A Project Report On

**Make Deal**

Submitted impartial fulfillment of the requirement for the award of the degree

MASTER OF COMPUTER APPLICATIONS (INTEGRATED)

from

Gujarat Technological University



Academic Year 2020 – 21

**Yash Dhameliya (165220686009)**

**Krunal Bhanderi (165220686004)**

|  |
| --- |
| **Internal Guide** |
| Dr. Sunil Bajeja |



Marwadi Education Foundation’sGroup of Institutions

(MEFGI)

Rajkot-Morbi Road, At & PO : Gauridad, Rajkot 360 003. Gujarat. India.



**Faculty of Computer Applications (FCA)**

****

**This is to certify that the project work entitled**

**Make Deal**

**submitted in partial fulfillment of the requirement for**

**the award of the degree of**

**Master of Computer Applications (Integrated)**

**of the**

**Gujarat Technological University**

**is a result of the bonafide work carried out by**

**Yash Dhameliya (165220686009)**

**Krunal Bhanderi (165220686004)**

**during the academic year 2020 – 2021**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Faculty Guide** |  |  |  | **HOD** |

**External Viva**

|  |  |  |
| --- | --- | --- |
| **Name of the Examiners** |  | **Signature with Date** |

**CERTIFICATE**

**DECLARATION**

We hereby declare that this project work entitled **MakeDeal** is a record done by us.

We also declare that the matter embodied in this project is genuine work done by us and has not been submitted whether to this University or to any other University / Institute for the fulfillment of the requirement of any course of study.

Place: **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Date: **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Yash Dhameliya (165220686009) Signature: \_\_\_\_\_\_\_\_\_**

**Krunal Bhanderi (165220686004) Signature: \_\_\_\_\_\_\_\_\_**

**ACKNOWLEDGEMENT**

It is indeed a great pleasure to express my thanks and gratitude to all those who helped me. No serious and lasting achievement or success one can ever achieve without the help of friendly guidance and co-operation of so many people involved in the work.

I am very thankful to my guide **Dr. Sunil Bajeja,** the person who makes me to follow the right steps during a project work. I express my deep sense of gratitude to for his guidance, suggestions and expertise at every stage. A part from that his/her valuable and expertise suggestion during documentation of my report indeed help me a lot.

Thanks to my friend and colleague who have been a source of inspiration and motivation that helped to me during my project work.

I would heartily thankful to Head and Dean of our department **Dr. Sunil Bajeja and Dr. R. Sridaran** to give me an opportunity to work over this report and for their end-less and great support. And to all other people who directly or indirectly supported and help me to fulfil my task.

**Yash Dhameliya (165220686009) Signature: \_\_\_\_\_\_\_\_\_**

**Krunal Bhanderi (165220686004) Signature: \_\_\_\_\_\_\_\_\_**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Chapters** | **Particulars** | **Page No.** |
| **1** | **SYNOPSIS** | 1 |
| **2**  2.1  2.2 | **PREAMBLE**  Project Description  Modules and Description | 2  3 |
| **3**  3.1  3.2 | **TECHNICAL DESCRIPTION**  Hardware Requirement  Software Requirement | 5  5 |
| **4**  4.1  4.2 | **SYSTEM DESIGN AND DEVELOPMENT**  File Structure with Description  Flow Structure | 6  8 |
| **5** | **SYSTEM TESTING** | 16 |
| **6** | **SCREEN SHOTS** | 18 |
| **7**  7.1  7.2  7.3  7.4  7.5  7.6 | **AGILE DEVELOPMENT**  Agile Project Charter  Agile Project Plan  Agile User Story  Agile Release Plan  Agile Sprint Backlog  Agile Test Plan | 25  29  32  34  \37  39 |
| **8** | **CONCLUSION** | 40 |
| **9**  9.1  9.2 | **BIBLIOGRAPHY**  Online References  Offline References | 41  41 |

**FIGURE INDEX**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No.** | **Figure No.** | **Particulars** | **Page No.** |
| 1 | 4.2.1 | Class Diagram | 8 |
| 2 | 4.2.2 | Use case Diagram | 9 |
| 3 | 4.2.3 | Sequence(User) Diagram | 10 |
| 4 | 4.2.4 | Sequence(Admin) Diagram | 11 |
| 5 | 4.2.5 | Activity(User) Diagram | 12 |
| 6 | 4.2.6 | Activity(Admin) Diagram | 13 |
| 7 | 4.2.7 | ER Diagram | 14 |
| 8 | 4.2.8 | 1-Level DFD Diagram | 15 |
| 9 | 6.1 | Login | 18 |
| 10 | 6.2 | Signup | 18 |
| 11 | 6.3 | Home(User) | 19 |
| 12 | 6.4 | Ads list | 19 |
| 13 | 6.5 | View Ad | 20 |
| 14 | 6.6 | Update User Detail | 21 |
| 15 | 6.7 | Home(Admin) | 21 |
| 16 | 6.8 | Manage User | 22 |
| 17 | 6.9 | Manage Ads | 22 |
| 18 | 6.10 | Ads List(Admin) | 23 |
| 19 | 6.11 | Chat List | 23 |
| 20 | 6.12 | Chat Box | 25 |

**TABLE INDEX**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No.** | **Table No.** | **Particulars** | **Page No.** |
| 1 | 3.1 | Hardware Requirement | 6 |
| 2 | 3.2 | Software Requirement | 6 |

**SYNOPSIS**

**Make deal** is the platform which provides local communities in high-growth markets with vibrant online marketplaces. **Make Deal** connects local people to buy, sell or exchange used goods and services by making it fast and easy for anyone to post a listing through their mobile phone or on the web.

**PREAMBLE**

**PROJECT DISCRIPTION**

Make deal is project of online buy and sell of all products. Users can search the products they want to buy or sell in nearby their area. This website has different variety of products such as electronics, vehicles, furniture, home utilities, sports equipment etc.

Users have to create an account if they want to sell something on this website at best price deal. Users simply post the ad of the product / property they want to sell. Buyers & sellers contact each other with messages or other contact details provided by the sellers.

In case of any query user can direct contact to the admin thorough admin feedback box. For post the ad on this website user have to create an account and sign in. Sellers have to add some information about product like photos, price, and product details. When user sign in there provide their buy & sell history.

Second part is of admin. Admin can see all the User details and manage the data. If any post is not proper or negative, then admin can delete that post or block the user.

**MODULES**

* Categories: -
* Properties: -

In this module users who want to buy property like lands, shops & offices, House, Flat etc.

* Electronics: -

In this module user select the type of electronics like phones, laptops, camera etc.

* Furniture: -

Users can find all the furniture with selecting the type of furniture they require.

* Fashion: -

In this Module all the fashionable things like clothes, shoes, jewelries etc. are provided by sellers.

* Vehicles: -

In this module all types of vehicles like bikes, cars, commercial vehicles etc. listed to be sold.

* Sports & Hobbies: -

In this module all type of sports equipment and the things related to hobbies are listed by the sellers.

* Manage ads(user): -
* Ad post:-

User who have to sell something then simply user can post an ad with giving photos & add description and also have to provide contact details so that buyers can direct contact to the sellers.

* Remove Post:-

If Product is sold out or some mistakes done at a time of adding a post, then user can delete that post.

* + - List Ads:-

User can see all the ads posted by other users.

* + - Search: -

In this module users can search what they looking for and also enter the company name. This web site there is providing advance search that is filters like location and price wise.

* Manage User(admin): -

If user don’t follow rules and post some improper ads so admin can remove the user.

* Manage Ads(admin): -

Admin can view list of all user’s ads. And also can delete the ads.

* My Account(user):-

In this user can get all his details like list of personal detail, posted ads, sold product, my chats, favorite list and update details like contact information, account password.

* About Us:-

This will give information about this site.

* Contact Us:-

This will give the contact details of admin like E-mail id, contact detail.

* Terms & condition:-

This will give display the rules of using the web site.

**3. TECHNICAL DESCRIPTION**

**3.1 Hardware Requirement**

|  |  |
| --- | --- |
| Preprocessor | Intel Pentium dual core |
| Hard Disk | 1 GB & above |
| RAM | 256 MB & above |

**3.2 Software Requirement**

|  |  |
| --- | --- |
| Programing Tool | Notepad/Notepad++ |
| Data Storage | MYSQL |
| Software | XAMPP/WAMP |

**4. SYSTEM DESIGN AND DEVELOPMENT**

**4.1 File structure with Description**

The record will be stores in tabular format as below:

4.1.1 LOGIN

|  |  |  |  |
| --- | --- | --- | --- |
| **Field\_Name** | **Data\_type** | **Size** | **Constraint** |
| Id | integer | 4 | Primary key |
| User\_id | Varchar | 20 | Not null |
| Password | Varchar | 8 | Not null |

4.1.2 USER

|  |  |  |  |
| --- | --- | --- | --- |
| **Field\_Name** | **Data\_type** | **Size** | **Constraint** |
| Id | integer | 4 | Primary key |
| Name | Varchar | 40 | Not null |
| Address | Text | - | Not null |
| City | Varchar | 20 | Not null |
| State | Varchar | 20 | Not null |
| Contact | Number | 10 | Not null |
| Email | Varchar | 40 | Not null |
| Img | Varchar | 30 | Not null |
| Delete | varchar | 3 | Not null |

4.1.3 Product

|  |  |  |  |
| --- | --- | --- | --- |
| **Field\_Name** | **Data\_type** | **Size** | **Constraint** |
| Id | integer | 4 | Primary key |
| Type | Varchar | 40 | - |
| Sold | Varchar | 3 | Not null |
| Delete | Varchar | 3 | Not null |
| Price | Integer | 7 | Not null |
| Imgs | Varchar | 30 | Not null |
| Description | Text | - | Not null |
| Title | Varchar | 200 | Not null |
| User\_id | Varchar | 4 | Foreign key |

4.1.4 PROPERTIES

|  |  |  |  |
| --- | --- | --- | --- |
| **Field\_Name** | **Data\_type** | **Size** | **Constraint** |
| Id | integer | 4 | foreign key |
| Bedrooms | integer | 2 | Not null |
| Bathrooms | Integer | 2 | Not null |
| total floor | integer | 2 | Not null |
| Floor no | Integer | 2 | Not null |
| Furnishing | Varchar | 30 | Not null |
| Area | Varchar | 10 | Not null |

