**Online Home Rental Management System**

*A Project Submitted in Partial Fulfillment of the Requirements for the*

*Degree of*

Bachelor of Science in Computer Science and Engineering

*by*

**ROKIBUL HASAN**

CSE 05406751

&

**SUMAN SARKER JOY**

CSE 05406745

Supervised by: NAZIA HOSSAIN

Senior Lecturer

|  |
| --- |
| C:\Users\Adnan\Desktop\1.jpg |

Department of Computer Science and Engineering

STAMFORD UNIVERSITY BANGLADESH

June 2018

**Abstract**

An efficient and dynamic home rental system which users can access through their mobile phone with no hassle is a desire of every tenant in Bangladesh. In this era of technology no one wants to roam around to find a vacant house. House for living is one of the basic needs of people, therefore to make it easier and accessible by every people we represent a home rental system which provides every needed facility. Our home rental system will have dynamic values for searching and easy way to rent a home. . The Rental Management System is best Suitable an the owners because time save and the only contact and the eligible person and there is no need to explain the room details and the speak. Customer and seller directly communicate each others to sell their Property. And provide good facility. Customer can also rent & sell house location wise.

**Approval**

The project report “Online Home Rental Management System” submitted by ROKIBUL HASAN ID: CSE O54 06751, SUMAN SARKER JOY ID: CSE 054 06745, to the Department of Computer Science & Engineering, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science (B.Sc.) in Computer Science & Engineering and as to its style and contents.

Board of Examiner’s Name, Signature and Date:

|  |  |  |
| --- | --- | --- |
| **……………………………...** | **……………………………..** | **……………………………..** |
|  |  |  |
| **(Board Member 1)** | **(Board Member 2)** | **(Board Member 3)** |
| Date: | Date: | Date: |

Supervisor’s Signature and Date:

**……………………………...**

**Nazia Hossain**

Date:

**Declaration**

We, hereby, declare that the work presented in this Project is the outcome of the investigation performed by us under the supervision of Nazia Hossain, Senior Lecturer, Department of Computer Science \& Engineering, Stamford University Bangladesh. We also declare that no part of this Project and thereof has been or is being submitted elsewhere for the award of any degree or Diploma.

Signature and Date:

**……………………………...**

**ROKIBUL HASAN**

Date:

**……………………………...**

**SUMAN SARKER JOY**

Date:

Dedicated to…

**Acknowledgements**

At first I would like to thank the almighty ALLAH for giving me ability, chance and such cooperating supervisor. With Her mercy, today I am completing my thesis work successfully.

I would like to take the opportunity to express my gratitude to Nazia Hossain, my respected supervisor. Although she was always loaded with several other activities, he gave me more than enough time in this work. He not only gave me time but also proper guidance and valuable advice whenever I faced with some difficulties. Her comments and guidance helped me in preparing my project report.

I would like to thank all my respected teachers, who inspired me in every step. I am also thankful to my friends who helped me in a number of ways by providing various resources and moral support.

Last but not least I am grateful to my family; who are, always with me in my every step of life.

**Table of Contents**

List of Figures 1

List of Tables 2

1: Introduction 4

1.1 Problem Defination …………………………………………...4

1.2 Motivation 5

1.3 Chapter Summary 5

2: Background Study 7

2.1 Existing related system……………………………………………………………….7

2.2 Literature Review…………………………………………………………………….8

2.2.1 PHP………………………………………...……………………………………8

2.2.2 HTML……………………………………...……………………..………….…16

2.2.3 CSS………………………………………...……………………………...……17

2.2.4 Jquery…………………………………...……………………..….……………17

2.2.5 MySQL………………………………………...………………………………18

2.3 Chapter Summary………………………………………………………………….18

3: System Requirement 19

3.1 Software & Hardware*……………………………………………………………..………....*20

3.1.1 Software Requirements………………………………………………………………………………………..20

3.1.2 Hardware Requirements………………………………………………………………………………………..20

3.2 Software setup process………………………………………………………………20

3.3 Chapter Summary………………………………………………………………….31

4: Project Details 32

4.1 Database Details 33

4.2 System Design 37

4.2.1 Timeline Chart………………………………………………………...……….37

4.2.2 System Flowchart……………………………………………………...……….38

4.2.3 E-R Diagram…………………………………...……………………...……….39

4.2.4 Use Case Diagram…………………………………………………...…..…….40

4.2.5 Activity Diagram……………………………………………………...……….42

4.2.6 Sequence Diagram……………………………………..……………...……….45

4.2.7 Collaboration Diagram………………………………………………...……….46

4.3 Input Project Design 49

4.4 Chapter Summary 56

5: Conclusion & Future Work 57

5.1 Future wrok 58

References 59

# List of Figures

Figure 3.0.2: Xampp Server Setup Process………………………………………….……….23

Figure 3.0.3: Xampp Control Panel………………………………………………………….27

Figure 3.0.4: CodeIgniter Download…………………………………………………………28

Figure 3.0.1:CodeIgniter Folder……………………………………………………………..29

Figure 4.0.2: Timeline Chart…………………………………………………………………37

Figure 4.0.3: System FlowChart…………………………………………………………......38

Figure 4.0.4:E-R Diagram……………………………………………………………………39

Figure 4.0.5:Use Case Diagram Admin…………………………………………………......40

Figure 4.0.5:Use Case Diagram Home Owner…………………………………………........41

Figure 4.0.6:Use Case Diagram Home Seeker…………………………………………........41

Figure 4.0.6:Activity Diagram Admin……………………………………………………….42

Figure 4.0.8:Activity Diagram Supplier……………………………………………………..43

Figure 4.0.9:Activity Diagram Home Seeker………………………………………………..44

Figure 4.0.7: Sequence Diagram Admin……………………………………………………..45

Figure 4.0.8: Sequence Diagram Supplier……………………………………………………45

Figure 4.0.12: Sequence Diagram Buyer…………………………………………………….46

Figure 4.0.9:Collaboration Diagram Admin…………………………………………………47

Figure 4.0.14:Collaboration Diagram Home Owner…………………………………………48

Figure 4.0.15:Collaboration Diagram Home Seeker…………………………………………48

Figure 4.0.10: User Interface Home………………………………………………….………49

Figure 4.0.11: User Interface Properties……………………………………………………..49

Figure 4.0.12: User Interface Contact………………………………………………………..50

Figure 4.0.13:Contact Form………………………………………………………………….50

Figure 4.0.14:Register New Account………………………………………………………...51

Figure 4.0.15:User Interface Login…………………………………………………………..52

Figure 4.0.16: User Interface Submit Property………………………………………………52

Figure 4.0.17: Admin Panel Login…………………………………………………………...54

Figure 4.0.18: Admin Panel Dashboard……………………………………………………...54

Figure 4.0.19: Admin Panel Property Table…………………………………………………55

Figure 4.0.20: Admin Panel Category Table…………………………………...……………55

Figure 4.0.21: Admin Panel User Table……………………………………………………..56

# List of Tables

Table 4.0.1: Users Table…………………………………………………………………....33

Table 4.0.2: Contact\_info Table…………………………………………………………….34

Table 4.0.3:Property Table………………………………………………………………….35

Table 4.0.4:Category Table……………………………………………………………...…36

Table 4.0.5: Booking Table…………………………………………………………………36

# 

# Chapter 1

# Introduction

# 1 Introduction

Every time we need to change our home, But its not an easy think to do if not have any rental system. We worked in such home rental system to build a more dynamic system than the existing ones. Our goal of this project is to make the system as user-friendly as possible. For this purpose, we decided to make a system where the users can search for vacant houses .To minimize any kind of communication gap, we introduce this system between users and admin. Users always want to get a result in the shortest span of time. In many cases, may be users get the reply lately due to internal long manual processes. To solve that, we introduce a dynamic mail system to notify the users’ search result in the least possible time. Therefore, the home rental system hold features like rental search, dynamic mail system, users can search for vacant houses randomly according to their wishes based on unique area or city, rent price as well. Additionally, they can add their wish list if their desired home is not available. There will be a contact feature in which admin and agent can chat directly. All the values which are used for processing result are dynamic in our home rental system. In our system, there is an option whether u want family house or bachelor, we implement IOT here to make more efficient. In Dhaka city, everyday so many new residents enter for job or study. However, recently it has become a trivial issue for the bachelors because none of the house owners are ready to rent their houses to them. Therefore, we create such a safe and user friendly platform for the bachelors so that they can rent house efficiently and in a hassle free procedure.

## Problem Definitions

Roam around to rent a house has always been a hassle for people. Especially, on recent times, people have so many priorities based on which they have to rent their house. Some people want their house to be in the commercial space, or some want in a chaos free space. Some people prefer to choose the area of their house relating the religion they belong. Again there are a lot of people who love pets; therefore they want a house which has pet allowance. Basically, in this era of modernism people want to rent their house like online shopping. To rent a house in physical world has become less popular now a days . No one wants to roam around here and there to search for a house. People would prefer a virtual system to rent a house.

To decode this situation and to represent a hassle free environment to the people, a dynamic system can be implemented. That system would give the tenants the best service for renting houses without any kind of hassle. Government can make one unique system where people can rent house based on their priority instead of having so many rental systems. In that system, all the vacant houses of any district of Bangladesh will be listed there. One system will hold every details of every vacant house from any district, any are. To, make the system more liable, there should be a system by which tenants can verify the owner or agent. Also to analysis the place they will rent for house they need to know the location of that. Hence, every information details which have minimum priority to rent a house will hold by the system. There a one special feature for the bachelors so that they can rent houses efficiently as now a day house owners do not want to rent their houses to the bachelors for safety issue.

* 1. ***Motivation***

According to the universal declaration of human rights by which we get to know that is belong to a human the right to have a proper standard of living. A standard living place is every citizen’s basic right. .Choosing that place is also their right. As technology is growing so fast every single day, it is necessary to make the system the most dynamic approach. By the need of this, a dynamic home rental system is needed where every citizen can choose their house according to their choice and also it has to be most dynamic as people will access this system at anytime from anywhere.

To make the system more dynamic we have implemented MySQL in the following way. In MySQL, first a priority can be set. Then user will give constraints, based upon the constraints and a priority domain will be created using the available data. The domain is mainly all the rentals related to the constraint set on the priority filed. Hence, users can give any constraints and also set any priority; suppose in our system user can set the priority by area and rent range, then he/she will give their constraints.

As a result, it is such a system which will represent a dynamic approach of renting home.

## Chapter Summary

In this chapter illustrates an introductory section of the system. Here we tried to introduce the users to our system. We tried to share the motivations and the objectives of the sytem with the users. In the upcoming chapters I will discuss about Literature Review of the Project, Project Outline, Requirement Analysis, Planning The Development Process, Project Requirement and Features and Workflow.

# 

# Chapter 2

# Background Study

# 2 Background Study

In this chapter, we decide to review some of already existing system. When I shifted to Dhaka city and had this problem. I had no friends here or no one else who can guide me well. My university place is in siddheswari which is south of Dhaka. Luckily found a colleague who was in need of place too. The hunt begins with him in nearby areas.   
Problems we faced  
1. Bachelors!! I never understand point given by those people.  
2. The place Farmgate, it had less number of flat societies and more of raw houses, owned by family who lives there and have a extra few floors on top to give on rent.  
They never have parking area.  
3. No security in those raw houses and you will hear about thief’s stealing bikes every few weeks.  
4. Since working in startup, faced problem of time a lot in searching the flat. Needed to manage a lot that time.  
5. Places in Farm gate area were very small and really costly.  
6. Most of them have water problems, ask if they have kaveri water line, if yes, great!

Finally decided to move a little away from this area and have a good place and less costly too. Mogbazar are really good option nearby and we found a place in society with security guard, no water problems and yes a society gym.

**2.1 Existing related system**

Housing has a central importance to quality of life with considerable economic, social, cultural and personal significance. The focus of this project is basically managing housing for low income, medium and high incomes households. Millions of people are living in Dhaka, Housing is a major problem in this city.

