19 Search with Constraint :** Searching done with constraint and Keyword

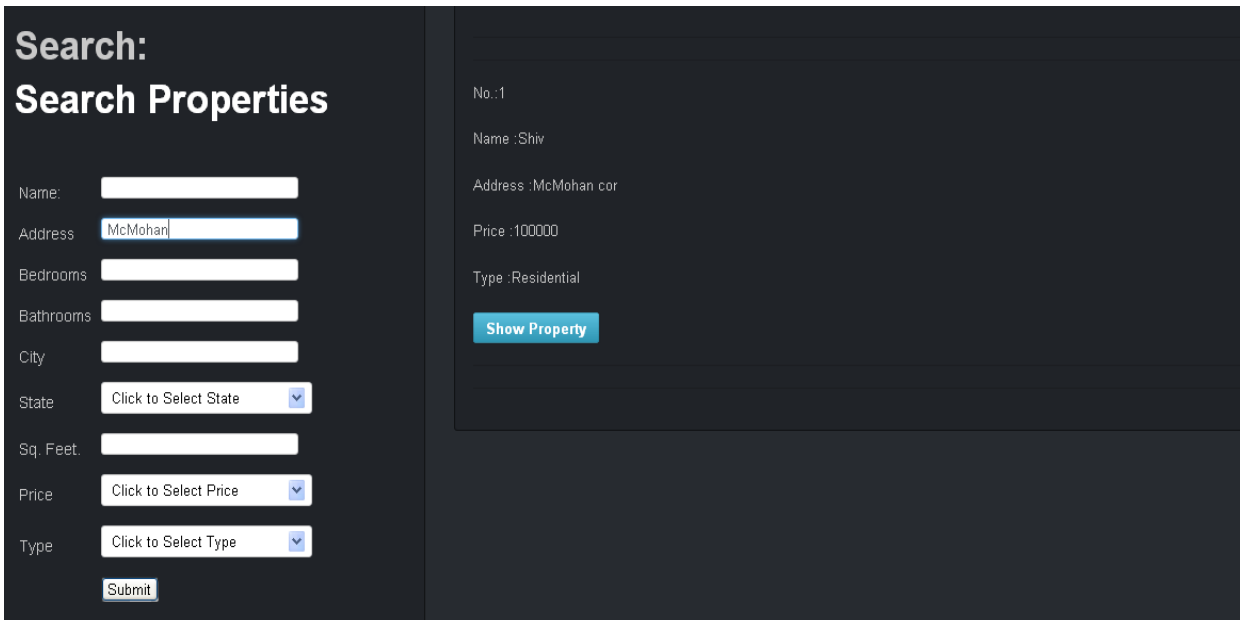



Fig 7.3.19 Search with constraint

**7.3.20 Property Information:** User can get property info.

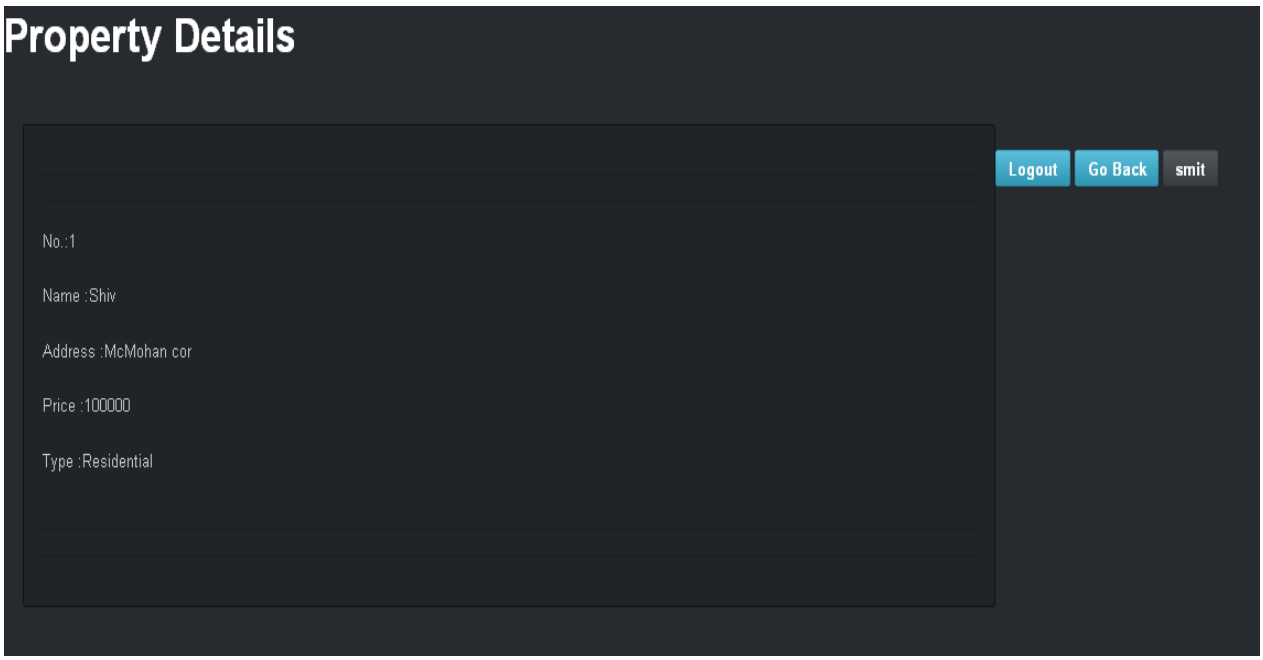



Fig 7.3.20 Property Information

**7.3.21 Browse Listing:** User can Browse the listing of properties

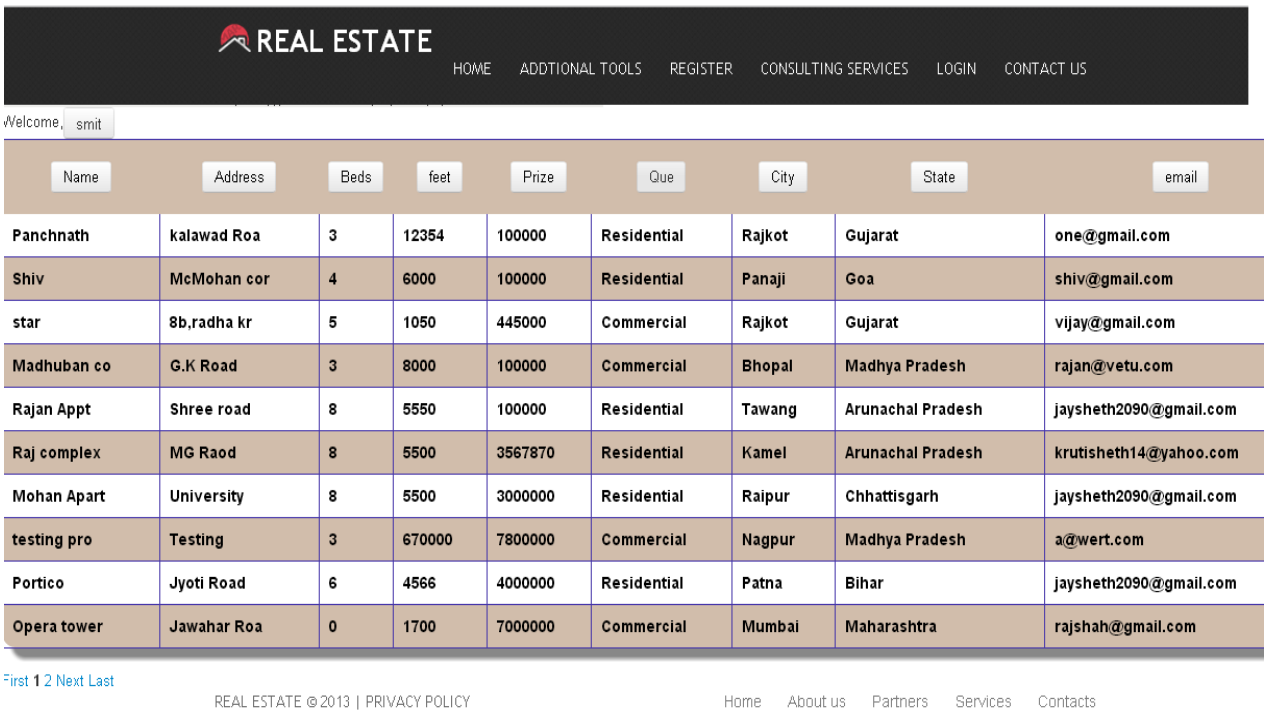



Fig 7.3.21 Browse Listing

**7.3.22 Listing with Pagination:** User can check Properties on different Pages.

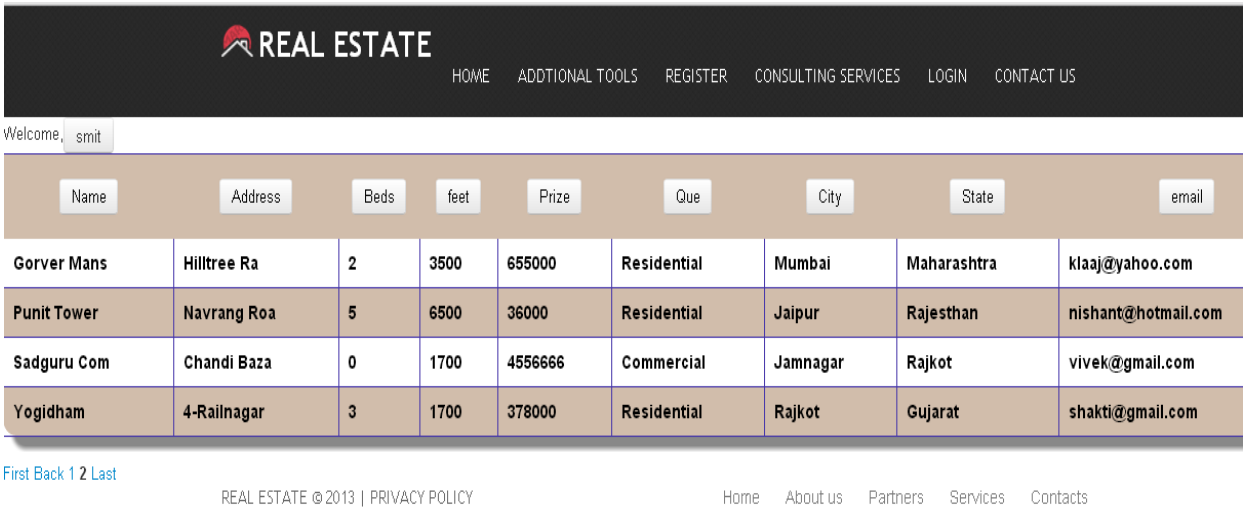



Fig 7.3.22 Listing with Pagination

**7.3.23 Tables :** Database Tables.

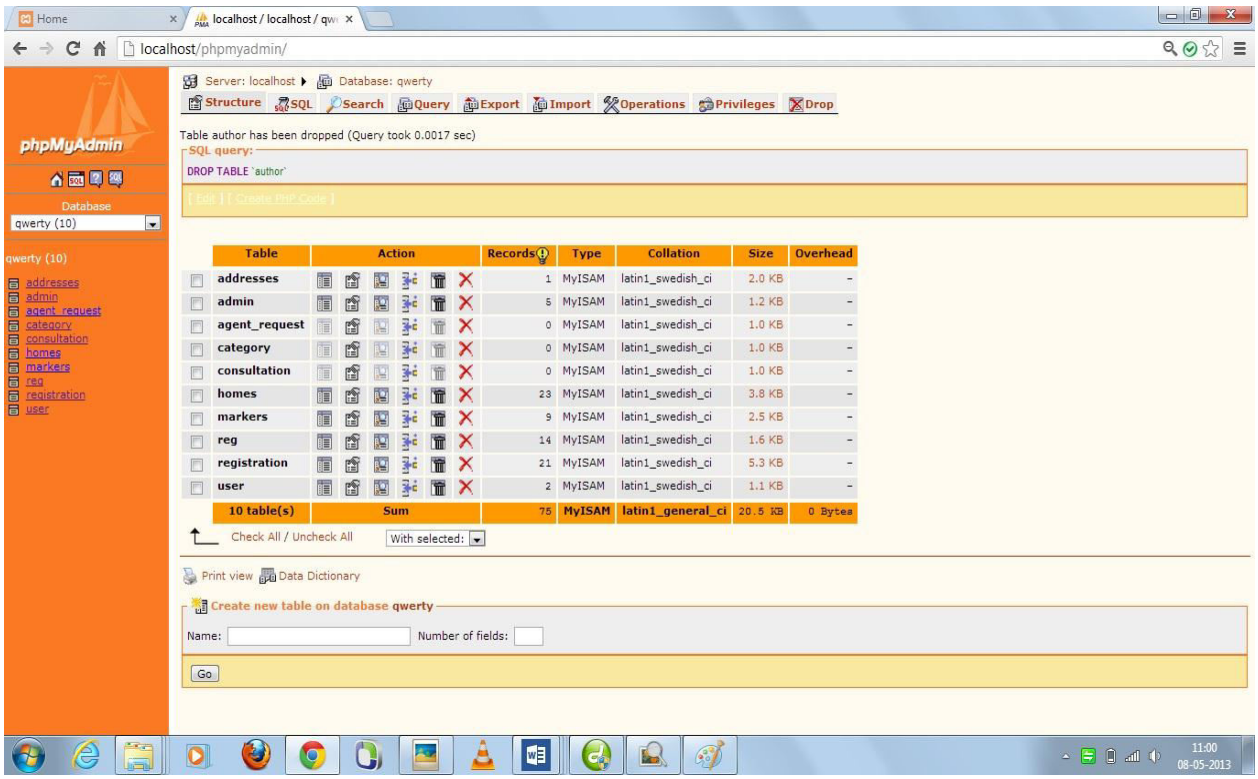



Fig 7.3.23 Database Tables

# 8. Testing

# 8. Testing

## 8.1 Testing Strategy

This chapter describes several approaches to testing software. Software testing must be planned carefully to avoid wasting development time and resources. Testing begins “in the small” and progresses “to the large”. Initially individual components are tested and debugged. After the individual components have been tested and added to the system, integration testing takes place. Once the full software product is completed, system testing is performed. The Test Specification document should be reviewed like all other software engineering work products.

Different strategies may be adopted depending on the type of system to be tested and the development process used. The testing strategies are

- **Unit Testing**

  Unit testing, also known as component testing, refers to tests that verify the functionality of a specific section of code, usually at the function level. In an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors.

  These types of tests are usually written by developers as they work on code (white-box style), to ensure that the specific function is working as expected. One function might have multiple tests, to catch corner cases or other branches in the code. Unit testing alone cannot verify the functionality of a piece of software, but rather is used to assure that the building blocks the software uses work independently of each other.

  Unit testing is a software development process that involves synchronized application of a broad spectrum of defect prevention and detection strategies in order to reduce software development risks, time, and costs. It is performed by the software developer or engineer during the construction phase of the software development lifecycle. Rather than replace traditional

- **Integration Testing**

  Integration testing is any type of software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an iterative way or all together ("big bang"). Normally the former is considered a better practice since it allows interface issues to be localised more quickly and fixed.

  Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.

- **System Testing**

  System testing tests a completely integrated system to verify that it meets its requirements.

  In addition, the software testing should ensure that the program, as well as working as expected, does not also destroy or partially corrupt its operating environment or cause other processes within that environment to become inoperative (this includes not corrupting shared memory, not consuming or locking up excessive resources and leaving any parallel processes unharmed by its presence).

- **Acceptance Testing**

  Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.

## 8.2 Testing Methods

### 8.2.1 Test Case

A test case in software engineering is a set of conditions or variables under which a tester will determine whether an application or software system is working correctly.

Test cases are often referred to as test scripts, particularly when written. Written test cases are usually collected into test suites.