4.1.5 Vehicle

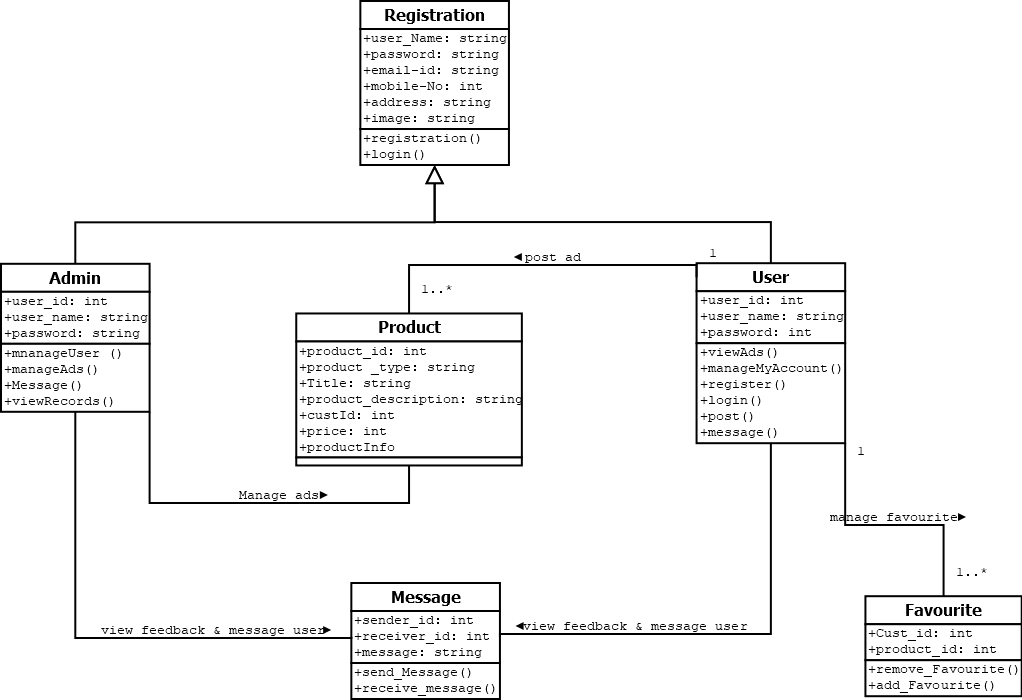
|  |  |  |  |
| --- | --- | --- | --- |
| **Field\_Name** | **Data\_type** | **Size** | **Constraint** |
| Id | integer | 4 | foreign key |
| Model | Varchar | 20 | Not null |
| Year | Varchar | 4 | Not null |
| Fuel type | Varchar | 7 | Not null |
| Kms driven | Integer | 6 | Not null |

4.1.6 Message

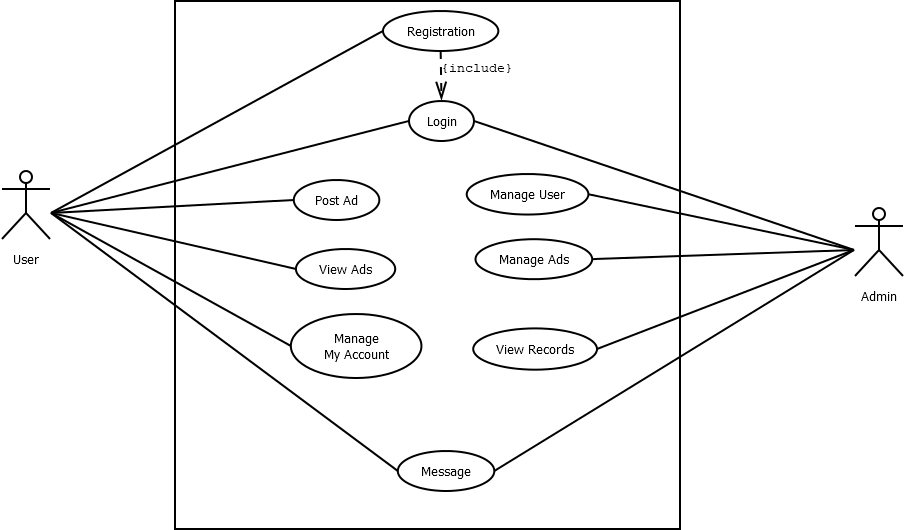
|  |  |  |  |
| --- | --- | --- | --- |
| **Field\_Name** | **Data\_type** | **Size** | **Constraint** |
| Id | integer | 4 | Primary key |
| Send message | Varchar | 200 | Not null |
| Receive message | Varchar | 200 | Not null |