1. In 1980, the first property management system was launched in the market. Property management system for hotel management, also known as PMS, is used only for hotel booking. web: <https://www.webrezpro.com/>
2. The VRBO is restal system in booking system of US.

web: <https://www.vrbo.com>

1. Sabbaticalhomes is rental system in booking system of New York web:<https://www.sabbaticalhomes.com>
2. Rent.com is rental system in booking system of Western countries. web:<https://www.rent.com/>
3. iproperty is rental management system in Malaysia

web: https://www.iproperty.com.

When we started our work, there was no system in Bangladesh. But recently launched a website in Bangladesh. web: <https://www.iproperty.com>.

In this existing system there are no email notification services. And its only for fixed property to rent or sell.

* 1. ***Literature Review***

2.2.1 PHP: PHP is a general-purpose scripting language that is especially suited to server-side web development, in which case PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on websites or elsewhere.

The major points of HTML are given below:

* PHP stands for Hypertext Preprocessor.
* PHP is an interpreted language, i.e. there is no need for compilation.
* PHP is a server side scripting language.
* PHP is faster than other scripting language e.g. asp and jsp.

PHP example.

<!DOCTYPE**>**

**<html>**

**<body>**

**<?php**

echo "**<h2>**Hello by PHP**</h2>**";

**?>**

**</body>**

**</html>**

***2.2.2*** HTML: HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages. HTML was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

The major points of HTML are given below:

* HTML stands for Hyper Text Markup Language.
* HTML is used to create web pages.
* HTML is widely used language on the web.
* We can create static website by HTML only.

HTML Example

<!DOCTYPE**>**

**<html>**

**<body>**

**<h1>**Write Your First Heading**</h1>**

**<p>**Write Your First Paragraph.**</p>**

**</body>**

**</html>**

2.2.3 CSS: CSS is used to control the style of a web document in a simple and easy way. The major points of CSS are given below:

* CSS stands for Cascading Style Sheet.
* CSS is used to design HTML tags.
* CSS is a widely used language on the web.
* HTML, CSS and JavaScript are used for web designing. It helps the web designers to apply style on HTML tags.

Example

<!DOCTYPE**>**

**<html>**

**<head>**

**<style>**

h1{

color:white;

background-color:red;

padding:5px;

}

p{

color:blue;

}

**</style>**

**</head>**

**<body>**

**<h1>**Write Your First CSS Example**</h1>**

**<p>**This is Paragraph.**</p>**

**</body>**

**</html>**

2.2.4 jQuery: jQuery is a fast and concise JavaScript library created by John Resig in 2006. jQuery simplifies HTML document traversing, event handling, animating, and Ajax interactions for Rapid Web Development.

* jQuery is a small and lightweight JavaScript library.
* jQuery is cross-platform.
* jQuery means "write less do more".
* jQuery simplifies AJAX call and DOM manipulation.

<!DOCTYPE html>

<html>

<head>

 <title>First jQuery Example</title>

<script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">

 </script>

 <script type="text/javascript" language="javascript">

 $(document).ready(function() {

 $("p").css("background-color", "pink");

 });

 </script>

 </head>

<body>

<p>This is first paragraph.</p>

<p>This is second paragraph.</p>

<p>This is third paragraph.</p>

</body>

</html>

2.2.5 MySQL: MySQL is the most popular Open Source Relational SQL Database Management System. MySQL is one of the best RDBMS being used for developing various web-based software applications. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company.

## Chapter Summary

In this chapter, we discuss the Code of the whole project what we have been use in our project. We use PHP,HTML,Jquery,CSS,MySQL.

# 

# Chapter 3

# System Requirement

# 3 System Requirement

This chapter aims to document the systems requirements and software setup process. There are two types of requirements Software Requirements and Hardware Requirements.

* 1. ***Software & Hardware***

The most common set of requirements defined by any [operating system](https://en.wikipedia.org/wiki/Operating_system) or [software application](https://en.wikipedia.org/wiki/Software_application) is the physical computer resources, also known as [hardware](https://en.wikipedia.org/wiki/Computer_hardware), A hardware requirements list is often accompanied by a [hardware compatibility list](https://en.wikipedia.org/wiki/Hardware_compatibility_list) (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

[Software requirements](https://en.wikipedia.org/wiki/Software_requirements) deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or prerequisites are generally not included in the software installation package and need to be installed separately before the software is installed.

* + 1. **Software Requirements:**

1. XAMPP :Web Application Server

2. NetBeens :  Integrated Tools

3. MySQL : Database

4. OS : 64-bit Windows 7

* + 1. **Hardware Requirements:**

1. Pentium 4 or AMD or Celeron Processor or above

2. RAM 512 MB or above

* 1. ***Software setup process***

**Installation** (or **setup**) of a [computer program](https://en.wikipedia.org/wiki/Computer_program) (including [device drivers](https://en.wikipedia.org/wiki/Device_driver) and [plugins](https://en.wikipedia.org/wiki/Plug-in_(computing))), is the act of making the program ready for [execution](https://en.wikipedia.org/wiki/Execution_(computing)). Because the process varies for each program and each computer, programs (including [operating systems](https://en.wikipedia.org/wiki/Operating_system)) often come with an *installer*, a specialized program responsible for doing whatever is needed for their installation. Installation may be part of a larger [software deployment](https://en.wikipedia.org/wiki/Software_deployment) process.

* + 1. **XAMPP server setup:**

1. **Open the XAMPP website.** Go to  <https://www.apachefriends.org/index.html> in computer's web browser.

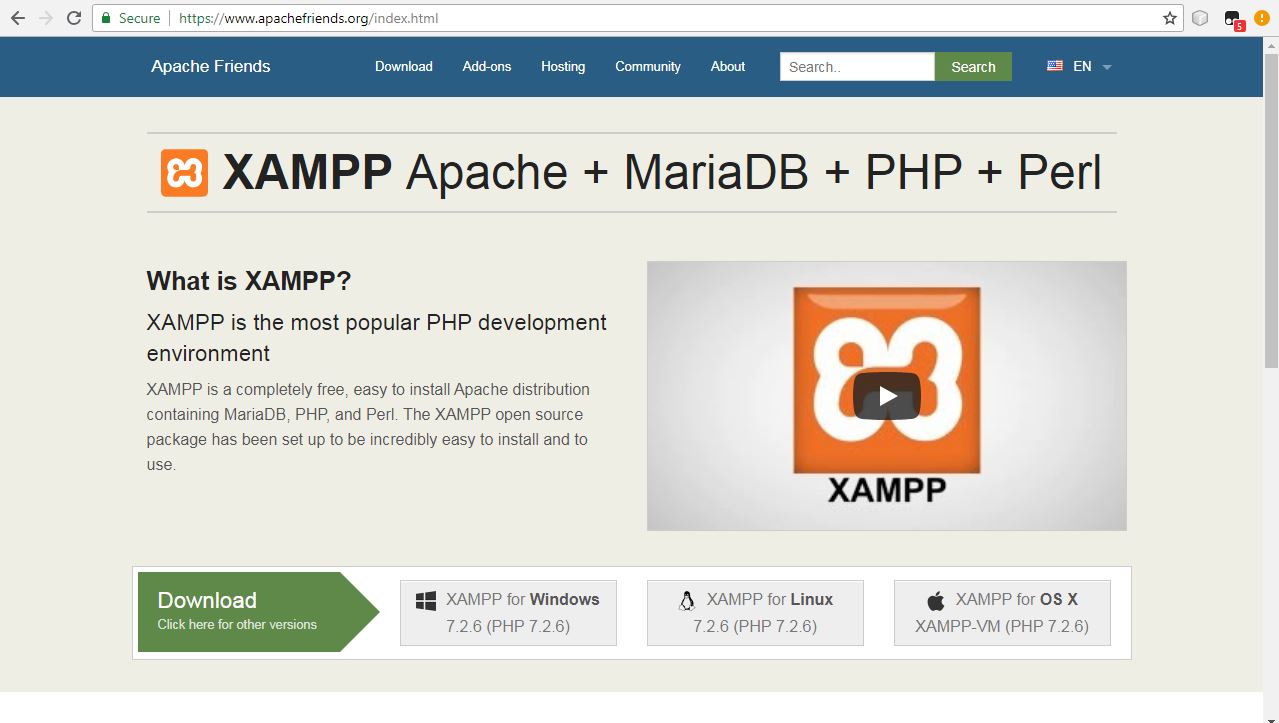


Figure 3.: Xampp Server Download

1. **Click XAMPP for Windows.** It's a grey button near the bottom of the page.

Depending on your browser, you may first have to select a save location or verify the download.

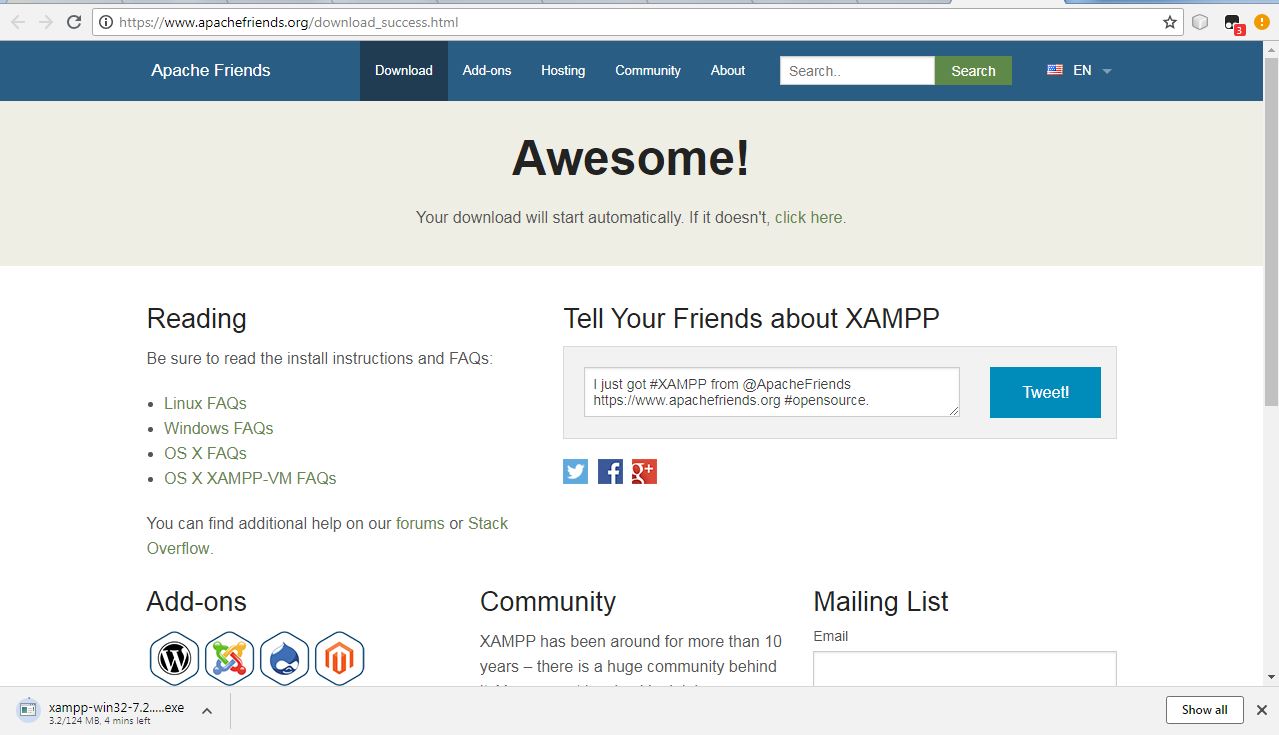


Figure 3.2: Xampp Server Downloading

1. **Double-click the downloaded file.** This file should be named something like **xampp-win32-7.2.4-0-VC15-installer**, and you'll find it in the default downloads location (e.g., the "Downloads" folder or the desktop).