- **Test Case for Admin:**

**Test Case: Check Login (valid)**

| | |
|---|---|
| Test Case ID | 1 |
| Description | Check Login (Valid) |
| Input | Enter username and Password |
| Action | Check for the validation |
| Expected Result | The Employee will be directed admin home page |
| Actual Result | Go to the home page |

Table 8.2.1.1 Check Login (Valid)

**Test Case: Check Login (invalid)**

| | |
|---|---|
| Test Case ID | 2 |
| Description | Check login - Invalid |
| Input | Enter username and Password |
| Action | Check for the validation |
| Expected Result | The Employee will be prompt for the invalid Username/Password, make log to invalid entry |
| Actual Result | Return back to login page with message indicating the invalid login |

Table 8.2.1.2 Check Login (In Valid)

**Test Case: Add User:**

| Test Case ID | 3 |
|---|---|
| Description | Add Employee |
| Input | Enter User Id, Password. |
| Action | Check for the required field validations |
| Expected Result | If valid than user will be added into the database |
| Actual Result | Return with the message box indicating the valid entry in the database |

Table 8.2.1.3 Add Employee

**Test Case: View User (Seller)**

| Test Case ID | 4 |
|---|---|
| Description | View Employee |
| Input | Enter employee id, name, surname |
| Action | Check for the database entry |
| Expected Result | If the search matches then user will be displayed |
| Actual Result | User will be displayed in gird view |

Table 8.2.1.4 View Employee

**Test Case: Delete User**

| Test Case ID | 5 |
|---|---|
| Description | Delete Employee |
| Input | Select Delete(option) in Grid view |
| Action | User entry will be deleted |
| Expected Result | User will be deleted |
| Actual Result | User will be deleted |

Table 8.2.1.5 Delete Employee

**Test Case: Edit User**

| | |
|---|---|
| Test Case ID | 6 |
| Description | Edit Employee |
| Input | Select(option) in Grid view, salary, designation |
| Action | User data will be load into new form |
| Expected Result | User data will be updated |
| Actual Result | User data will be updated |

Table 8.2.1.6 Edit Employee

## 8.3 Test Summary

To test our project we have carried out different type of testing tools. We have tested the system with selenium testing tool. We have also used the unit testing. First we have tested the each module separately and we have tested the entire system with selenium tool. Most of the test cases are successfully executed. We have also checked the validation testing

# 9. Limitation and Future Enhancement

# 9. Limitation and Future Enhancement

A more reliable facility for the backup should be provided.

- Mobile Interface: In future user may be able to perform all the operation through mobile device.
- Mail server can be implemented.
- Mobile services can be implemented to get the new available property to the user via SMS.
- Discussion Module can also be implemented.

It is not possible to develop a system that makes all the requirements of the user. User requirements keep changing as the system is being used. Some of the future enhancements that can be done to this system are:

- As the technology emerges, it is possible to upgrade the system and can be adaptable to desired environment.
- Because it is based on object-oriented design, any further changes can be easily adaptable.
- Based on the future security issues, security can be improved using emerging technologies.
- Tutorial module can be added.
- Sub admin module can be added
- Video conference can be added to our system
- Auto backup of the database can be generated

# 10. Conclusion

# 10. Conclusion

- The working on the project “Real-Estate” Was an extremely learning experience. We came across a no of new concepts and also enhanced our knowledge.
- Although we faced some problem during the designing and the implementation part but due to the moral support of our guide we were able to overcome the situation.
- This project is an entirely team effort and both of us completely enjoyed being part of this project.

The project has been completed as per the schedule. The system can be enhanced to provide different functionality like user can represent file through speech. The goals that are achieved by the software are:

- Instance Access
- Improved productivity
- Optimum utilization of resources
- Portable and flexible for further enhancement
- Simplification of the operations
- Less processing time and getting required information
- User friendly
- Efficient management of records

# Appendix A – Data Dictionary

It is a record of metadata about an actual data. It may be manually defined with the elements of system, processes and storage of data.

It provides additional information about the system. The data dictionary is developed during system analysis phase which involves different system requirements. If the system analyst wants to know how many characters are represented in a data item, by which different names it is referenced in the system is properly managed by developing data dictionary. These elements of data are organized in a well-structured manner to meet user request and organization needs. It is a common source of definitions for the users and investigations. Data dictionaries are an integral component of structured analysis as they fully describe the project activities. It also manages the details in a large system

Data Dictionaries are an integral component of structured analysis, since data flow diagrams by themselves do not fully describe the subject of the investigation. The data dictionary provides additional information about the system.