**4.2. Flow structure**

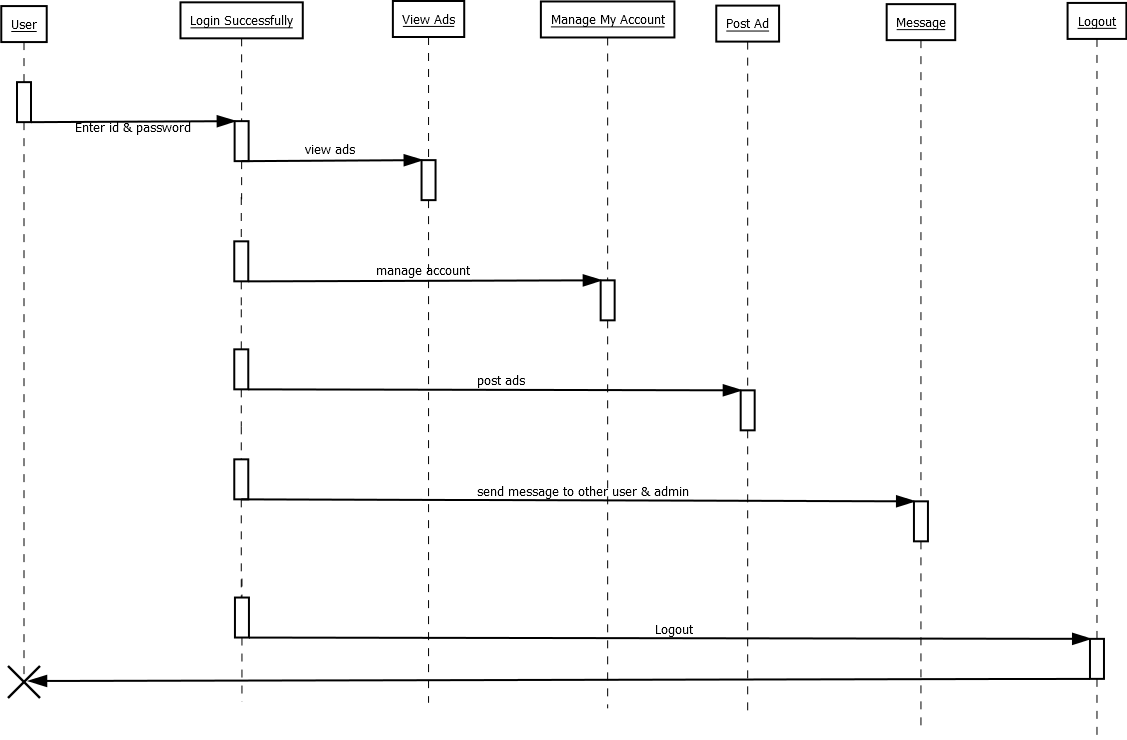
**4.2.1. Class Diagram**

****

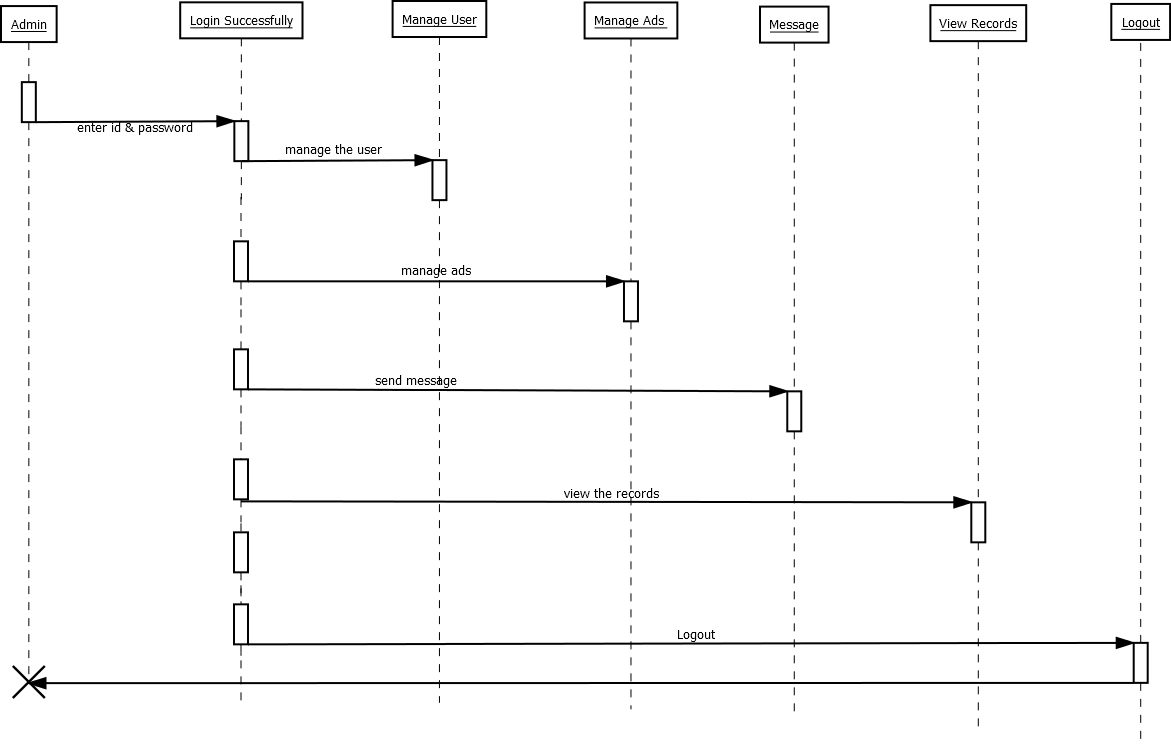
**4.2.2 Use case Diagram**

****

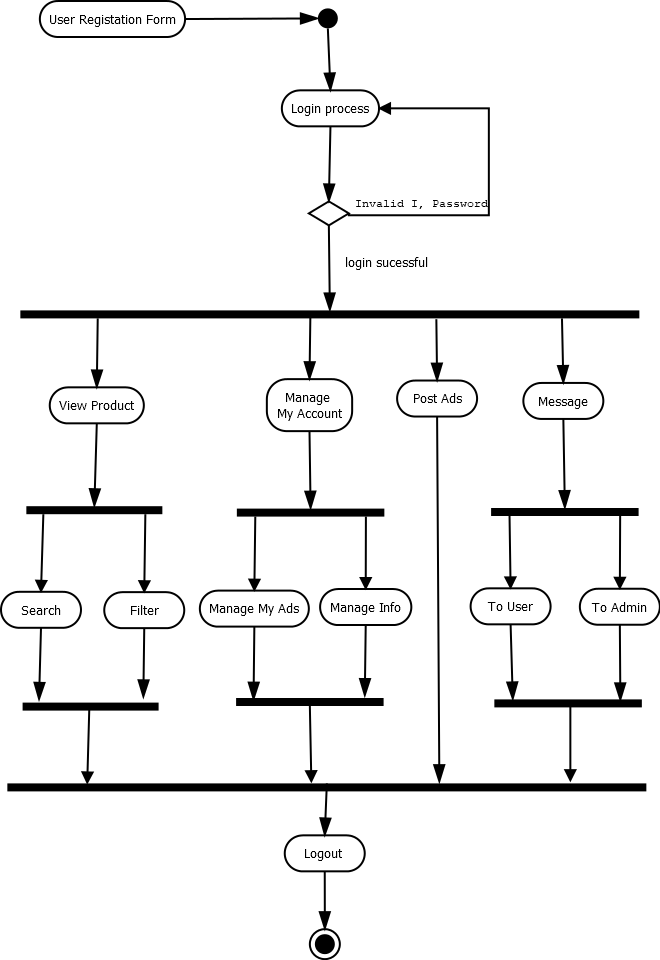
**4.2.3. Sequence Diagram (user)**

****

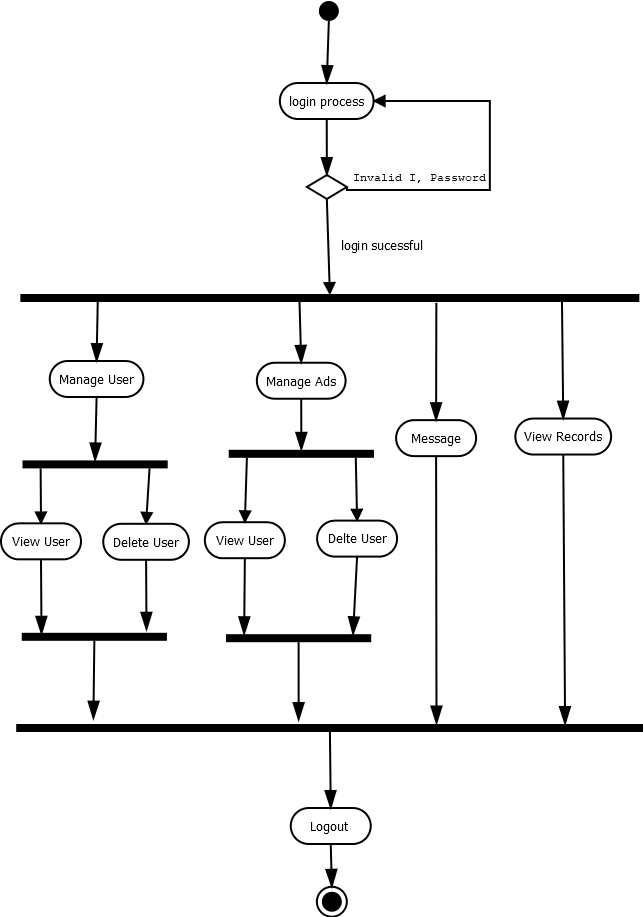
**4.2.4 Sequence Diagram (Admin)**

****

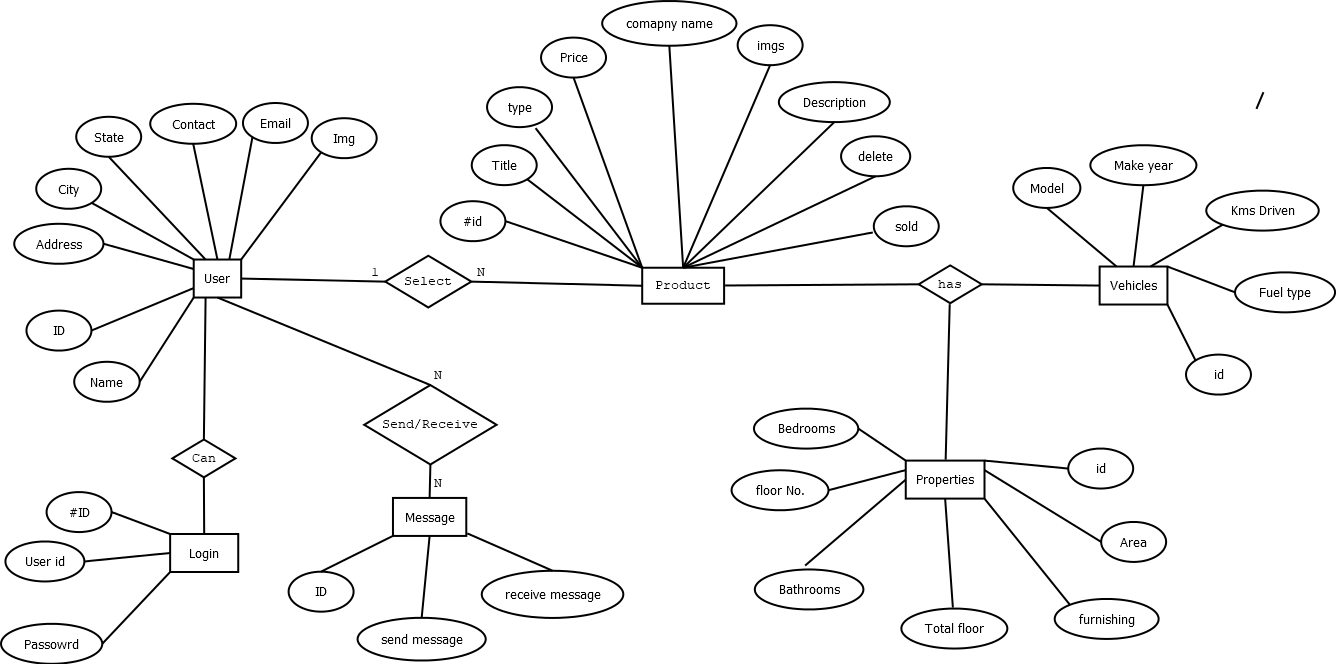
**4.2.5 Activity Diagram(User)**

****

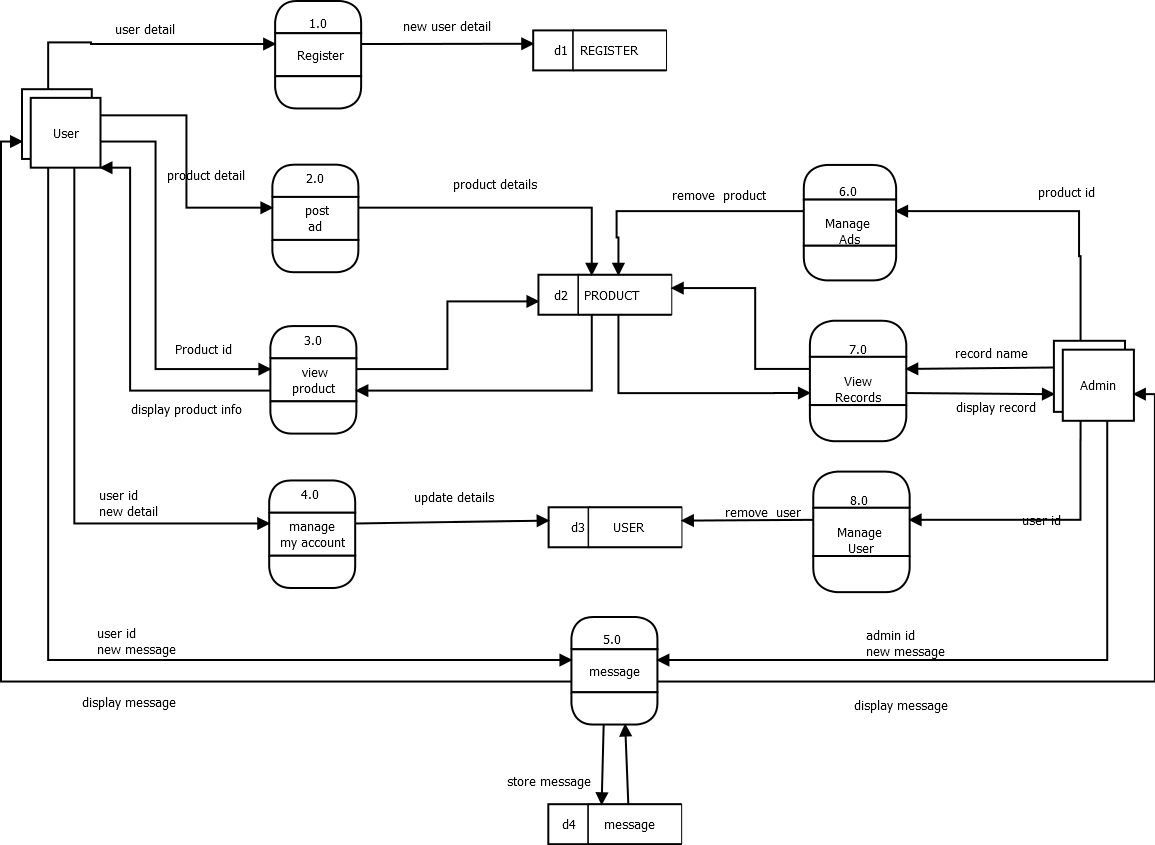
**4.2.6 Activity Diagram(Admin)**

****

**4.2.7 ER Diagram**

****

**4.2.8 1-Level DFD Diagram**

****

**5. SYSTEM TESTING**

**5.1 Login**

|  |  |
| --- | --- |
| Page name | Login page |
| Description | In this page, one form will be provided  to user(Login Form) |
| Input | User id and Password |
| Output / Result | If information is correct then enter to system otherwise error message. |

**5.2 Signup**

|  |  |
| --- | --- |
| Page name | Signup page |
| Description | In this page, one form will be provided  to user(Signup Form) |
| Input | User id , name, Password, address, E-mail id, contact No. |
| Output / Result | If information is correct then enter to system otherwise error message. |

**5.2 Add Product**

|  |  |
| --- | --- |
| Page name | Add Product |
| Description | In this page, one form will be provided  to user for adding product details |
| Input | Title, Price, Address, Type, Description, Image. |
| Output / Result | If information is correct then data will stored in database |

**5.3 Delete Product(Admin/User)**

|  |  |
| --- | --- |
| Page name | Delete product |
| Description | In this page, the product of that customer remove which is selected. |
| Input | Product id. |
| Output / Result | Update in database will stored. |

**5.4 Search Product**

|  |  |
| --- | --- |
| Page name | Search Product |
| Description | In this page, the product relevant to search is display. |
| Input | Product title, filter value. |
| Output / Result | Display product according to search value and filter value. |

**5.4 Remove User (Admin)**

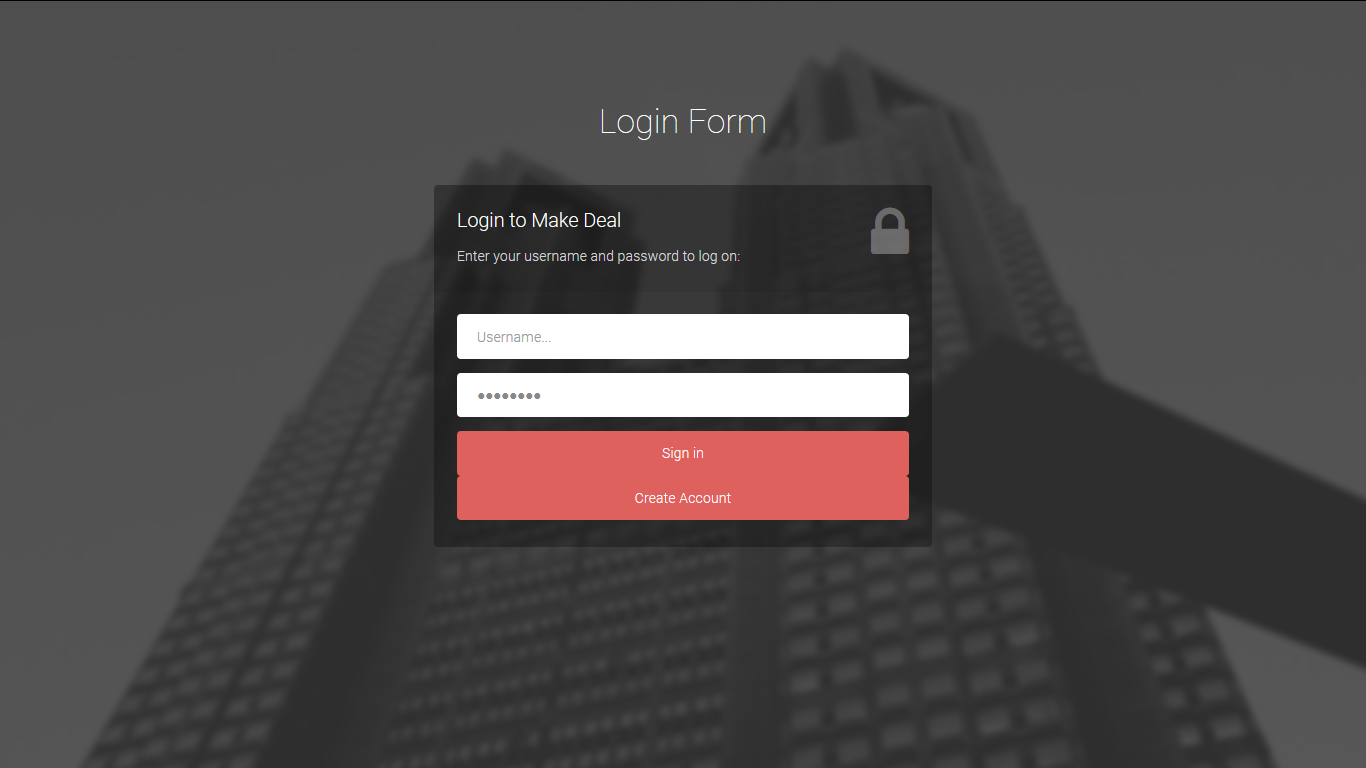
|  |  |
| --- | --- |
| Page name | Remove User |
| Description | In this page, Admin delete the user who don’t follow guidelines. |
| Input | User id |
| Output / Result | If information is correct then it will update the database value. |

