Class Diagram(Final)

Class diagram with implementation of different design patterns is introduced and finalized. MVC design pattern in implemented in the class diagram which makes the implementation work in different layout which improves functionality and maintenance in the future. A class diagram with final design for classified web advertising is proposed as below.

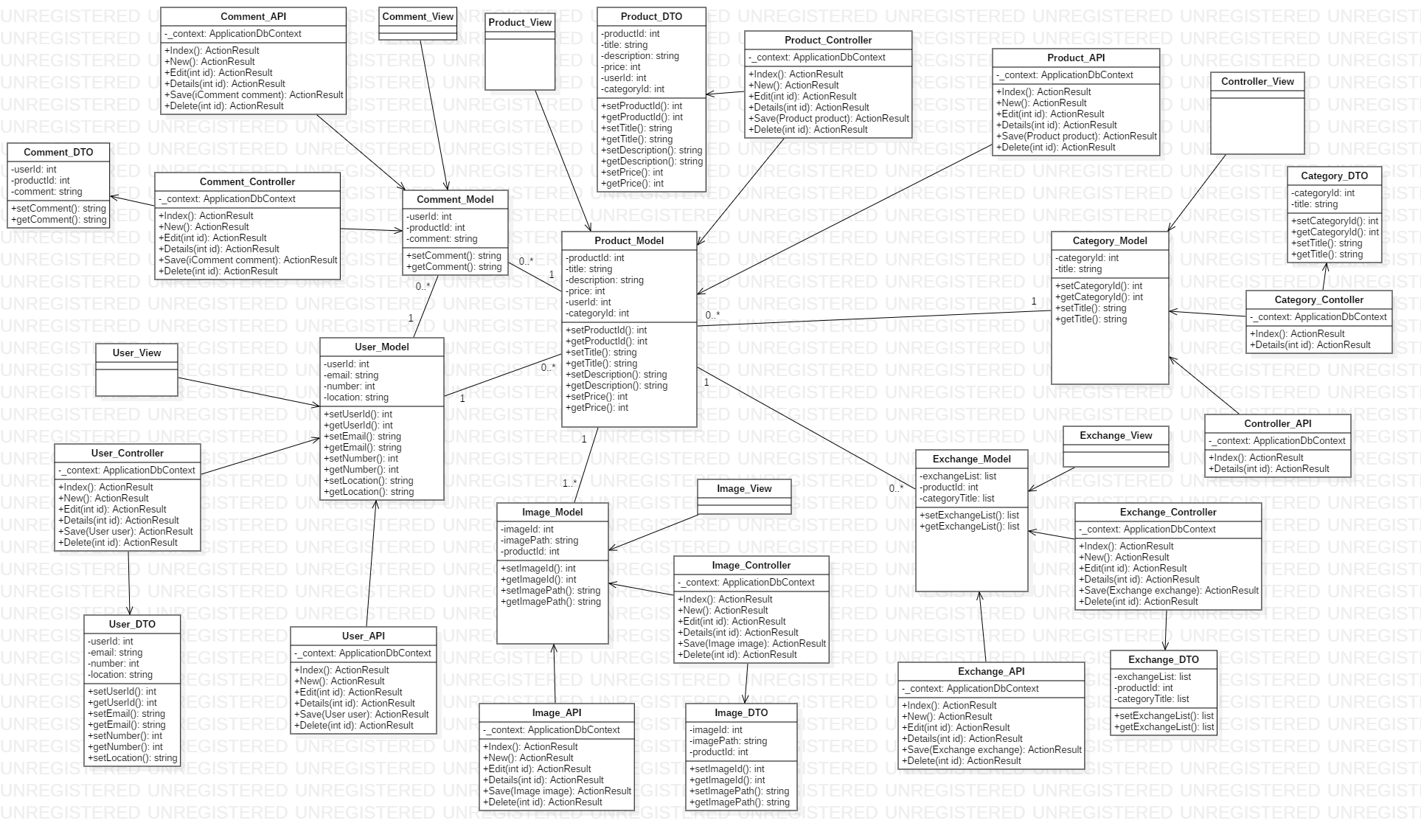


Fig: Finalized class diagram

In the diagram MVC design pattern is introduced. Alongside API of each controller is designed and DTO object class is introduced. Each model has its view and controller with respect to its DTO and API.