**Registration**:

| # | Column | Type | Collation | Attributes | Null | Default | Extra |
|---|---|---|---|---|---|---|---|
| 1 | user_id | int(11) | | | No | *None* | AUTO_INCREMENT |
| 2 | fname | varchar(20) | latin1_swedish_ci | | No | *None* | |
| 3 | lname | varchar(20) | latin1_swedish_ci | | No | *None* | |
| 4 | email | varchar(20) | latin1_swedish_ci | | No | *None* | |
| 5 | password | varchar(20) | latin1_swedish_ci | | No | *None* | |
| 6 | ph_no | int(10) | | | No | *None* | |
| 7 | address | varchar(100) | latin1_swedish_ci | | No | *None* | |
| 8 | city | varchar(50) | latin1_swedish_ci | | No | *None* | |
| 9 | state | varchar(50) | latin1_swedish_ci | | No | *None* | |
| 10 | zipcode | int(20) | | | No | *None* | |
| 11 | Que | text | latin1_swedish_ci | | No | *None* | |
| 12 | ans | text | latin1_swedish_ci | | No | *None* | |
| 13 | validated | text | latin1_swedish_ci | | No | *None* | |

**Admin**:

| # | Column | Type | Collation | Attributes | Null | Default | Extra |
|---|---|---|---|---|---|---|---|
| 1 | uname | text | latin1_swedish_ci | | No | *None* | |
| 2 | pass | text | latin1_swedish_ci | | No | *None* | |

**Homes:**

| # | Column | Type | Collation | Attributes | Null | Default | Extra |
|---|---|---|---|---|---|---|---|
| 1 | **Name** | varchar(11) | latin1_swedish_ci | | No | *None* | |
| 2 | **Address** | varchar(11) | latin1_swedish_ci | | No | *None* | |
| 3 | **beds** | int(11) | | | No | *None* | |
| 4 | **feet** | int(11) | | | No | *None* | |
| 5 | **City** | text | latin1_swedish_ci | | No | *None* | |
| 6 | **State** | text | latin1_swedish_ci | | No | *None* | |
| 7 | **Prize** | int(11) | | | No | *None* | |
| 8 | **ID** | int(11) | | | No | *None* | AUTO_INCREMENT |
| 9 | **email** | text | latin1_swedish_ci | | No | *None* | |
| 10 | **ph_no** | int(11) | | | No | *None* | |
| 11 | **Que** | text | latin1_swedish_ci | | No | *None* | |
| 12 | **ans** | text | latin1_swedish_ci | | No | *None* | |
| 13 | **seller** | text | latin1_swedish_ci | | No | *None* | |

**Contact Us:**

| # | Column | Type | Collation | Attributes | Null | Default | Extra |
|---|---|---|---|---|---|---|---|
| 1 | **Subject** | text | latin1_swedish_ci | | No | *None* | |
| 2 | **Details** | text | latin1_swedish_ci | | No | *None* | |
| 3 | **Name** | text | latin1_swedish_ci | | No | *None* | |
| 4 | **email** | text | latin1_swedish_ci | | No | *None* | |

**Consulting:**

| # | Column | Type | Collation | Attributes | Null | Default | Extra |
|---|---|---|---|---|---|---|---|
| 1 | **Subject** | text | latin1_swedish_ci | | No | *None* | |
| 2 | **Details** | text | latin1_swedish_ci | | No | *None* | |
| 3 | **Name** | text | latin1_swedish_ci | | No | *None* | |
| 4 | **email** | text | latin1_swedish_ci | | No | *None* | |

# Appendices 2 – References


[1] http://www.emyrtlebeachrealestate.com

[2] http://www.anytimeproperty.com

[3] http://www.exitgrandstrand.com

[4] http://www.sfrea.com

[5]http://www.99acres.com

[6] http://www.w3schools.com

[7] http://www.phpcode.com

[8] http://www.sqltuner.com