**5.6 Chat Box**

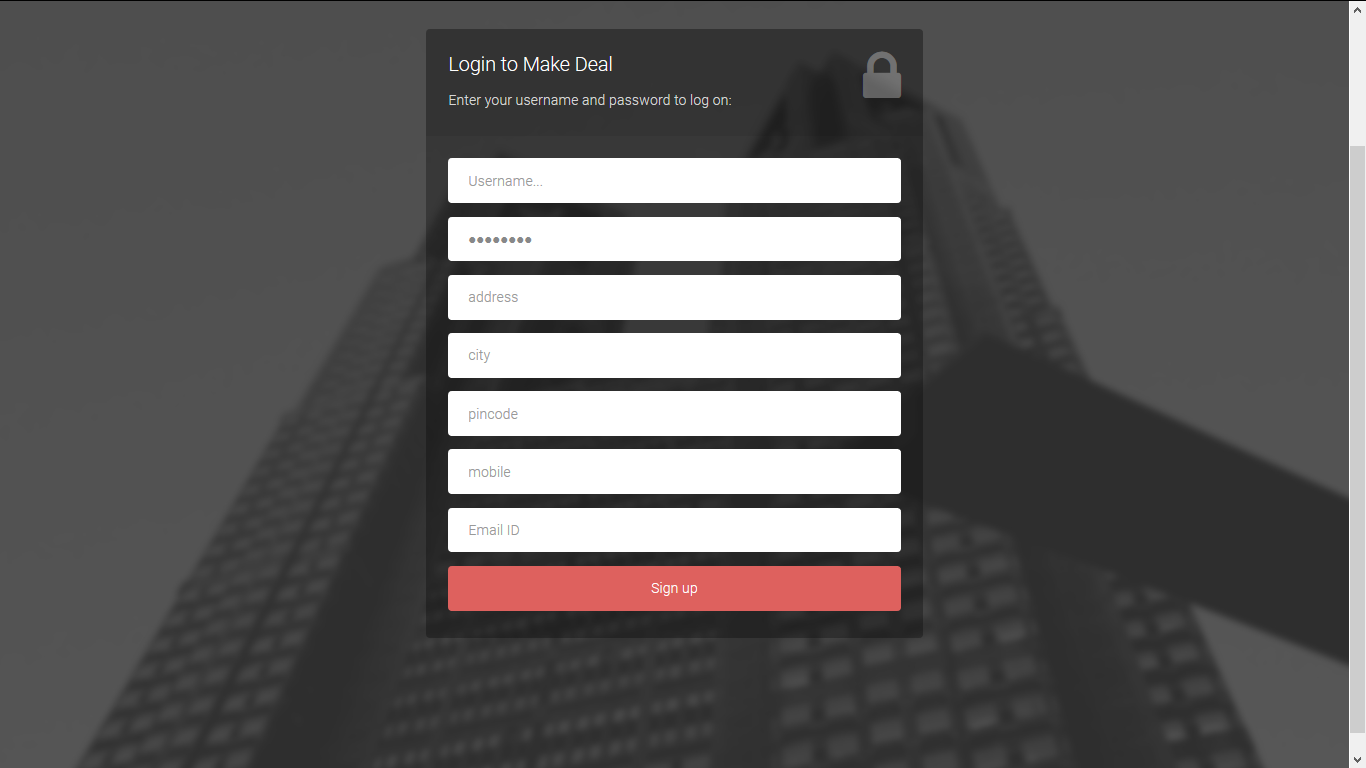
|  |  |
| --- | --- |
| Page name | Chat Box |
| Description | In this user can chat with other user(buyer/seller) and admin. |
| Input | User id. |
| Output / Result | Display all receive messages. |