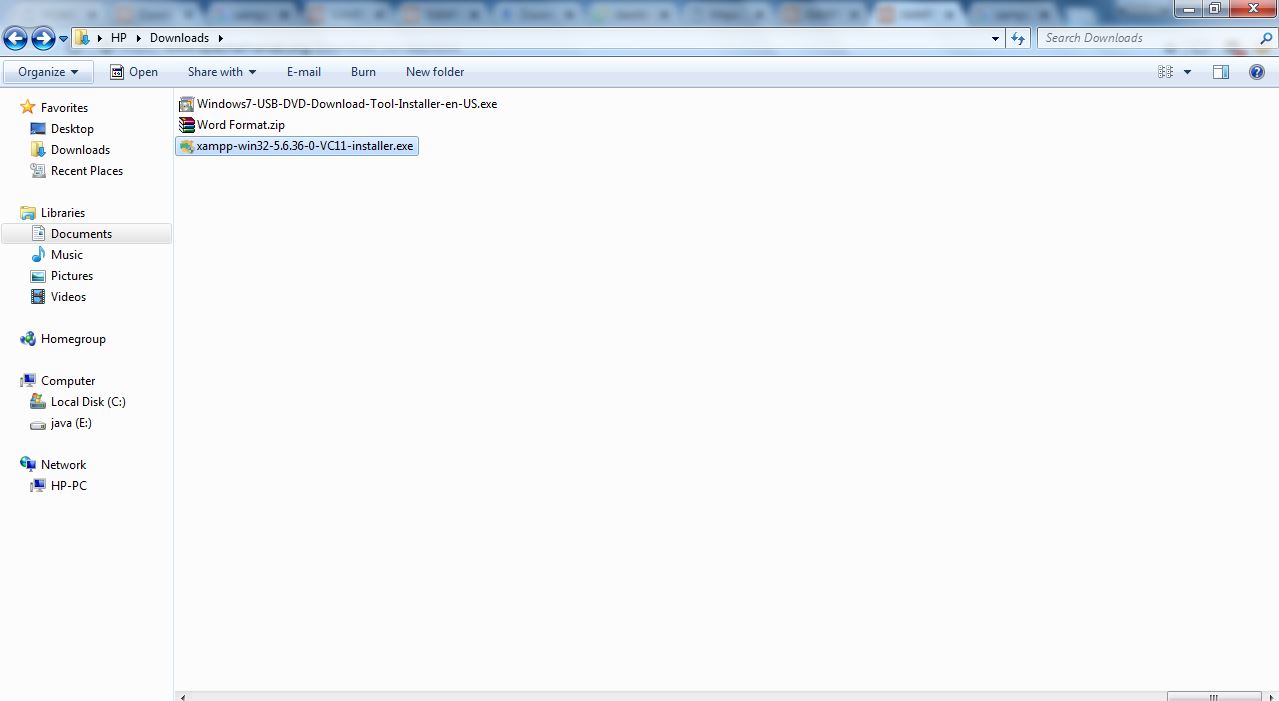
****

Figure 3.3: Xampp Server Downloading File

**4. Click Yes when prompted** This will open the XAMPP setup window. You may have to click **OK** on a warning if you have User Account Control (UAC)activated on your computer.

**5. Click Next.** It's at the bottom of the setup window.

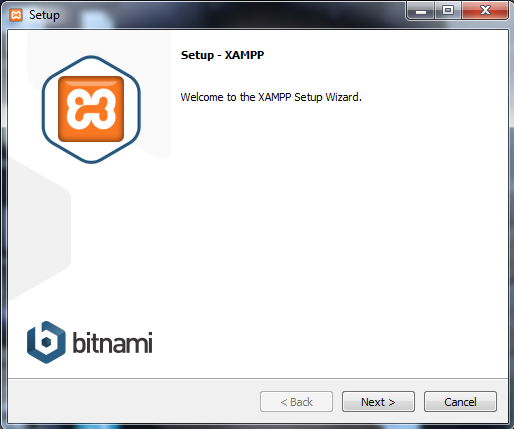


Figure 3.4:Xampp Server Setup Process

**6. Select aspects of XAMPP to install.** Review the list of XAMPP attributes on the left side of the window; if you see an attribute that you don't want to install as part of XAMPP, uncheck its box.

* By default, all attributes are included in your XAMPP installation.

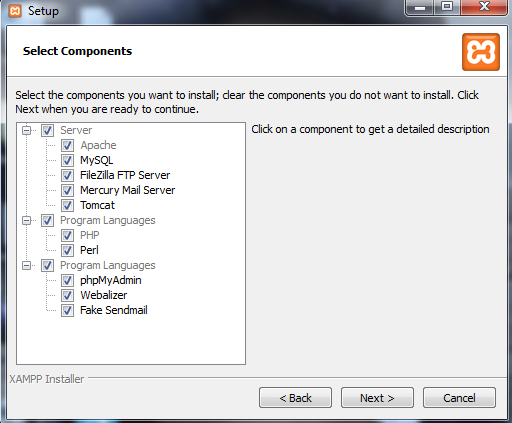


Figure 3.5:Xampp Server Setup Process

**7. Select an installation location.** Click the folder-shaped icon to the right of the current installation destination, then click a folder on your computer.

If you have the UAC activated on your computer, avoid installing XAMPP in your hard drive's folder (e.g., **OS (C:)**).

You can select a folder (e.g., **Desktop**) and then click **Make New Folder** to create a new folder and select it as the installation destination.

1. **Begin installing XAMPP.** Click **Next** at the bottom of the window to do so. XAMPP will begin installing its files into the folder that you selected.
2. **Click Finish when prompted.** It's at the bottom of the XAMPP window. Doing so will close the window and open the XAMPP Control Panel, which is where you'll access your servers.

**Click Next.** It's at the bottom of the window.

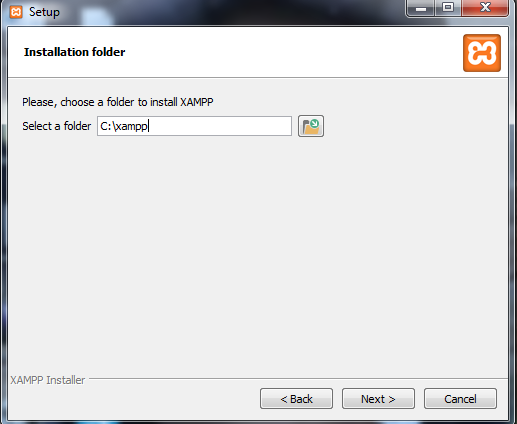


Figure 3.6:Xampp Server Setup Process

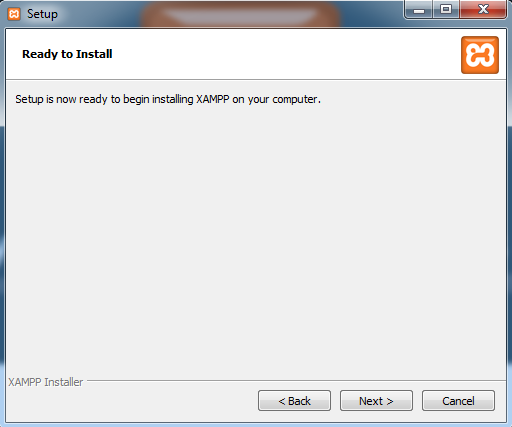


Figure 3.7: Xampp Server Setup Process

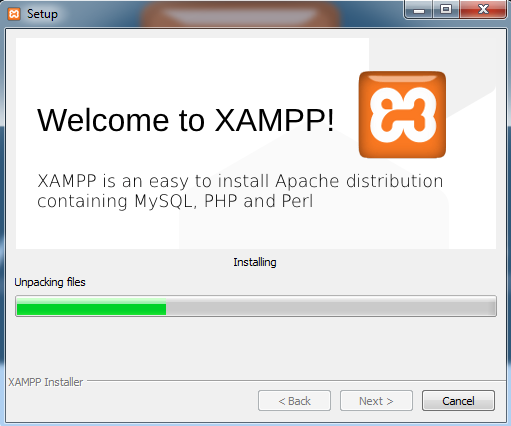


Figure 3.8: Xampp Server Setup Process

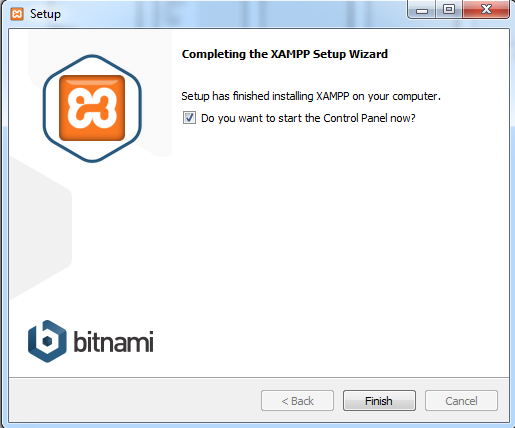


Figure 3.8: Xampp Server Setup Process

**8. Select a language.** Check the box next to the American flag for

English, or check the box next to the German flag for German.

**9. Click Save.** Doing so opens the main Control Panel page.



Figure 3.7: Xampp Server Setup Process(select a language)

10. Start XAMPP from its installation point. If you need to open the XAMPP Control Panel in the future, you can do so by opening the folder in which you installed XAMPP, right-clicking the orange-and-white xampp-control icon, clicking Run as administrator, and clicking Yes when prompted.

* When you do this, you'll see red X marks to the left of each server type (e.g., "Apache"). Clicking one of these will prompt you to click Yes if you want to install the server type's software on your computer.
* Counter intuitively, double-clicking the xampp\_start icon doesn't start XAMPP.

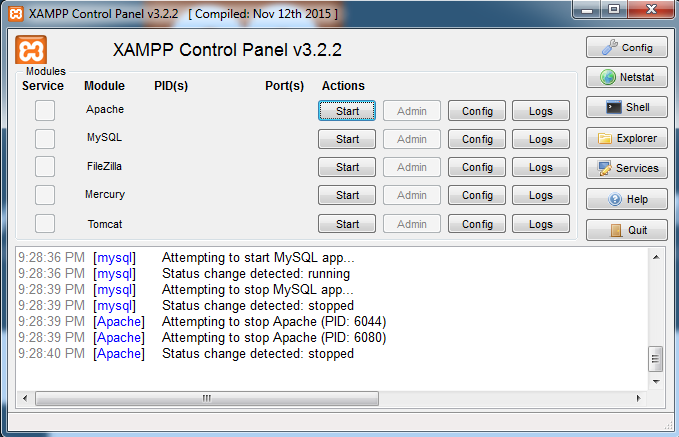


Figure 3.8: Xampp Control Panel

**11. Resolve issues with Apache refusing to run.** On some Windows 10 computers, Apache won't run due to a "blocked port". This can happen for a couple of reasons,

but there's a relatively easy fix:[[1]](https://www.wikihow.com/Install-XAMPP-for-Windows#_note-1)

* + Click **Config** to the right of the "Apache" heading.
  + Click **Apache (httpd.conf)** in the menu.
  + Scroll down to the "Listen 80" section (you can press Ctrl+F and type in listen 80 to find it faster).
  + Replace 80 with any open port (e.g., 81 or 9080).
  + Press Ctrl+S to save the changes, then exit the text editor.
  + Restart XAMPP by clicking **Quit** and then re-opening it in administrator mode from its folder.
    1. CodeIgniter server setup:

CodeIgniter is installed in five steps:

1. **Open the** CodeIgniter **website.** Go to  https://codeigniter.com/download in computer's web browser and download it.