**6.SCREEN SHOTS**

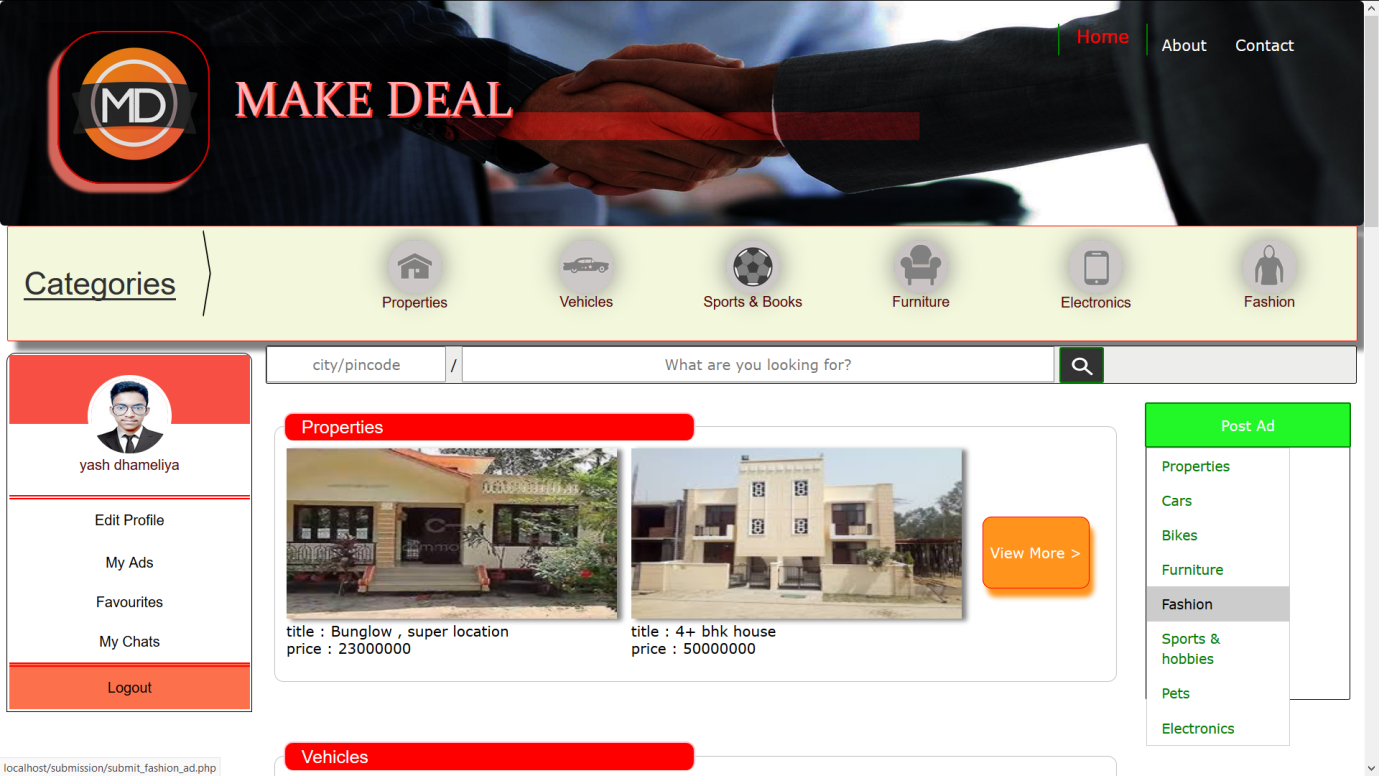
6.1) Login:-



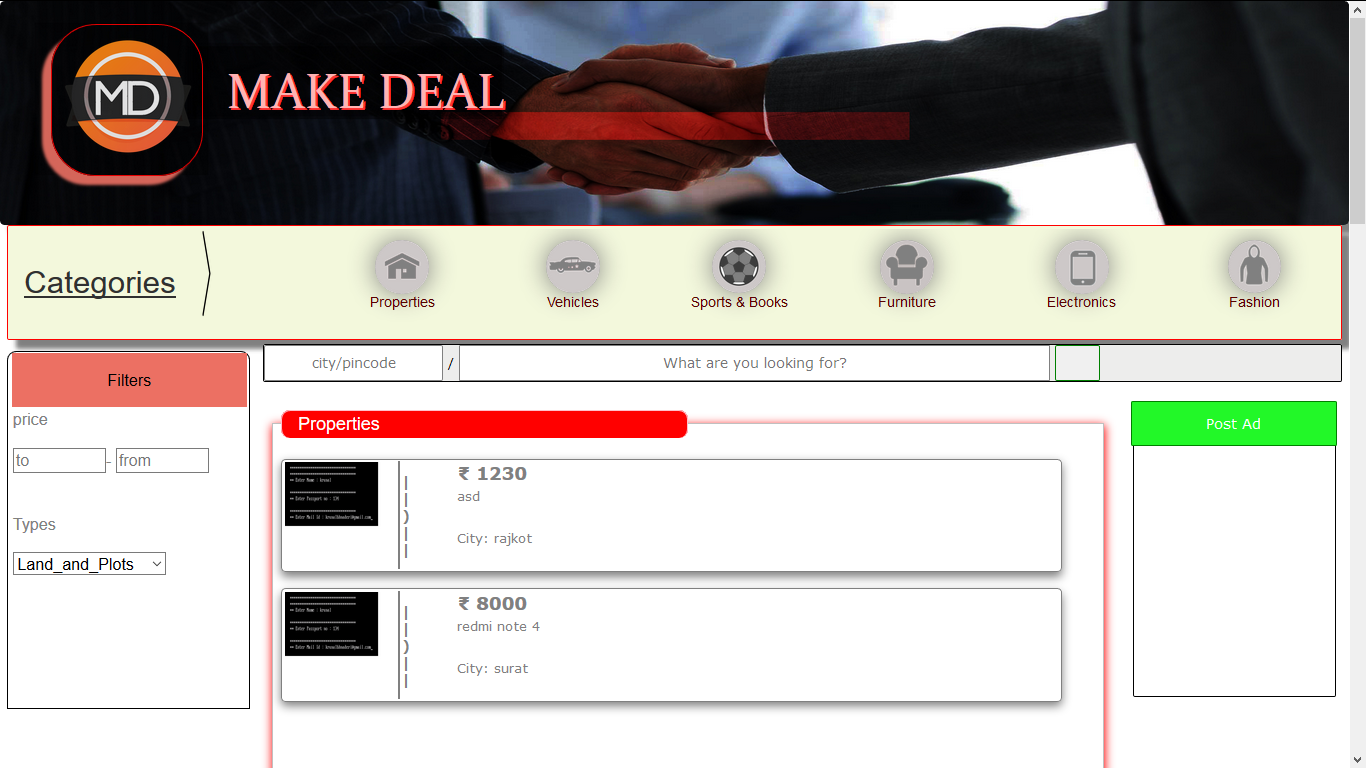
6.2) Sign up:-



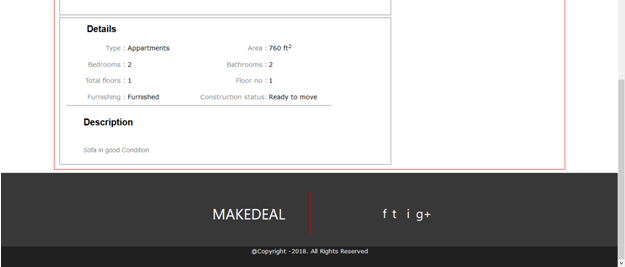
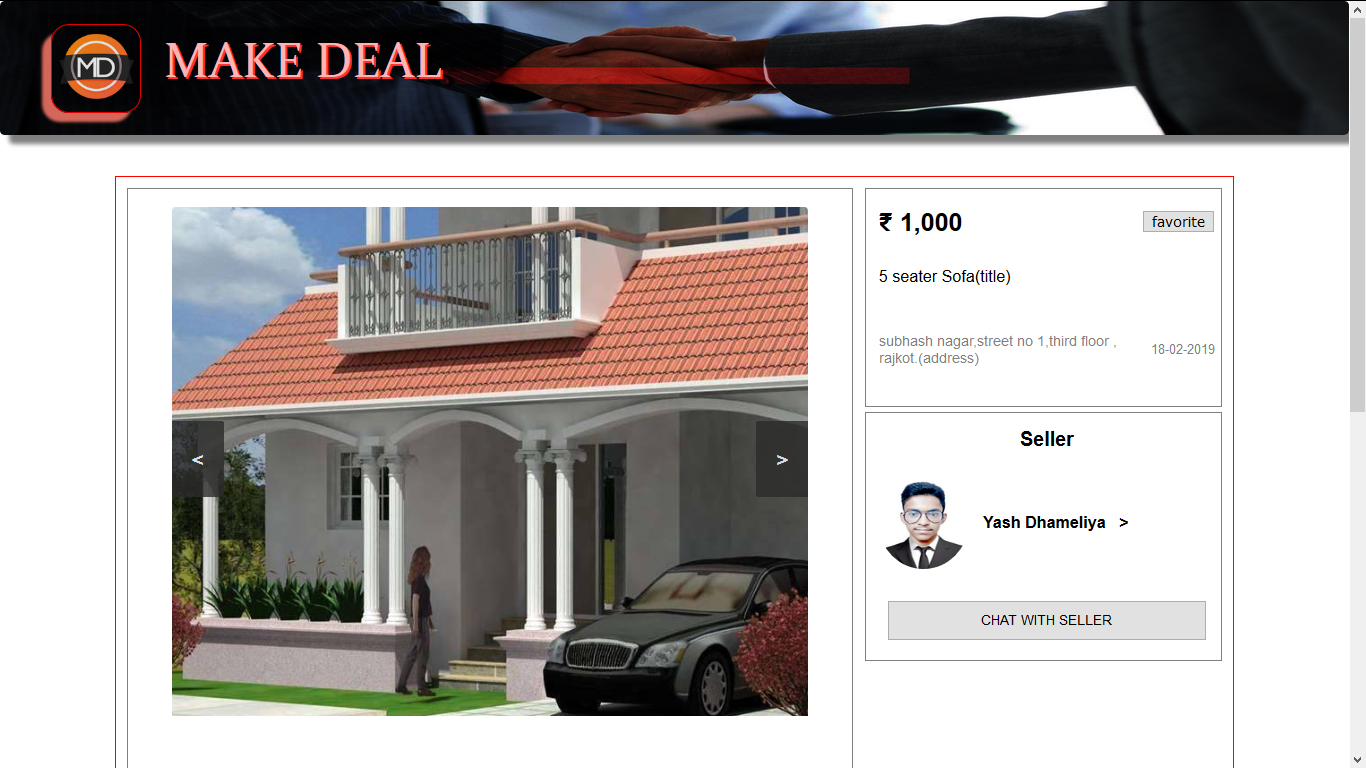
6.3) Home Page(User):-