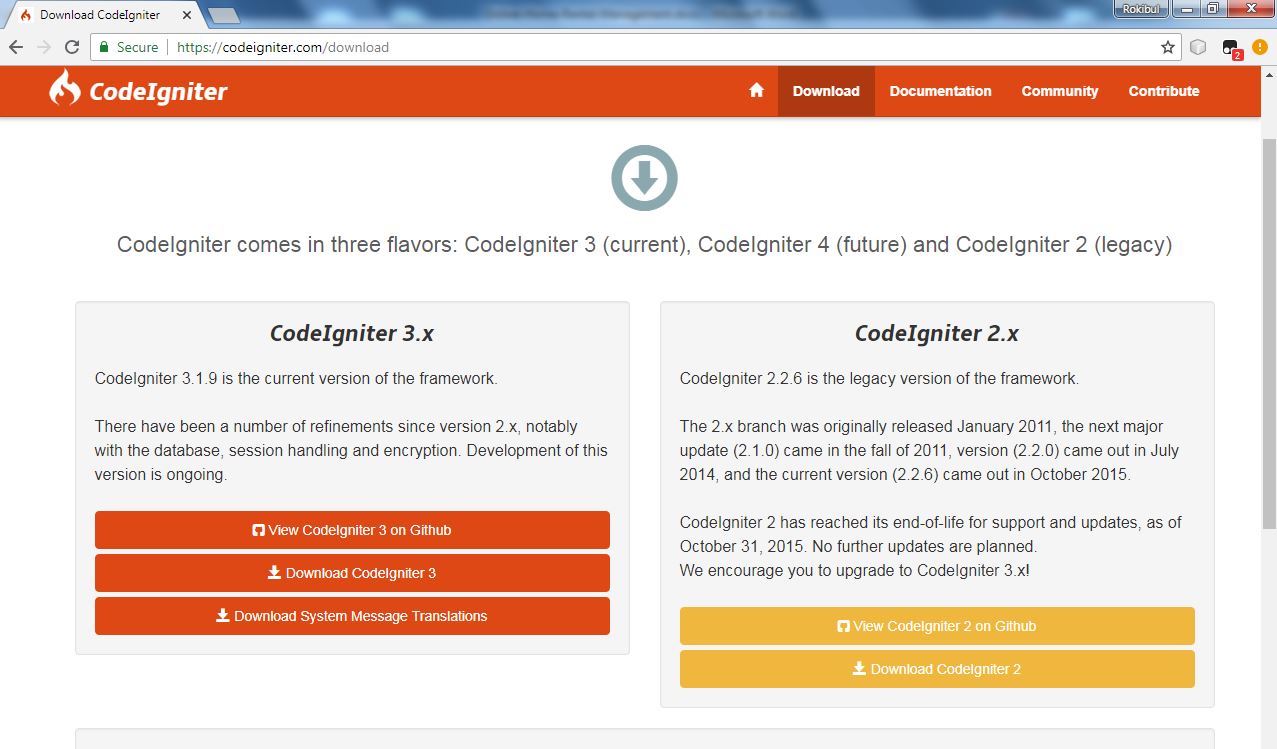


Figure 3.9: CodeIgniter Download

1. Unzip the package.

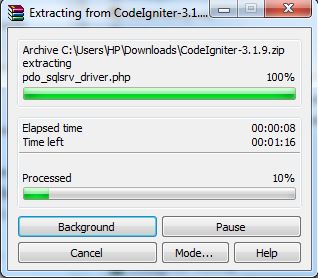


Figure 3.10: CodeIgniter Download

1. Upload the CodeIgniter folders and files to your server. Normally the *index.php* file will be at your root.

Copy all CodeIgniter folders all files and paste it C:\xampp\htdocs\home\_rent.

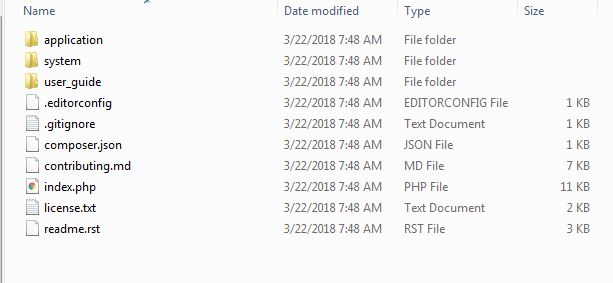


Figure 3.11: CodeIgniter Folder

1. Open the *application/config/config.php* file with a text editor and set your base URL. If you intend to use encryption or sessions, set your encryption key.

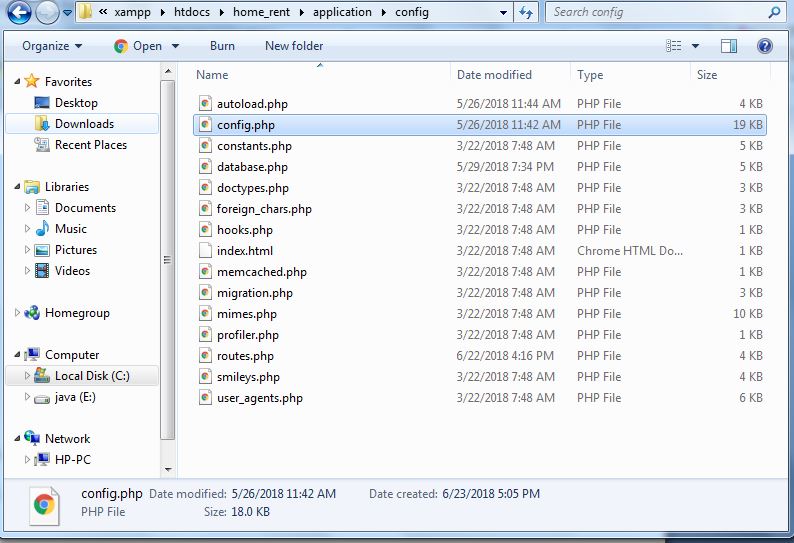


Figure 3.12: Inside Application config.php File

1. If you intend to use a database, open the *application/config/database.php* file with a text editor and set your database settings.

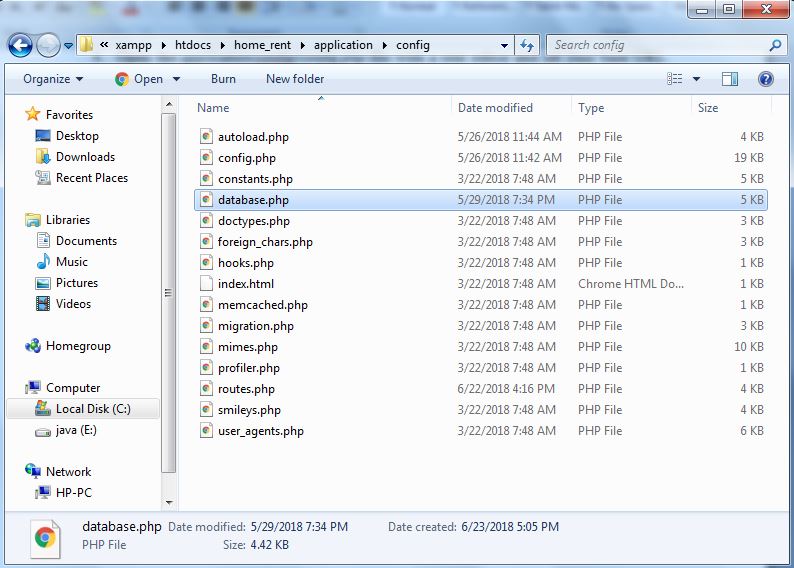


Figure 3.11: Inside Application config.php File

3.3 Chapter Summary:

In this chapter, we discuss about the system requirements, what kind of software and hardware are require in this project that have been discuss. There have also the software setup process.

# Chapter 4

# Project Details

# Project Details

Functional Requirements are the fundamental / basic requirements of a system. To capture functional requirements we have used the most widely used Database details and the system flowchart of the system.

**4.1 Database Details**

Database is the base of system in project, Here we add all the database of the project. we use MySQL for the database. because it is user friendly.

Table-Name: Users

Primary-key: user\_id

Description: This table Users is used to registration and login into all user to access the Website.

Table 4.: Users Table

|  |  |  |
| --- | --- | --- |
| Name | Type | Comments |
| user\_id | Int(11) |  |
| userName | Varchar(50) |  |
| address | Varchar(50) |  |
| email | Varchar(50) |  |
| city | Varchar(50) |  |
| division | Varchar(50) |  |
| zipcode | Varchar(50) |  |
| contactNo | Varchar(50) |  |
| userRole | Tinyint(1) | Admin=1,home\_seeker=2,  home\_owner=3. |
| password | Varchar(50) |  |
| country | Varchar(50) |  |

Table-Name: Contact\_info

Primary-key: info\_id

Description: This table Contact\_info ,here user can send there feedback to the admin from the website.

Table 4.: Contact\_info Table

|  |  |  |
| --- | --- | --- |
| Name | Type | Null |
| info\_id | Int(10) | No |
| username | Varchar(50) | No |
| email | Varchar(50) | No |
| subject | Varchar(50) | No |
| message | Varchar(50) | No |

Table-Name: property

Primary-key: property\_id

Description: This table property, here have all the property details,user can show that details and home owner add the their property here and need to admin approval for post the property to the property table.

Table 4.: Property Table

|  |  |  |
| --- | --- | --- |
| Name | Type | Null |
| property\_id | Int(11) | No |
| property\_location | Varchar(50) | No |
| Num\_of\_room | Int(50) | No |
| Num\_of\_bath | Int(50) | No |
| Property\_size | Varchar(50) | No |
| Property\_image | Varchar(500) | No |
| Property\_description | Varchar(255) | Yes |
| Property\_price | Double(255) | No |
| Category\_id | Int(11) | No |
| User\_id | Int(11)` | No |
| Status | Tinyint(1) | No |
| Date\_added | Timestamp | No |
| Country | Varchar(50) | No |
| city | Varchar(50) | No |
| Phone\_number | Varchar(50) | No |

Table-Name: tbl\_category

Primary-key: category\_id

Description: This table tbl\_category, have all the category of the property, only admin can add/update/delete the information of this table.

Table 4.: Category Table

|  |  |  |
| --- | --- | --- |
| Name | Type | Null |
| Category\_id | Int(11) | No |
| Category\_name | Varchar(100) | No |

Table-Name: Order Details

Primary-key: order\_id

Description: This table order details, have the information of those home-seeker who order or booked the selected home.

Table 4.: Booking Table

|  |  |  |
| --- | --- | --- |
| Name | Type | Null |
| Order\_id | Int(11) | No |
| homeSeeker\_id | Int(11) | No |
| Property\_id | Int(11) | No |
| Date\_of\_booked | Timestamp | No |

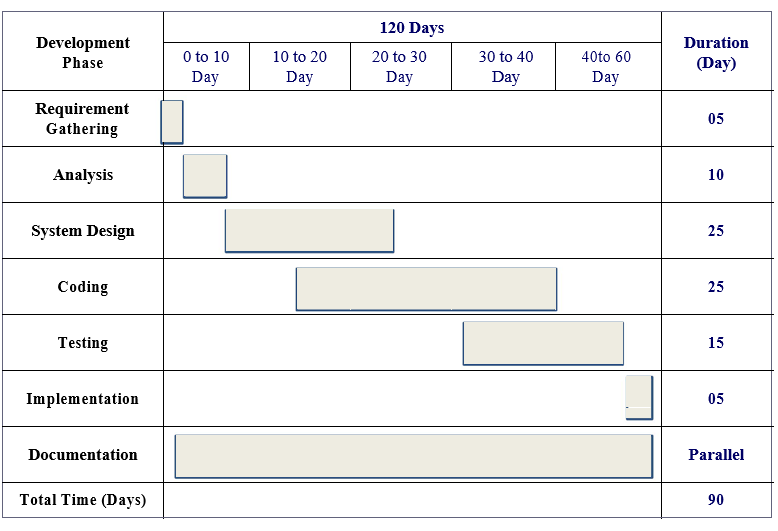
**4.2 System Design**

**Systems design** is the process of defining the [architecture](https://en.wikipedia.org/wiki/Systems_architecture), modules, interfaces, and [data](https://en.wikipedia.org/wiki/Data) for a [system](https://en.wikipedia.org/wiki/System) to satisfy specified [requirements](https://en.wikipedia.org/wiki/Requirement). Systems design could be seen as the application of [systems theory](https://en.wikipedia.org/wiki/Systems_theory) to [product development](https://en.wikipedia.org/wiki/Product_development). There is some overlap with the disciplines of [systems analysis](https://en.wikipedia.org/wiki/Systems_analysis), [systems architecture](https://en.wikipedia.org/wiki/Systems_architecture) and [systems engineering](https://en.wikipedia.org/wiki/Systems_engineering).