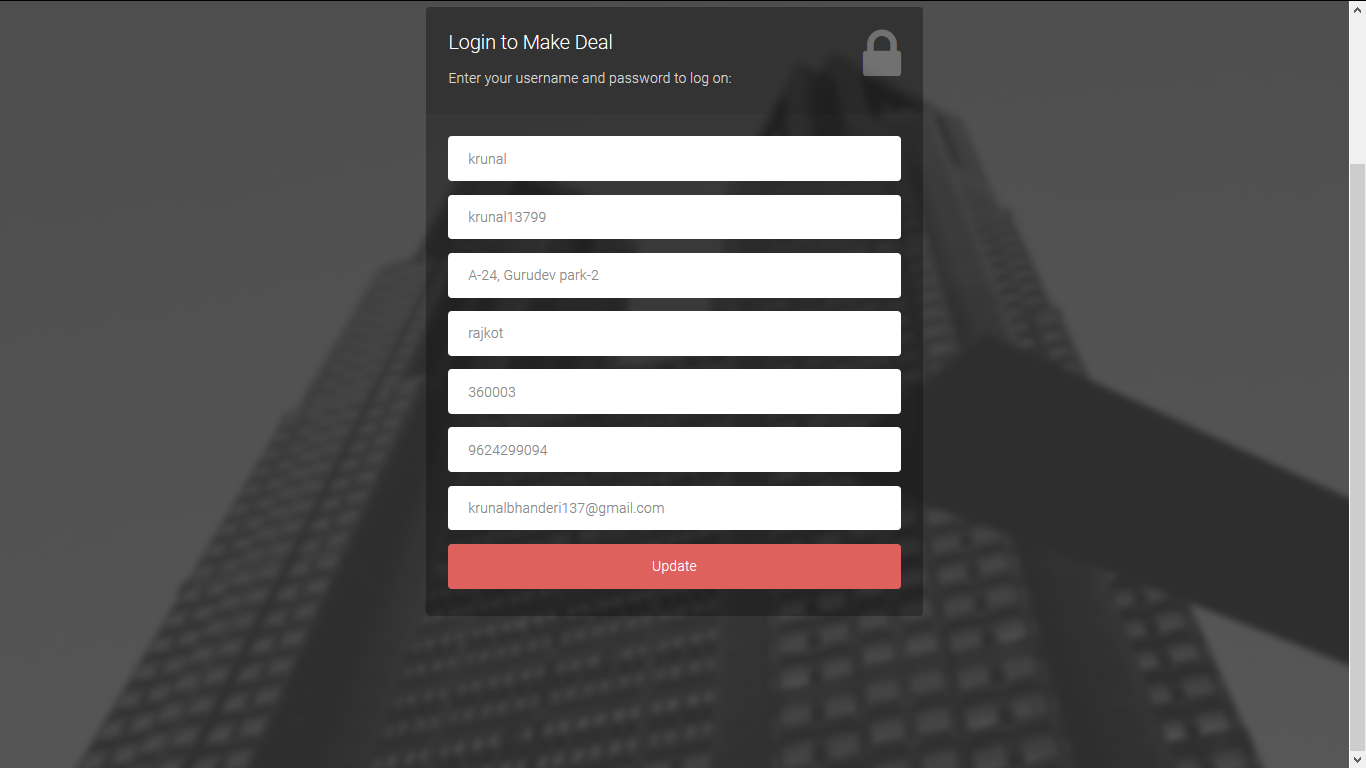
6.4) Ads list:-



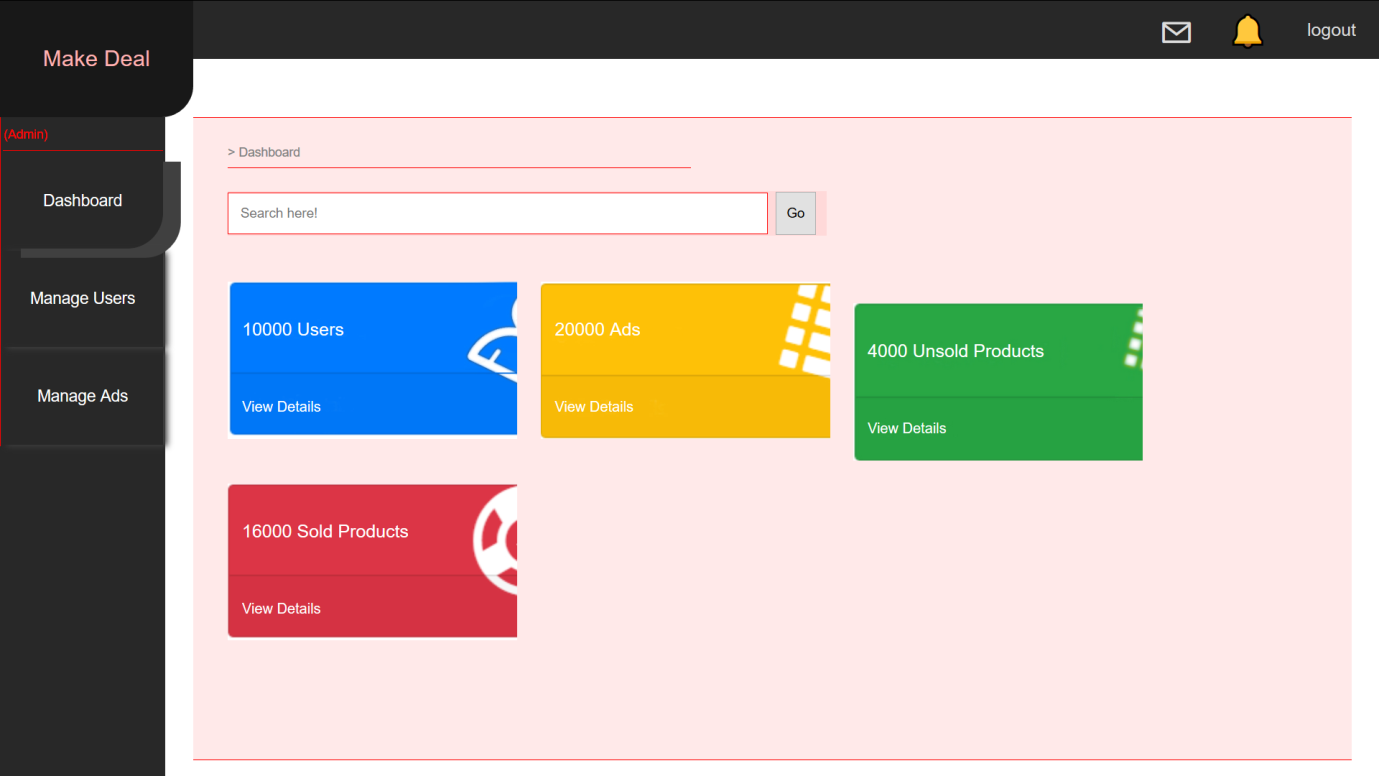
6.5) View ad:-



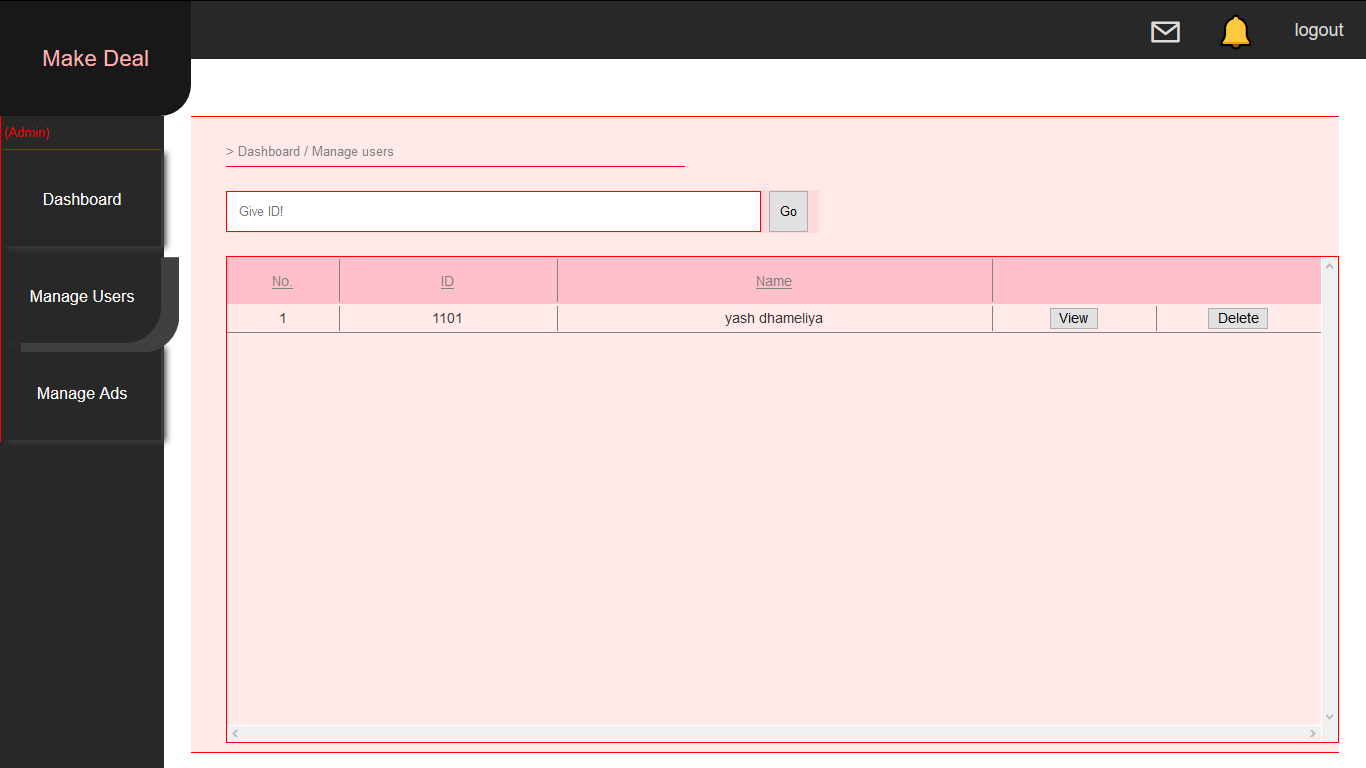
6.6) Update User Details:-



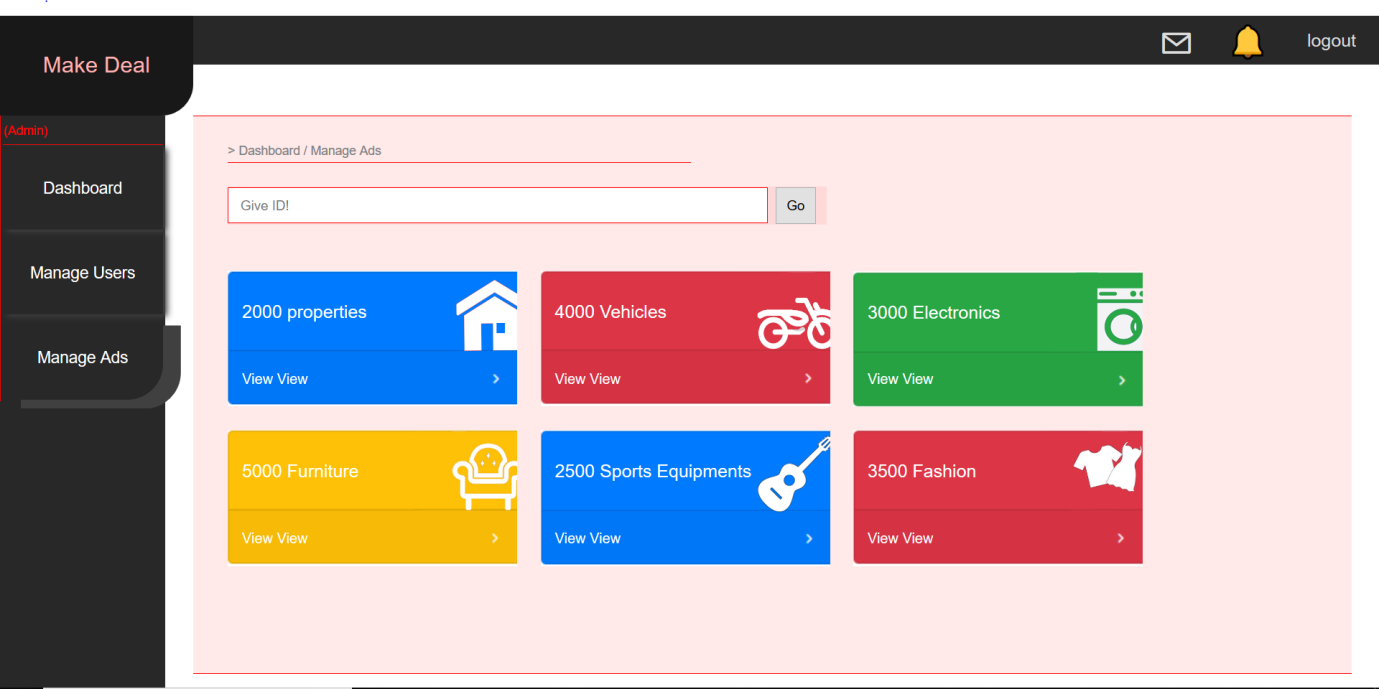
6.7) Home page(Admin):-



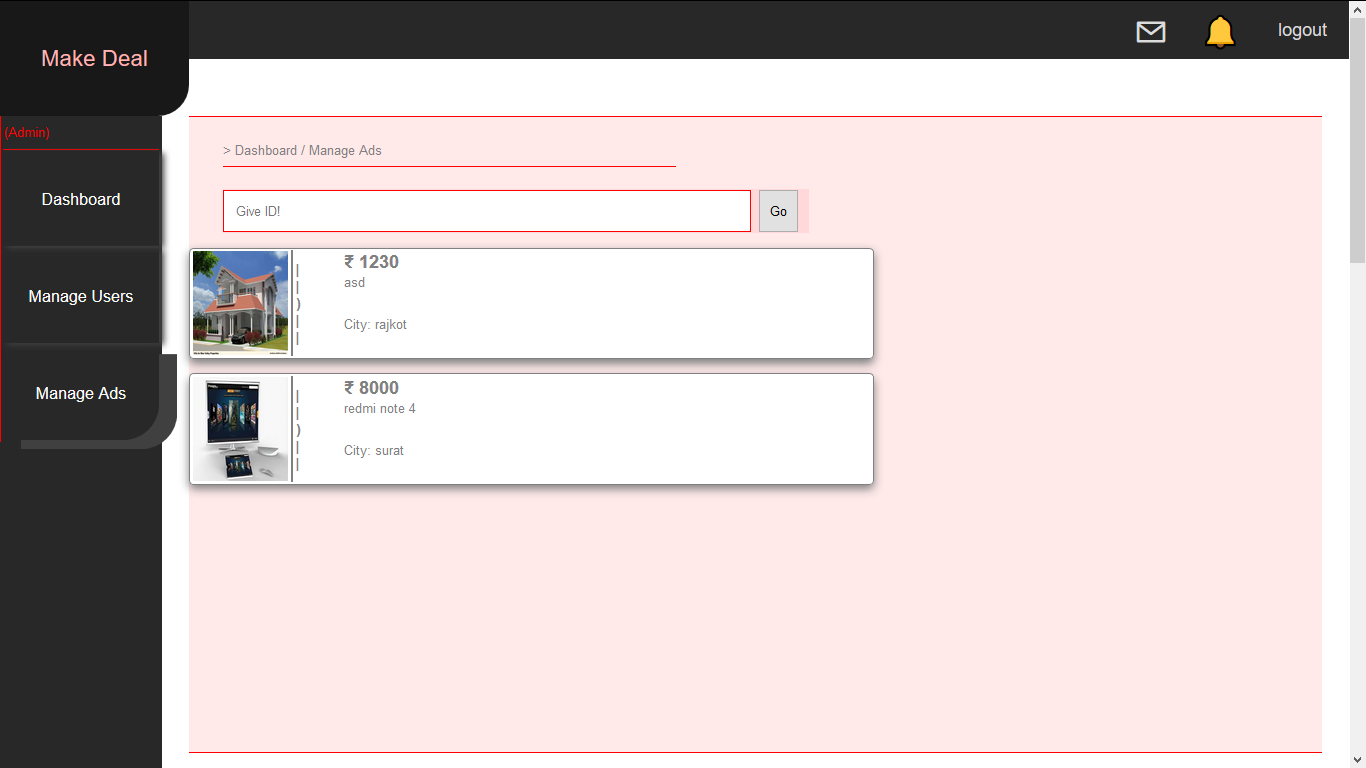
6.8) Manage User:-



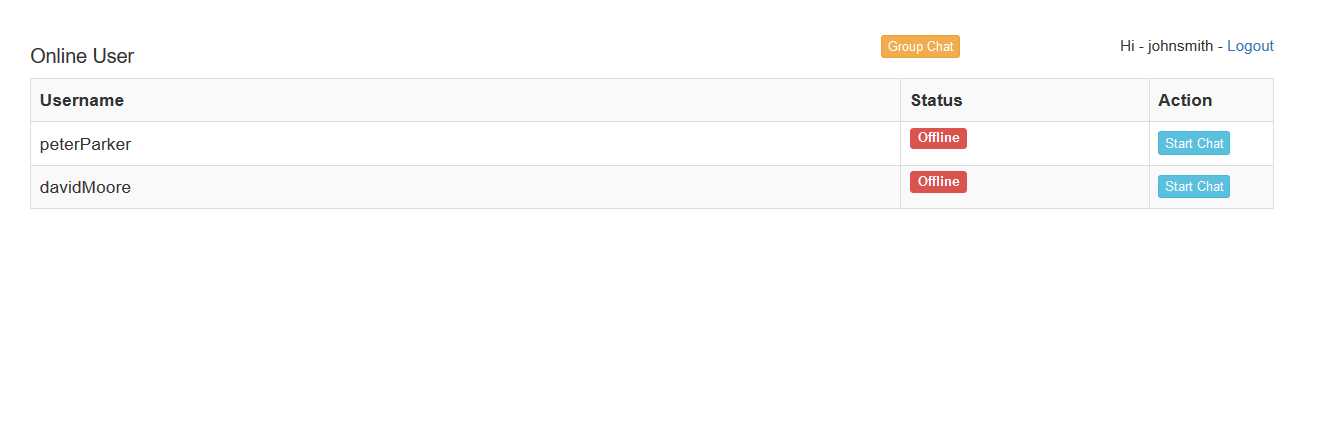
6.9) Manage Ads(Admin):