**4.2.1 Timeline Chart**

**This chart** describe the project timeline overview. Here we divided our work within deadline by day.

Figure 4.: Timeline Chart



* + 1. **System Flowchart**

All function worked as the following chart. For using this system registration first. If you registered then you can login. If you are not valid users then you go back the login page. If valid then check what type of users you are? If you are admin then manage all function. If you home owner then you upload your home fore rent. If you Home seeker then just order home. Finally logout.



Figure 4.: System FlowChart

* + 1. **E-R Diagram**

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes.

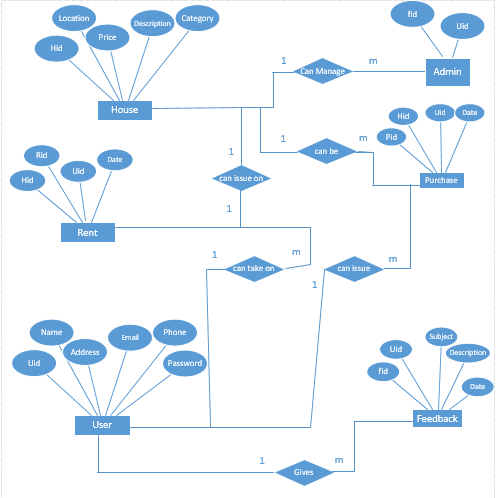
****

Figure 4.: E-R Diagram

* + 1. **Use Case Diagram**

A Usecase is a summary of scenarios for a single task or goal. An actor is who or what initiates the events involved in that task. Actors are simple roles that people or objects play. The picture below showing the Usecase diagram



Figure 4.: Use Case Diagram For Admin



Figure 4.: Use Case Diagram For Home Owner



Figure 4.: Use Case Diagram For Home Seeker

* + 1. Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system.Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc

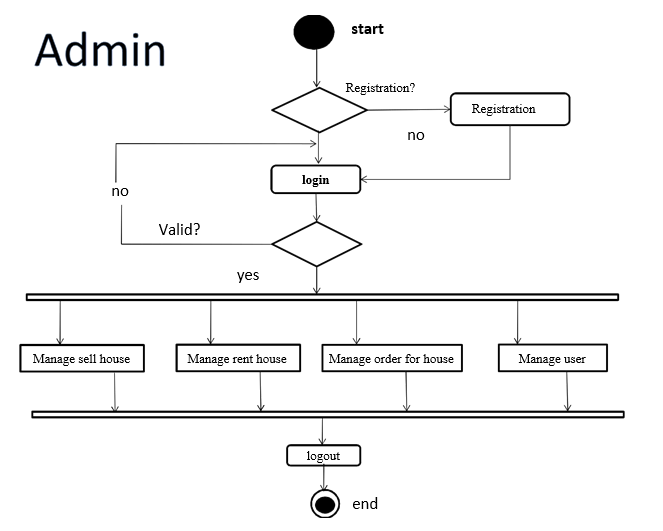


Figure 4.:Activity Diagram For Admin

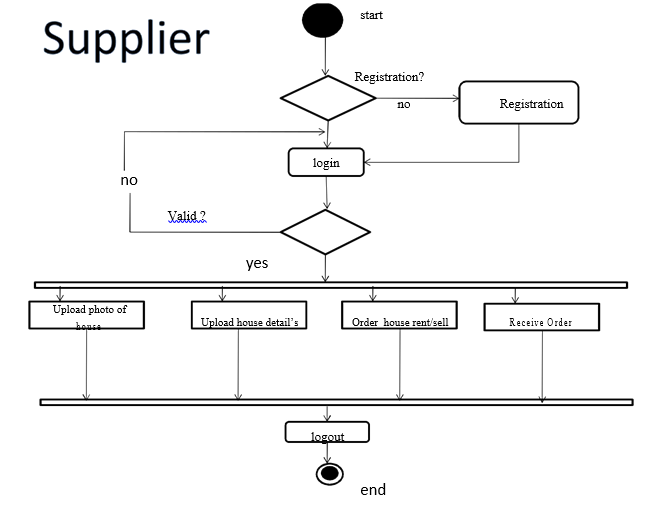


Figure 4.: Activity Diagram Supplier/Home Owner

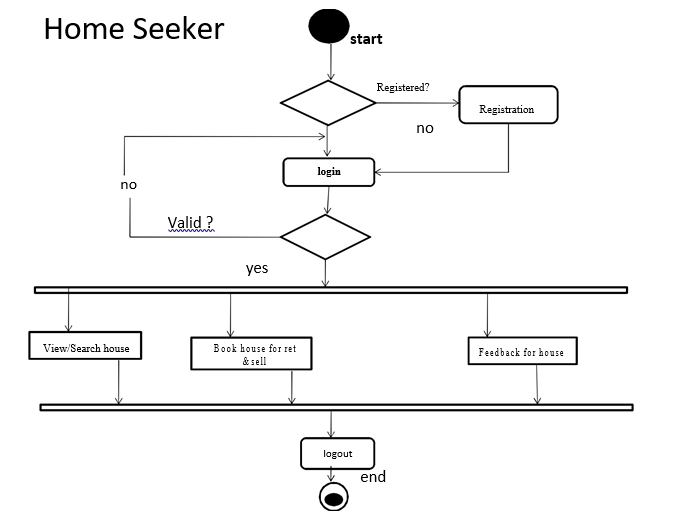
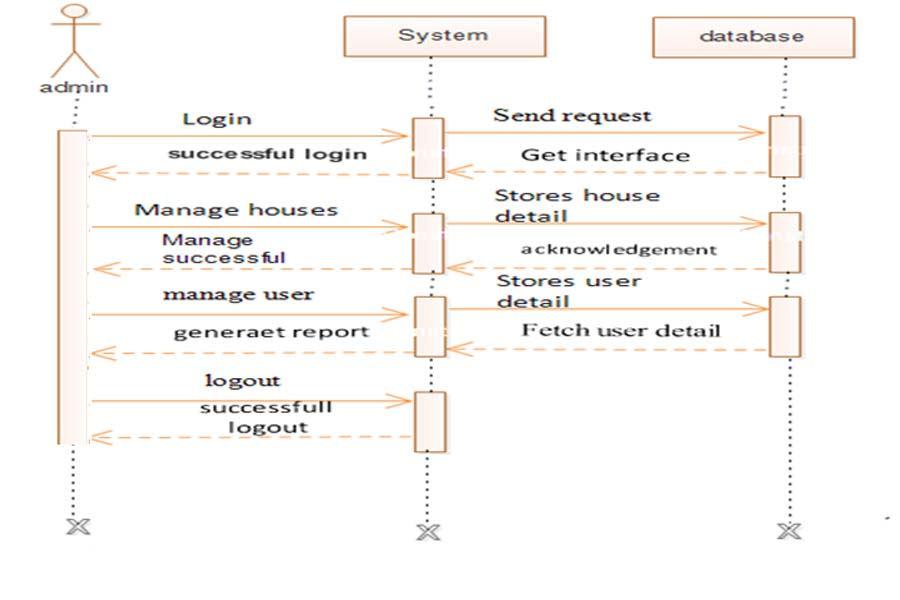


Figure 4.: Activity Diagram For Home Seeker

* + 1. **Sequence Diagram**

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

A sequence diagram shows, as parallel vertical lines , different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

****

**Figure 4.10: Sequence Diagram For Admin**

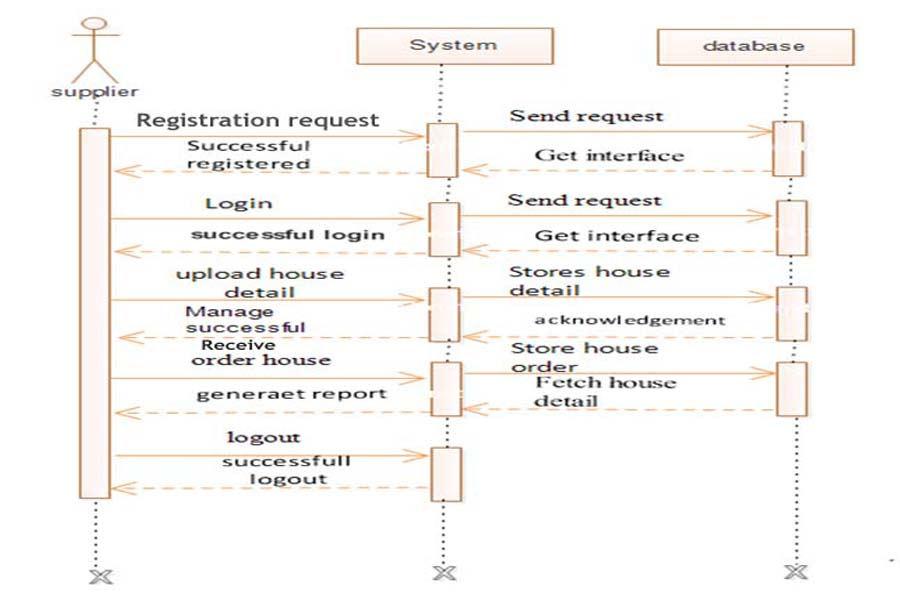


Figure 4.: Sequence Diagram For Supplier

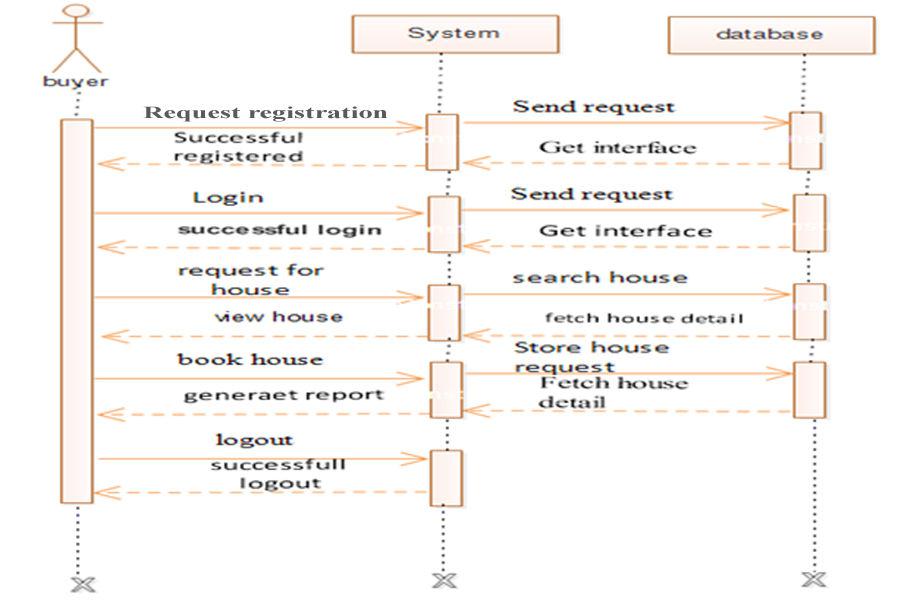
****

Figure 4.: Sequence Diagram Buyer

* + 1. **Collaboration Diagram**

A collaboration diagram is a type of visual presentation that shows how various software objects interact with each other within an overall IT architecture and how users can benefit from this collaboration. A collaboration diagram often comes in the form of a visual chart that resembles a flow chart. It can show, at a glance, how a single piece of software complements other parts of a greater system.

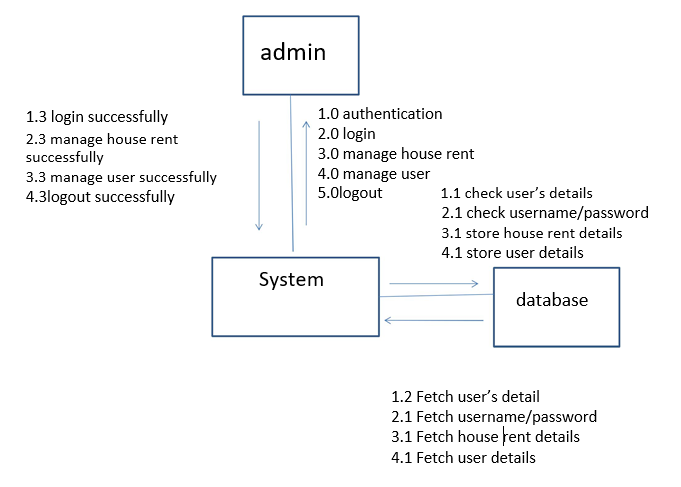


Figure 4.:Collaboration Diagram For Admin

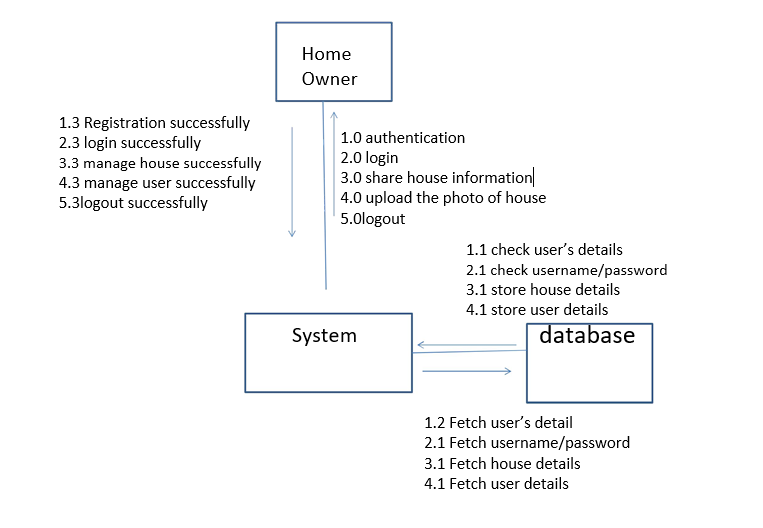
****

Figure 4.: Collaboration Diagram For Home Owner

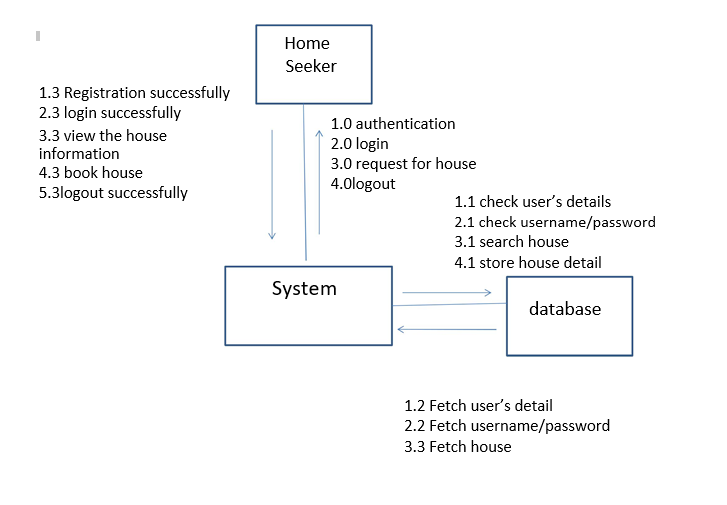
****

Figure 4.: Collaboration Diagram For Home Seeker

* 1. **Input Project Design**

**Page-Name:** User Interface Home

**Description:** In the home page,It is the user interface of the system,here home seeker can search their home with the city and category of their own choice,And home seeker have tobe login first to submit their property.

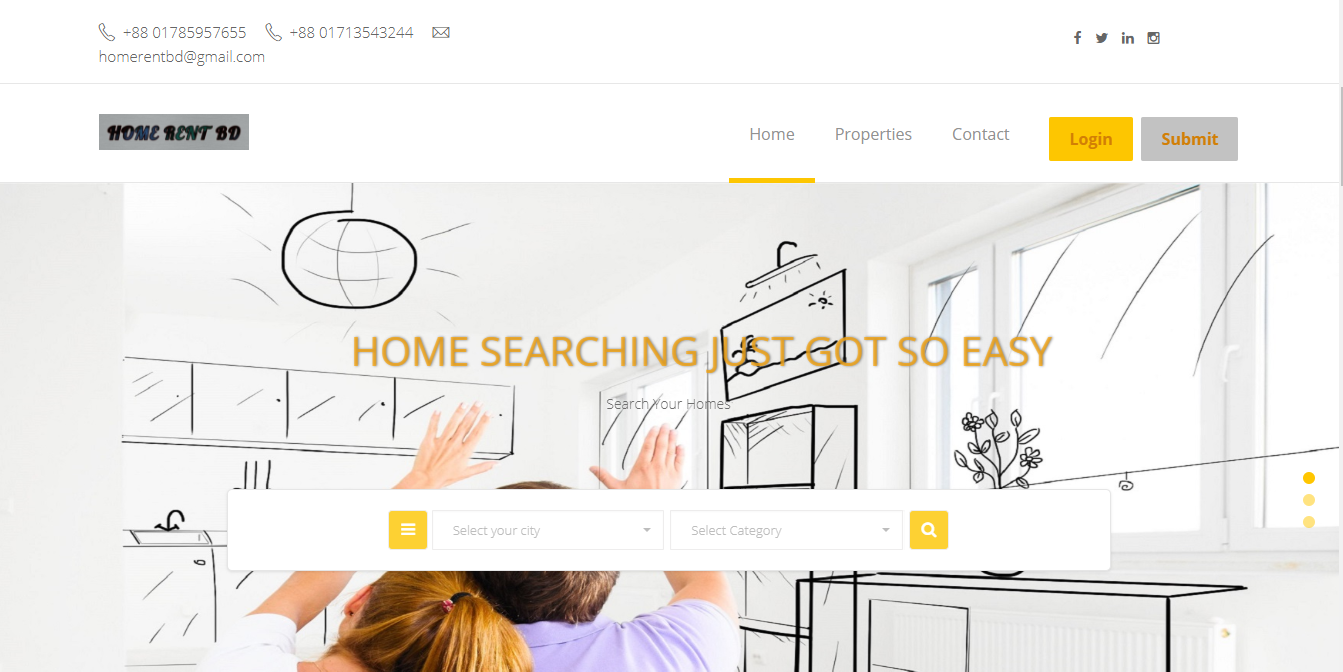


Figure 4..: User Interface Home

**Page-Name:** User Interface Properties

**Description:** In this page, home seeker can see the property what have been post by admin or the home owner. When user press the view button then he can see the information of the selected property. Home can only edit what file he been posted.

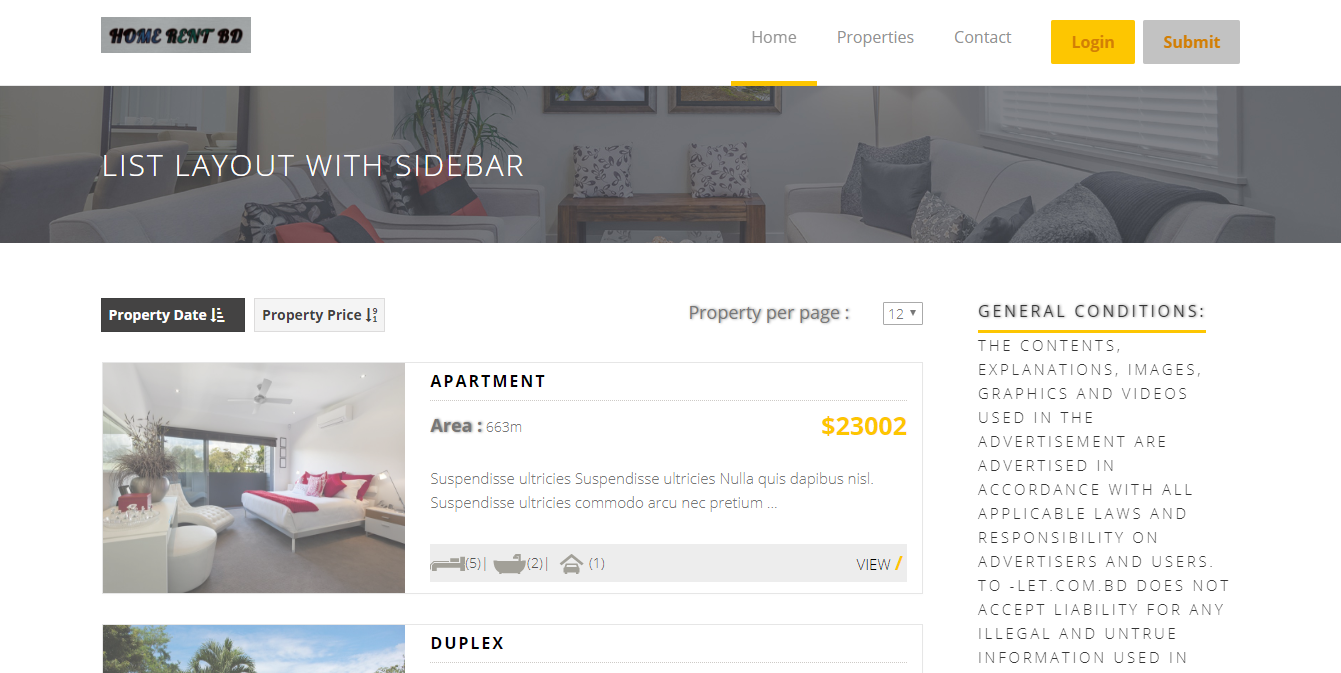


Figure 4..: User Interface Properties

**Page-Name:** User Interface Contact

**Description:** In this page user can contact with the admin with the phone-number or the email have been provided.

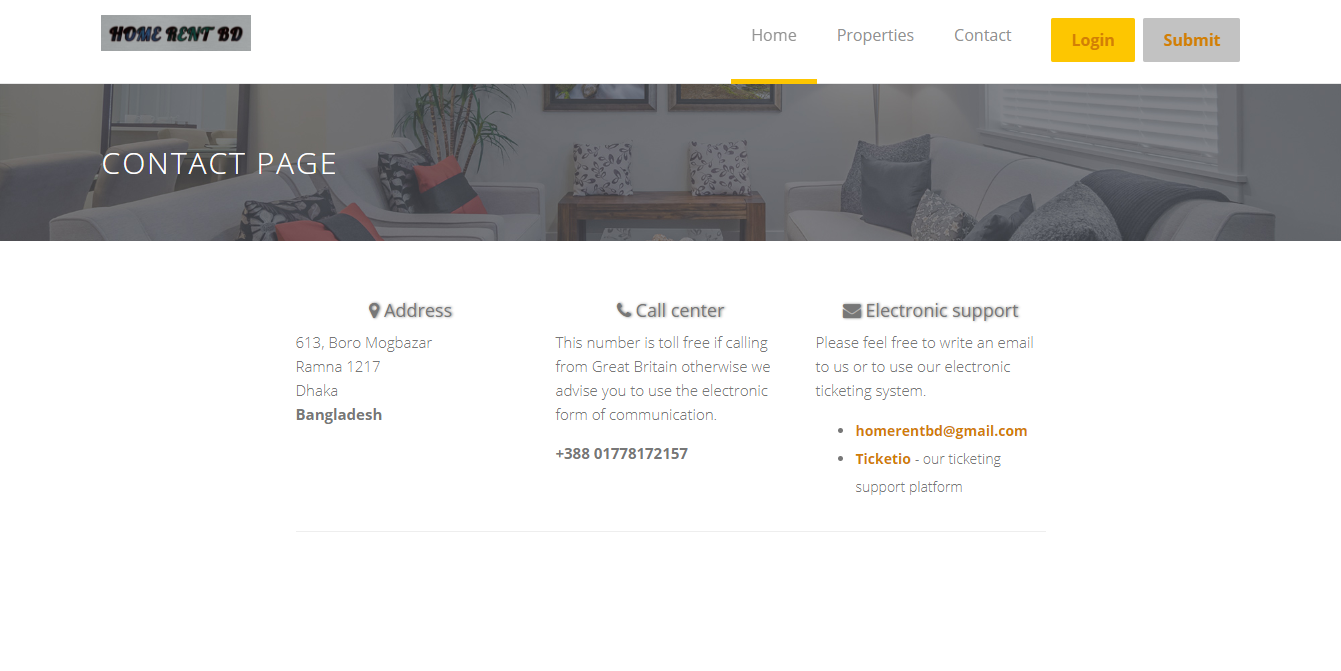
****

Figure 4..: User Interface Contact

**Page-Name:** Contact Form

**Description:** It is the contact page of the user interface, user can directly send message to the admin. And this message show in the admin panel.