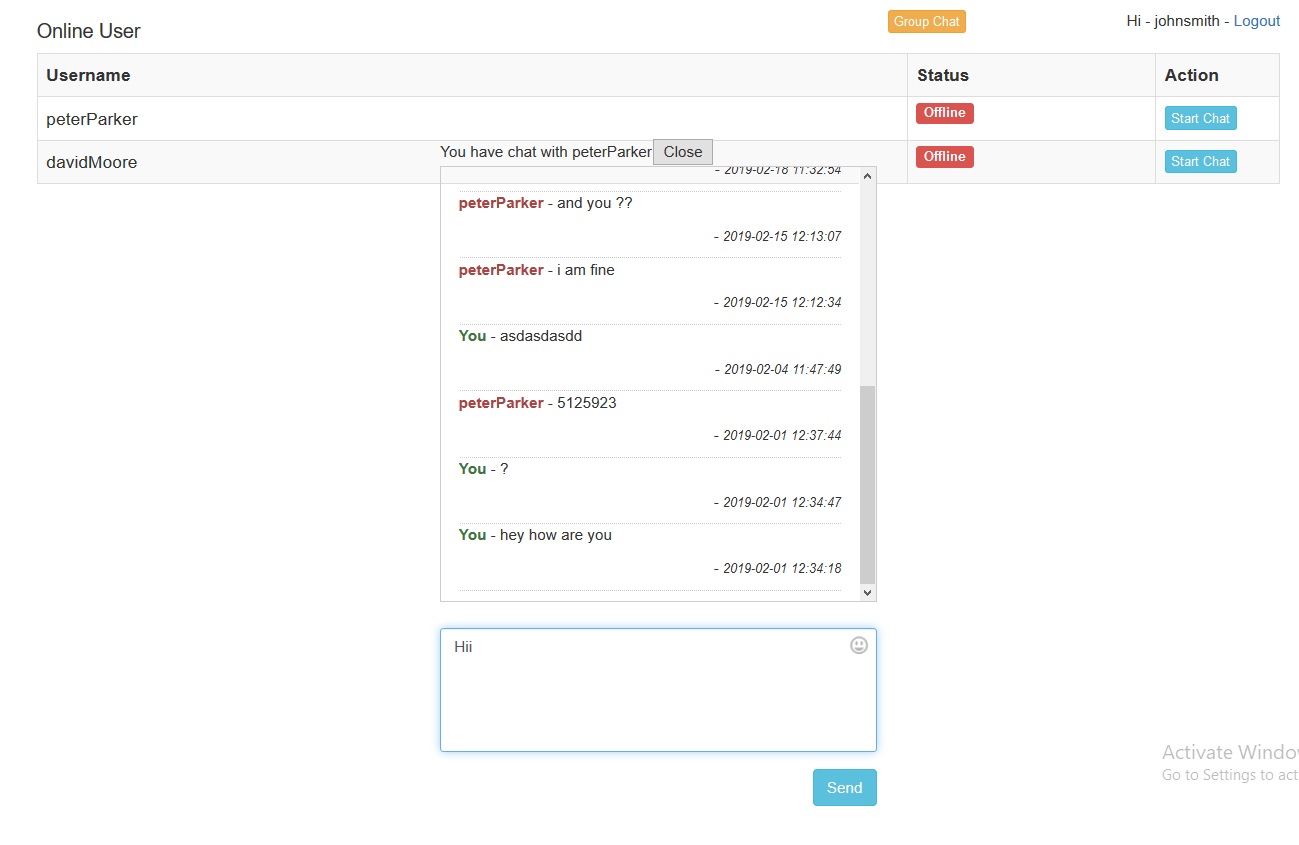
6.10) Ads list(Admin):-



6.11) Chat List:-



6.12) Chat Box:-



**7. Agile Documentation**

**7.1 Agile Project Charter**

In Agile project management, the Team Charter can be thought of as the foundation upon which all of the team’s work, rules, tools, and behaviors are built. Unlike traditional project management, where a charter defines the project scope and success criteria, often pre-determined by senior management/sponsors, an Agile Team Charter is built and agreed upon by the project team exclusively. The results of the project team directly contributing to and building the charter are immediate buy-in and a vested interest in the success of the project. When project team members are able to directly contribute to and influence a project, they will be much more motivated for success.

## What Does the Agile Team Charter Include?

There is no standard or universal template for an Agile Team Charter. One of the reasons for this is Agile’s emphasis on people instead of processes. Additionally, there’s more of a focus on interpersonal communication over formalized documentation in an Agile project environment.

It is also important that the Team Charter be readily available or prominently posted in a team room so all team members have immediate access to it.

## Project Specific Information

As Agile project management is iterative by nature and evolves throughout the project life cycle, there is not much need for a great level of detail in the charter. In fact, since the charter is created in the beginning of the project, the team may not even know or understand a great level of detail yet. The project specific information that is included in the charter should be limited to a high level vision (why the project has been initiated), a description of the mission or objectives, and what criteria constitute success for the project. All of this information is at a high enough level that it should be known to the project team at the outset of the project.

**Example of Agile Project Charter**

**Project Name : Invoice Tracking Database**

**Vision :**Help the billing department reduce the number of monthly past-due accounts by providing an easy to use platform for tracking accounts receivable.

**Mission :**Create an invoice tracking database that enables billing employees to view real-time account statuses, engage vendors proactively to resolve accounts, and generate weekly and monthly account reports.

**Project Team**

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | **Phone** |
| A. Black | Product Owner | 1234567894 |
| B. White | Scrum Master | 1234567894 |
| C. Green | Programmer | 1234567894 |
| D. Teal | Database Administrator | 1234567894 |
| E. Gray | Network Administrator | 1234567894 |
| F. Brown | Billing Coordinator | 1234567894 |

**Rules of Behavior:**

1. All team members will treat each other with respect at all times
2. Constructive feedback is a valuable part of our success so we will not take offense and all team members will ensure all feedback is provided in a constructive manner
3. Open communication among the team is always welcomed and valued
4. We will recognize and celebrate all individual and team accomplishments
5. All personal cell phones will be turned off prior to beginning any of our meetings or discussions
6. We will accept responsibility and be accountable for our actions
7. We will give consideration to whomever is speaking and avoid sidebars or speaking over one another
8. We will work collaboratively when possible and use a consensus approach when making team decisions

**Communications:**

1. We will hold regular daily meetings in the team room at 9am each work day
2. We will make every effort to attend all scheduled meetings in person (exceptions being scheduled and/or sick leave)
3. We will update our tasks on the kanban board each work day morning before 9am
4. Meeting minutes will be sent out within 24 hours of each meeting
5. The responsibility for meeting scribe will be shared by all team members on a rotating basis
6. If a meeting must be cancelled or additional meetings are required, the Product Owner will send out notifications as early as possible
7. All team members are expected to be on time for all meetings

**Agile Charter**

|  |  |
| --- | --- |
| **GENERAL PROJECT INFORMATION** | |
| **PROJECT NAME** | Make Deal |
| **PROJECT SPONSOR** | Self |
| **PROJECT MANAGER** | Mr. Stefan Salvator |
| **EMAIL ADDRESS** | [makedeal@gmail.com](mailto:makedeal@gmail.com) |
| **PHONE NUMBER** | 962499094 |
| **ORGANIZATIONAL UNIT** | Marwadi University |
| **START DATE** | July 11, 2020 |
| **EXOECTED END DATE** | October 15, 2020 |

|  |  |
| --- | --- |
| **DESCRIBE THE PROBLEM OR ISSUE, GOALS, OBJECTIVES, AND DELIVERABLES OF THIS PROJECT** | |
| **PROBLEM OR ISSUE** | Difficult to maintain every user’s product data. |
| **PURPOSE OF PROJECT** | Make buying and selling of used product easy. |
| **GOALS / METRICS** | Affective website which will which connects  Buyer and seller. |

|  |  |
| --- | --- |
| **DEFINE THE PROJECT SCOPE AND SCHEDULE** | |
| **WITHIN SCOPE** | 1.User register them self and post information of product or purchase other products . 2. Admin can manage user and product data. |
| **OUTSIDE OF SCOPE** | 1. it is a regional project. 2. There is no online payment available. |

|  |  |
| --- | --- |
| **PROJECT RESOURCE** | |
| **STACK HOLDER** | 1. Krunal Bhanderi (CEO) 2. Yash DhameliyaCFO) 3. Nashir Junach (DBA) 4. Mr. Stefan Salvator (Project Manager) 5. Mr. Ram Jani (Jr. Developer) 6. Mr. John Smith (Sr. Developer) 7. Mr. Nilesh Modi (Sr. Developer) 8. Ms. Caroline Forber (Sr. Tester) 9. Dr. Usman Memon (Client) |

|  |  |
| --- | --- |
| **DESCRIBE PROJECT RISKS, CONSTRAINTS, AND ASSUMPTIONS** | |
| **RISKS** | 1. Mid Project Change in Scope 2. Heavy Site load could lead to server failure 3. Delay in project due to unforseen Complication |
| **CONSTRAINTS** | 1. Budget Is enough to fulfil project Monetary requirements 2. Key person Will Not Leave the Organisation before complition of project |
| **ASSUMPTIONS** | 1. All the stack holder Complete assigned activity without any delay 2. Server are strong enough to handle the load |

**7.2 Agile Project Plan**

Project planning is focused on answering the questions, what needs to be built, when it needs to be completed, how much will it cost, and who needs to be involved. As project managers, we also want to know any dependencies between activities so we can minimize idle time and optimize the schedule.

Agile and traditional project planning deal differently with this uncertainty. Ultimately the goal is to create a project plan that moves the project from its initial state to where the project meets the project objectives. The traditional project planning approach creates a project plan by following several planning steps. These steps include:

1. Determine the project objectives
2. Collect the project requirements
3. Define the project scope on a work level
4. Identify dependencies between activities
5. Estimate work effort and dependencies
6. Prepare the overall schedule and project budget
7. Receive approval
8. Baseline the plan

It is an iterative project management process that promotes open communication amongst team members. The agile project plan template was created for teams that are using this process. The template was designed for project planners who need to communicate the progressive requirements of their projects to the team and stakeholders.

The agile project management process was created to remove the complexity out of managing projects, particularly as projects grow in size and scope. The agile process divides responsibilities amongst the whole team. These team members require a better, more visual way of seeing the long term requirements of a project.

There are many tools which are available for agile project planning. Some of the tools and information are as follows.

1. [JIRA](https://www.atlassian.com/software/jira/try) is a defect tracking tool which is used for Agile testing as well as project management. This tool is not only used for recording, reporting but also integrated with code development environment.

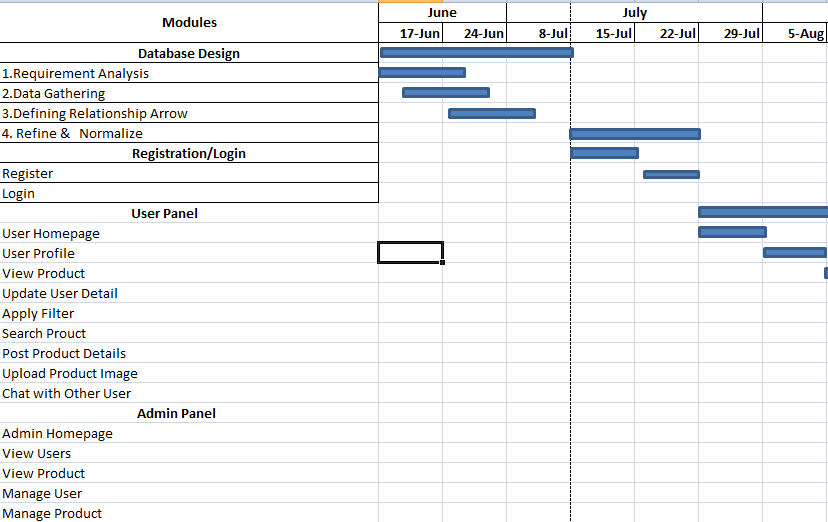
**Features:**

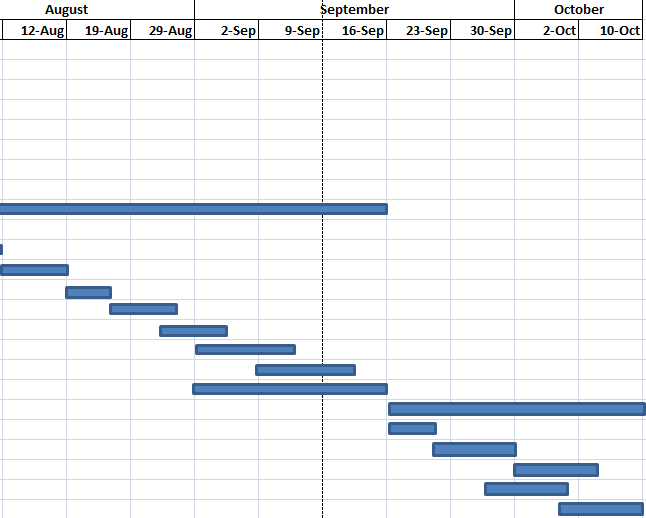
* JIRA Query Language helps to create quick filters with a single click
* Estimations help your team become more accurate and efficient
* Reporting functionality gives team critical insight into their agile process
* Extensive reporting functionality gives your team critical insight into their agile process.
* Allows creating custom workflows of any size which is helpful to build, test, and release software

2. [SoapUI](https://www.soapui.org/downloads/download-soapui-pro-trial.html) is an agile testing tool for service-oriented architectures (SOA) and REST. Its functionality includes web service inspection, invoking, development, functional testing, and load testing.