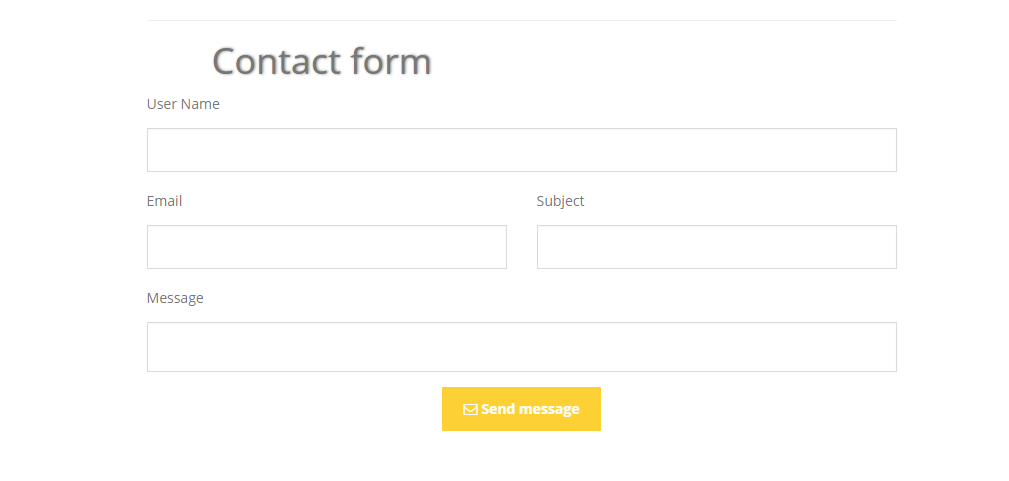
****

Figure 4..:Contact Form

**Page-Name:** Register New Account

**Description:** When A user want to submit their property, they must have to register first. Register with their name, email, phone number, password and have to retype password and it must be match. And also need to fill-up captcha for confirmation its not robot. Then he is be a home owner.

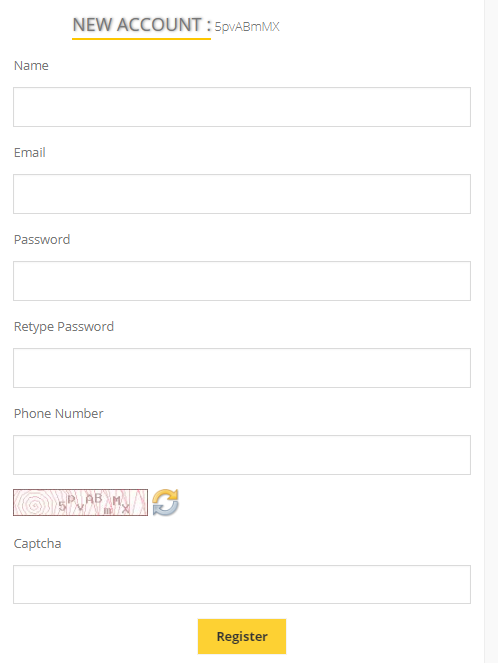
****

Figure 4..:Register New Account

**Page-Name:** User Interface Login

**Description:** In this section a home seeker login with their email and password. Then he submit his property and also edit or delete his property.

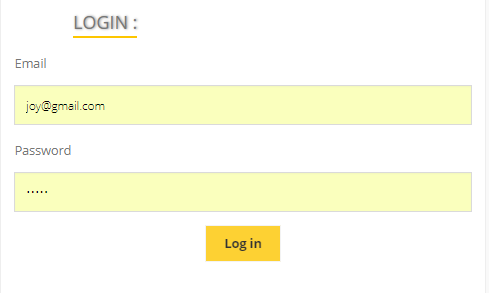


Figure 4..:User Interface Login

**Page-Name:** User Interface Submit Property

**Description:** Home owner submit their property here, and directly go to the admin panel for admin approval.

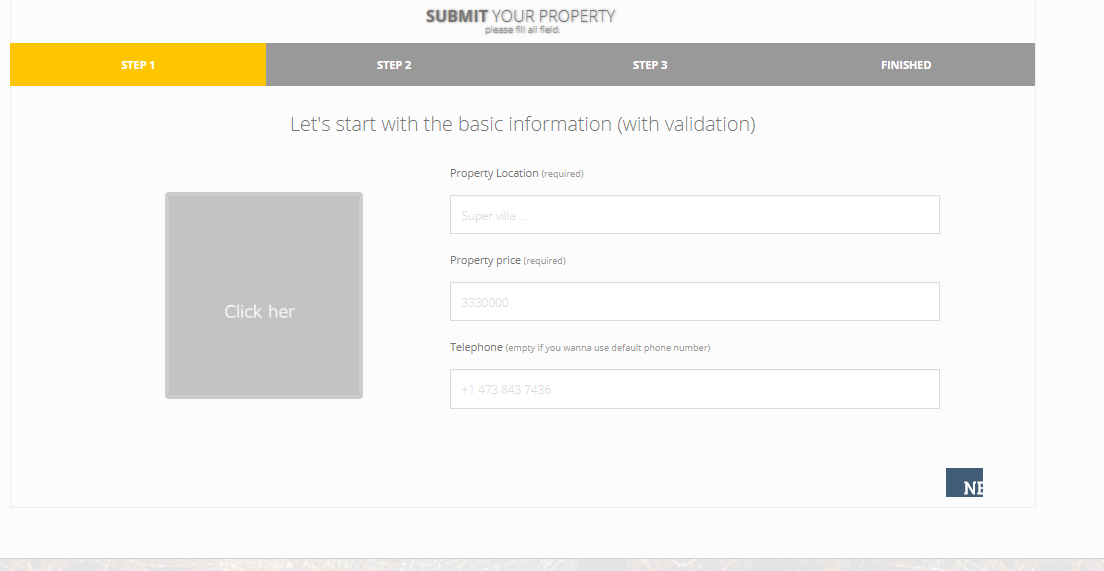
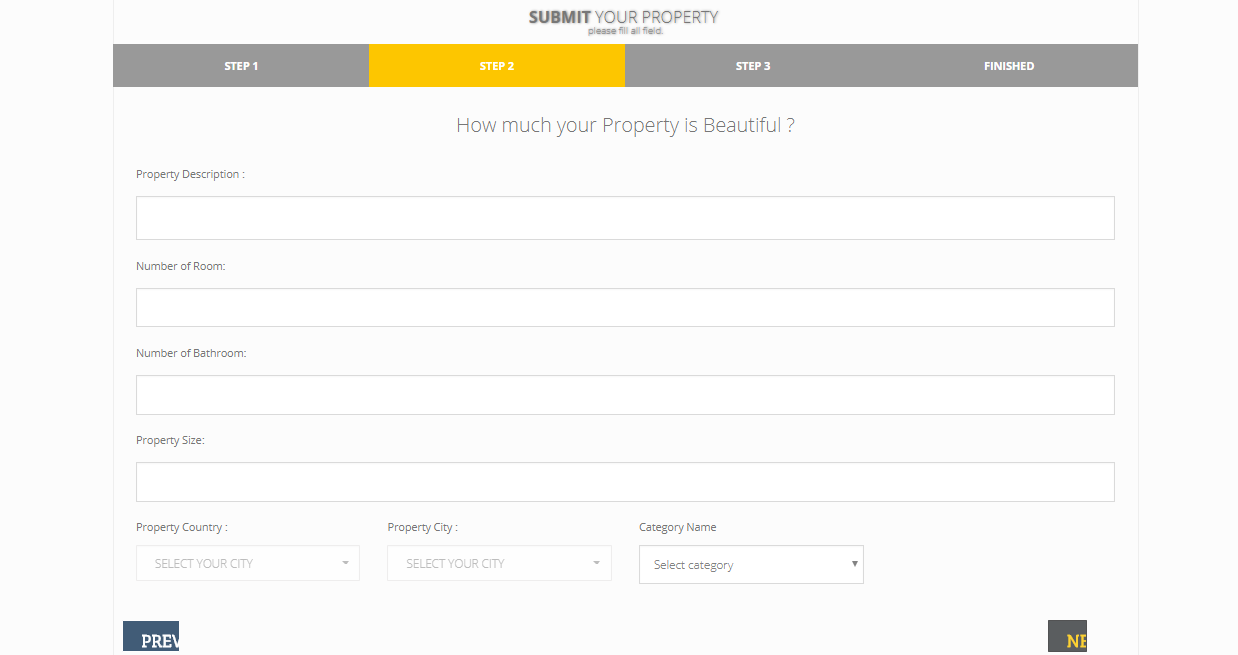
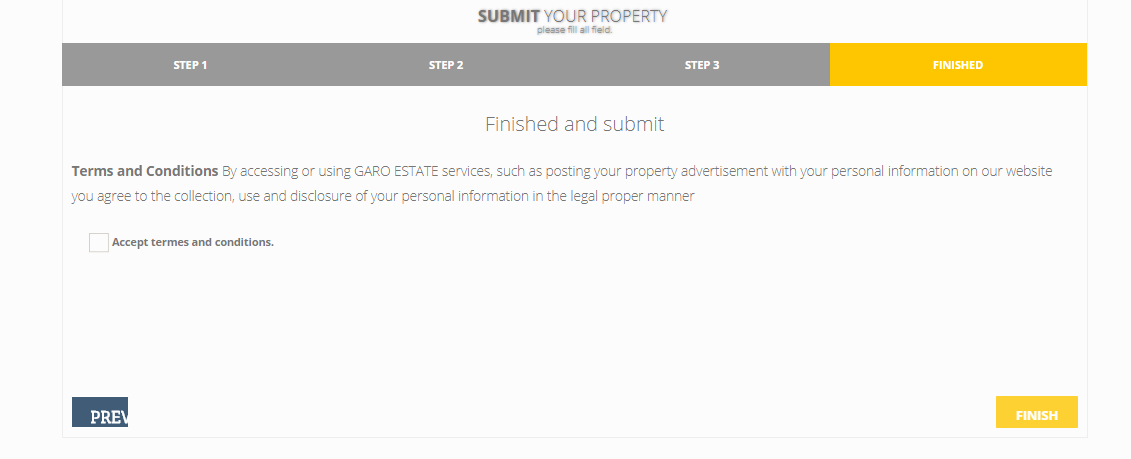
****

Figure 4..: User Interface Submit Property

****

****

**Page-Name:** Admin Login

**Description:** It’s the admin login page, here admin login with their valid email and password. If not match he can not be enter in the dashboard.