**Features:**

* It is open source testing tool
* Drag and Drop Test Creation
* It allows reusing functional test cases and security scans in just a few clicks
* It Supports Data-Driven Testing
* Multi Environment Support
* Allows service Simulation
* Static Content Mocking

****

****

**7.3 Agile User Story**

**What is a User Story?**

Agile is a value based development methodology, where all those features of the product which can add value to the customer are recognized, prioritized and developed based on customer needs. These valuable features are best described in users own words. Hence finding out who are the users is an important task. Once all the users are recognized, requirements which add value to the user are written down considering the needs of those specific users. Since those requirements are coming keeping users in mind, they are called User Stories.

**How to Write a User Story?**

User Stories are best written in following format.

**As a <User> , I want to <Have> so that <Benefit>**

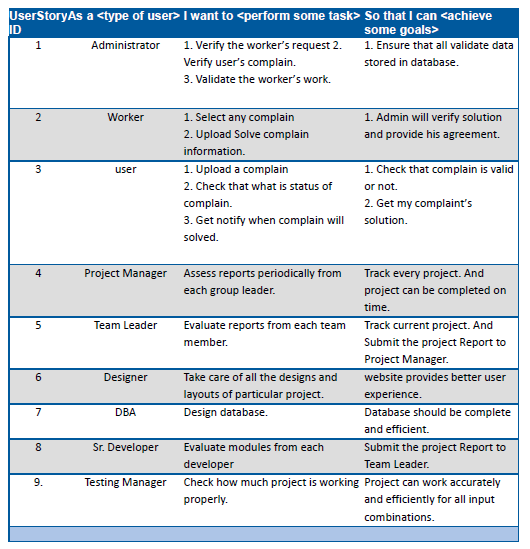
Here user story describes how a user (a manager, a clerk, a developer, a librarian, an owner etc.) will want to interact with the system and get a certain benefit out of that interaction.

**User**

A user is the user of the system.

* a managera manager a managera managera manager
* a clerka clerk a clerk a clerk
* a developera developer a developer a developer a developer
* a librariana librarian a librarian a librarian a librarian
* An owner etc.An owner etc. An owner etc. An owner etc. An owner etc.An owner etc.

**USER STORIES**



**7.3 Agile Release Plan**

Agile release planning is an approach to product management that takes into account the intangible and flexible nature of software development—as part of this approach, teams plan iterative sprints across incremental releases.

In other words, instead of trying to develop every proposed feature in one large, regimented project,[agile planning](https://www.lucidchart.com/blog/what-is-agile-methodology) breaks down the development process into stages called releases. In this context, releases are essentially periods of time set apart to work on a limited scope of the overall project. An agile release plan maps out how and when features (or functionality) will be released and delivered to users.

By scheduling a project into agile releases, product managers can better manage project constraints and adapt to evolving needs or challenges that arise through the development stage while regularly producing product deliverables for the end user.

## Elements of a product release map

Despite its name, agile release planning is highly structured. Each step is carefully outlined and measured to create high-level project calendars for teams to follow.

Release maps will vary slightly between organizations, but the general elements will include:

* The proposed release(s) for the project
* Plans for each release
* Subsequent iterations for the release(s)
* Plans for each iteration
* Feature development within an iteration
* Individual tasks necessary to deliver a feature

This level of planning, combined with an iterative schedule to account for the dynamic nature of software, is what makes agile product development so valuable.

The iterative release schedule gives teams the space to make course corrections without derailing the entire project, while the detailed roadmap and focus on the planning stage ensure everyone is on the same page.

## 

## How to create an agile release plan

### Step 1: Define your vision

One of the most important steps in the planning process is defining the vision for your product. The vision will guide subsequent decisions on which features to prioritize, where to focus effort and resources, and how to adapt if the project requires adjustment during development.

You may need to consult with executives or other high-level stakeholders to ensure your vision aligns with both the market and the organization’s overall objectives.

### Step 2: Rank the product backlog

Next, you will need to review your product backlog and rank the features. Use the product vision as well as input from stakeholders to determine product priorities and map out user stories. User stories are user-centered descriptions of the proposed functionality (or feature).

During this step, the product manager should outline a basic release plan or roadmap that includes the release goal, release target date, and the ranked user stories.

### Step 3: Hold a release planning meeting

Once the overall product vision and release map are outlined, it’s time to gather all stakeholders together in a release planning meeting to review the proposed plan, add to or edit the plan as needed, and align on the product deliverables.

This step ensures everyone is on the same page regarding strategy and collaboration before diving into the project.

**AGILE RELEASE PLAN**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Module** | **Task** | **Start** | **End** | **Status** | **Release Date** |
| **1** | Login / Registration | Registration | 08-07-20 | 16-07-20 | Relesed | 29-07-20 |
| Login | 17-07-20 | 22-07-20 |
| **2** | User Panel | Home | 23-07-20 | 29-07-20 | Relesed | 17-09-20 |
| Profile | 30-07-20 | 05-09-20 |
| View Products | 13-08-20 | 18-08-20 |
| Apply Filter | 19-08-20 | 25-08-20 |
| Search Product | 23-08-20 | 27-08-20 |
| Post Product Details | 02-09-20 | 13-09-20 |
| Upload Product Image | 09-09-20 | 19-09-20 |
| Chat with other user | 02-09-20 | 16-09-20 |
| **3** | Admin Panel | Home | 17-09-20 | 21-09-20 | Relesed | 10-10-20 |
| View User | 22-09-20 | 29-09-20 |
| View Product | 30-09-20 | 02-10-20 |
| Manage User | 27-09-20 | 01-10-20 |
| Manage Product | 01-10-20 | 10-10-20 |

**7.4 Agile Sprint Backlog**

The Sprint Backlog is the set of Product Backlog items selected for the Sprint, plus a plan for delivering the product Increment and realizing the Sprint Goal. The Sprint Backlog is a forecast by the Development Team about what functionality will be in the next Increment and the work needed to deliver that functionality into a “Done” Increment.

The Sprint Backlog makes visible all the work that the Development Team identifies as necessary to meet the Sprint Goal. To ensure continuous improvement, it includes at least one high priority process improvement identified in the previous Retrospective meeting.

The Sprint Backlog is a plan with enough detail that changes in progress can be understood in the [Daily Scrum.](https://www.scrum.org/resources/what-is-a-daily-scrum) The Development Team modifies the Sprint Backlog throughout the [Sprint](https://www.scrum.org/resources/what-is-a-sprint-in-scrum). Only the Development Team can change its Sprint Backlog during a Sprint. The Sprint Backlog is a highly visible, real-time picture of the work that the Development Team plans to accomplish during the Sprint, and it belongs solely to the Development Team.

### Parameters for the Sprint

Remember, a sprint is usually only over a two-week period, though this time can differ depending on the size of the team and project resources, so the sprint length is another variable to determine. The sprint, while short, must not over tax the team or force them to rush and produce a sloppy deliverable. Therefore, as the sprint backlog and the steps necessary to complete are being figured out by the development team, [brainstorm with them](https://www.projectmanager.com/blog/key-to-better-brainstorming), and open up a dialogue to determine what is feasible in terms of a strategy for the sprint.

**SPRINT BACKLOGS**

|  |  |  |
| --- | --- | --- |
| Story | TO DO | Status |
| Team Leader Assess reports from each Team Member | Will check Report form Team Member | Done |
| Project Manager will view the status report from the Team Leader | Will view the reports from Team Leader | Done |
| Database Designer will design and manage database for the project. | Database Design & Manage | Done |
| Senior Developer will make sure that the modules are bug free completed on time | Guide the developers and designers to complete tasks on time | Done |
| Programmer will Develop the Module as Per Requirement | Develop Module | Done |
| Designer will Develop an UI as per Customer Requirement | Working UI of Website | Done |
| Testing Manager will Find as much as bug as possible | For Error free Project | Done |

**7.5Agile Test Plan**

A test plan is a detailed document that outlines the test strategy, testing,objectives, resources (manpower, software, hardware) required for testing, test schedule, Test, Estimateand test.

The test plan serves as a blueprint to conduct software testing activities as a defined process which is minutely monitored and controlled by the test manager.

## Importance of Test Plan

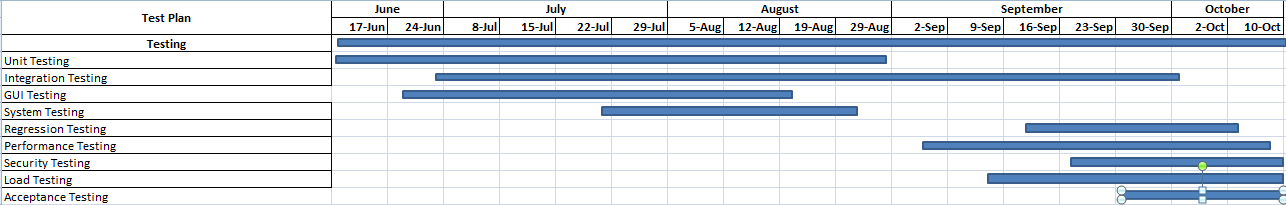
* Test Plan helps us determine the effort needed to validate the quality of the application the under test
* Help people outside the test team such as developers, business managers, customers understand the details of testing.
* Test Plan guides our thinking. It is like a rule book, which needs to be followed.
* Important aspects like test estimation, test scope Test strategy documents Test Plan, so it can be reviewed by Management Team and re-used for other projects.

## How to write a Test Plan

You already know that making a **Test Plan** is the most important task of Test Management Process. Follow the seven steps below to create a test plan as per IEEE 829

1. Analyze the product
2. Design the Test Strategy
3. Define the Test Objectives
4. Define Test Criteria
5. Resource Planning
6. Plan Test Environment
7. Schedule & Estimation
8. Determine Test Deliverables

**AGILE TEST PLA**

****

**8. CONCLUSION**

By this application user can sell different types of product. User can also buy product from other user. User can manage his ads. Users can contact each other with chat box. In this application admin can manage user’s product and also manage user.

E-Commerce site which helps user to buy and sell used products online.

**9. BIBLIOGRAPHY**

**9.1 Online References:**

[www.w3schools.com](http://www.w3schools.com)

[www.tutorialspoint.com](http://www.tutorialspoint.com)

[www.stackoverflow.com](http://www.stackoverflow.com)

[www.guru99.com](http://www.guru99.com)

[www.phptpoint.com](http://www.phptpoint.com)

**9.2 Offline References:**

Julie C Meloni, “Sams Teach Yourself PHP, MYSQL and Apache All in One “4th edition Pearson Education.