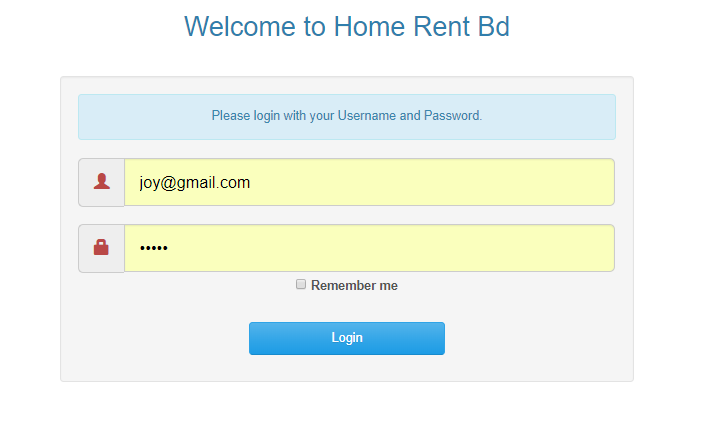
****

Figure 4..: Admin Panel Login

**Page-Name:** User Interface Login

**Description:** It’s the dashboard of the admin panel. Here have all the option what can do by admin.

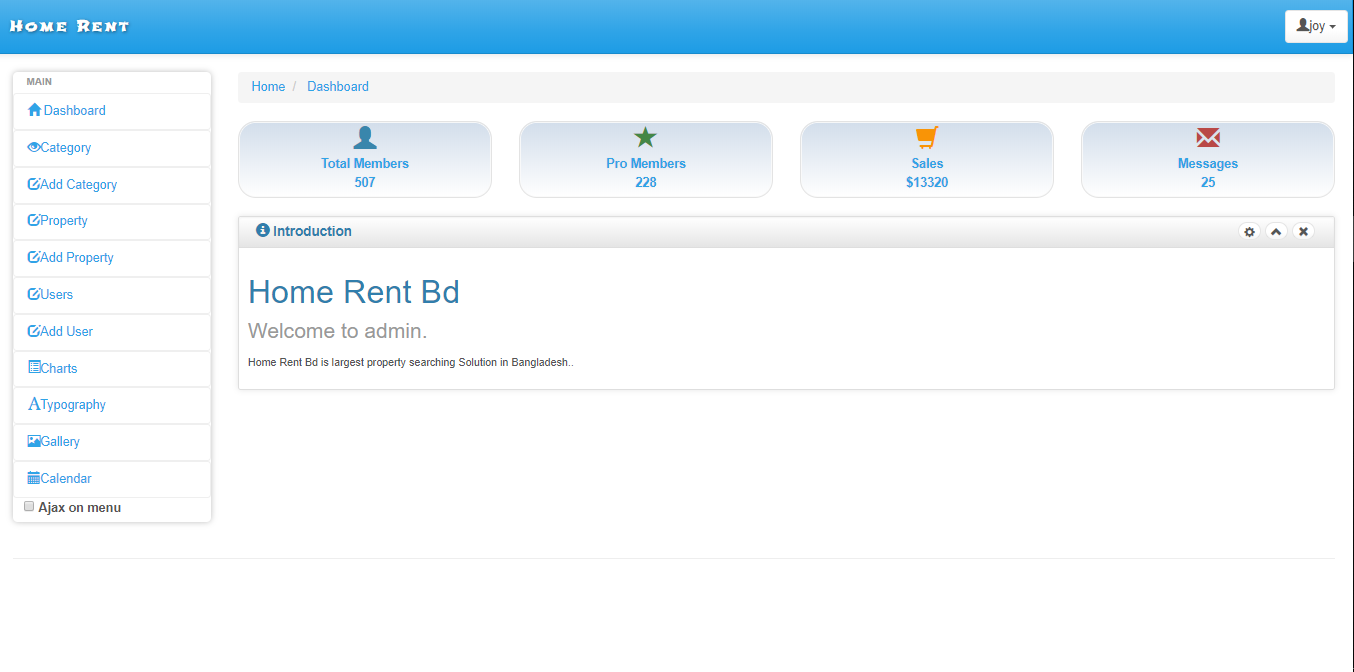
****

Figure 4..: Admin Panel Dashboard

**Page-Name:** Admin Panel Property Table

**Description:** Here Show the property list what have been post by admin and home owner. Here also edit and delete action. And the admin approval option admin have to be approved the property. And when a property rent rent successfully then admin disapproved those property from this table. And then user can see that property in properties page any more.

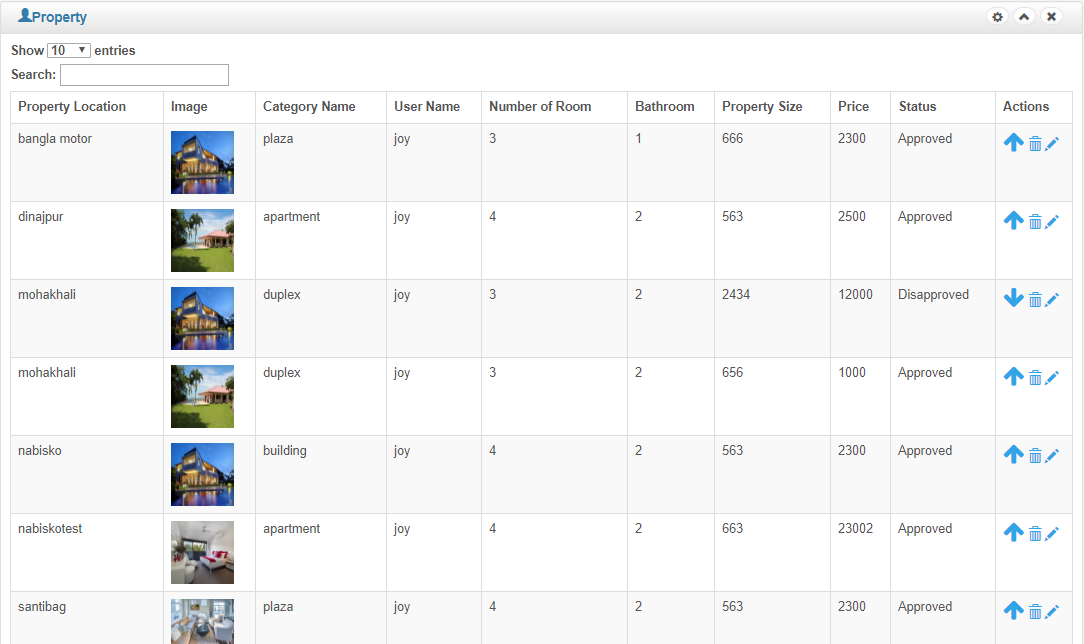
****

Figure 4..: Admin Panel Property Table

**Page-Name:** Admin Panel Category Table

**Description:** It’s the category table of the admin panel, here show the category of the home, admin can edit and delete the category here.

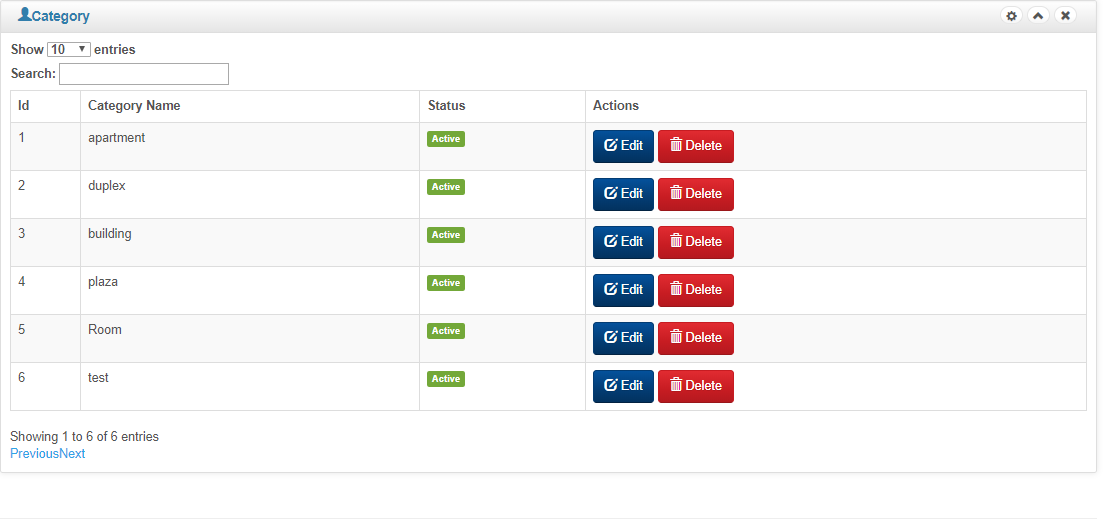
****

Figure 4..: Admin Panel Category Table

**Page-Name:** Admin Panel User Table

**Description:** In this table , user information showed here, admin can edit and delete user. Admin can not able to delete Him.

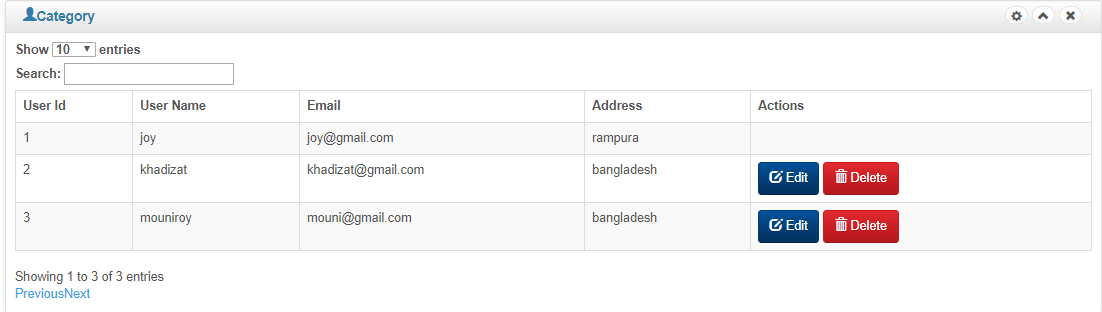
****

Figure 4.: Admin Panel User Table

* 1. **Chapter Summary**

In This chapter, we discuss about project details. Where we have database details, system design(In System design here different kinds of flowchart ,diagram etc ),Developed project design.

**Chapter 5**

**Conclusion & Future Work**

**5 Conclusion & Future Work**

In this following chapter, we sum our work in brief and conclude the paper. We also represent a future works which we could not able to do because of some contradictions we face during our research. We will work more on this and we have mentioned in the future work what we tend to next.

In this paper, we analyzed how the system can be made more dynamic. As the systems’ main users are the general people so it has to be more user-friendly and dynamic. We made comparison with SQL search by implementing the search option in this. But, then we had to do that manually. In that case, general people would not be able to use that. Then we implemented CSP in the search option to make this dynamic. Finally, to give special concern on the case of renting house by bachelors, we implemented IOT to make that selection more precise and efficient. This work, to our best knowledge, is implementation of CSP algorithm in the search option of the system with a combination of dynamic mail alert and chat server.

## Future Work

In Future we want to done order/booking system in our project, we have to plan to add a new user manager. In our Project, we compared our work with SQL. But, there so many other algorithms which can be used in search field as well. The existing system will never reveal which algorithm they are using. Therefore, our next work is to compare it with the other algorithms and check for is it more dynamic than CSP or not.

# References

1. www.w3schools.com[online].
2. Tutorials Point. (2016). SQL RDBMS Databases. [online] Available at: https://www.tutorialspoint.com/sql/sql-databases.htm [Accessed 28 Nov. 2016].R. Harper, *Inside the smart home,* Springer and Business Media, 2006.
3. Tutorials Point. (2016). What is AJAX?. [online] Available at: https://www.tutorialspoint.com/ajax/what\_is\_ajax.htm [Accessed 28 Nov. 2016]..

[4] Gommans, H., Njiru, G. and Owange, A. (2014). Rental House Management System. International Journal of Scientific and Research Publications, [online] 4(11). Available at: http://s3.amazonaws.com/academia.edu.documents/35728176/ijsrpp35101.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1480359917&Signature= e0UNdGLLuatBwTFr%2Bh16SiHRi50%3D&response-contentdisposition=inline%3B%20filename%3DRental\_House\_Management\_System.pdf [Accessed 28 Nov. 2016].

[5] Tutorials Point. (2016). SQL RDBMS Databases. [online] Available at: https://www.tutorialspoint.com/sql/sql-databases.htm [Accessed 28 Nov. 2016].

[6] Bacchus, F. (n.d.). A Uniform View of Backtracking. [online] Tortonto, Ontario: University of Toronto. Available at: http://citeseerx.ist.psu.edu/viewdoc/versions?doi=10.1.1.65.1771 [Accessed 29 Nov. 2016].

[7]Joubert, N. (2012). CSP.JS. [online] GitHub. Available at: https://github.com/njoubert/csp.js/blob/master/README.md [Accessed 28 Nov. 2016].

[8] Rossi, F., Beek, P. and Walsh, T. (2006). Handbook of Constraint Programming. [online] Elsevier.Availableat: http://www.informatik.uniulm.de/pm/fileadmin/pm/home/fruehwirth/Papers/FAI.pdf [Accessed 28 Nov. 2016].