Implementation of MVC pattern:

* Httprequests are handled by controller and API
* Frontend design are processed by view
* Model holds certain business logics which makes the architecture of the application independent to each other

Data Flow Diagram

DFD illustrates the flow of the system. It especially justifies how the data is handled, where the data is stored and how it is used as output in the system. DFD of different functionality of the system is designed in order to understand the definition of data which may help find data redundancy in the system and can solve it.

Importance of DFD in our system:

* Justifies how the data is handled
* Explains where the data is stored
* Explains how data is used as output in the system
* Can find data redundancy
* Early problem statement

Registration DFD



Fig: Registration DFD

Login DFD



Fig: Login DFD

Create Ad DFD



Fig: Create Ad DFD

Search DFD

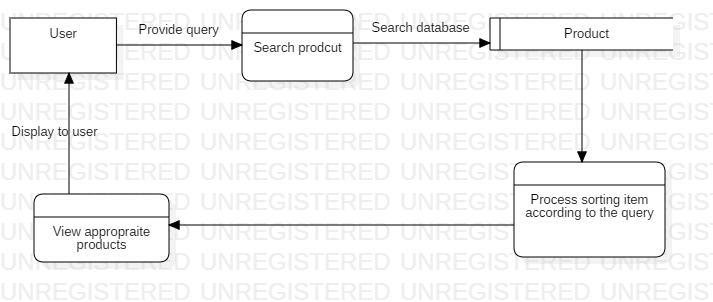


Fig: Search Ad DFD

Activity Diagram

An activity diagram represents a series of actions and the flow of control in a system. Before any implementation of the system, activity diagram illustrates the flow of the code in the system, which classes are involved and how the code is debugged in the real time. Activity diagram helps to understand complex program in easy method and can help debug early bugs.

Use of activity diagram in system:

* Illustrates the flow of the code
* Better understanding of complex program
* Can help debug early bugs
* Model in detail complex activities in a high-level activity Diagram

Activity diagram of some of the functionality in the system is presented.

* Create Ad

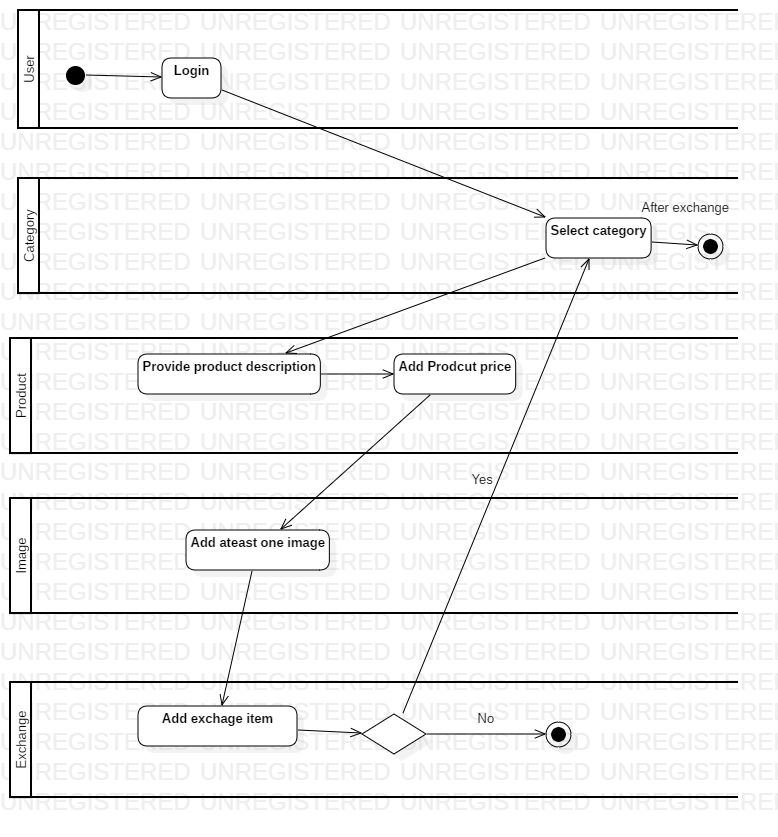


Fig: Create Ad activity diagram

* Search Ad



Fig: Search Ad activity diagram

Sequence Diagram

Sequence diagram illustrates the instance of the object created in the runtime behavior of the system overtime. It defines high level understanding between the users and systems.

Why we need sequence diagram in our system:

* Understand interaction between active objects in the system
* Understand interaction between instances of objects in the system
* Understand the life time of the instances of the object

Registration Sequence Diagram

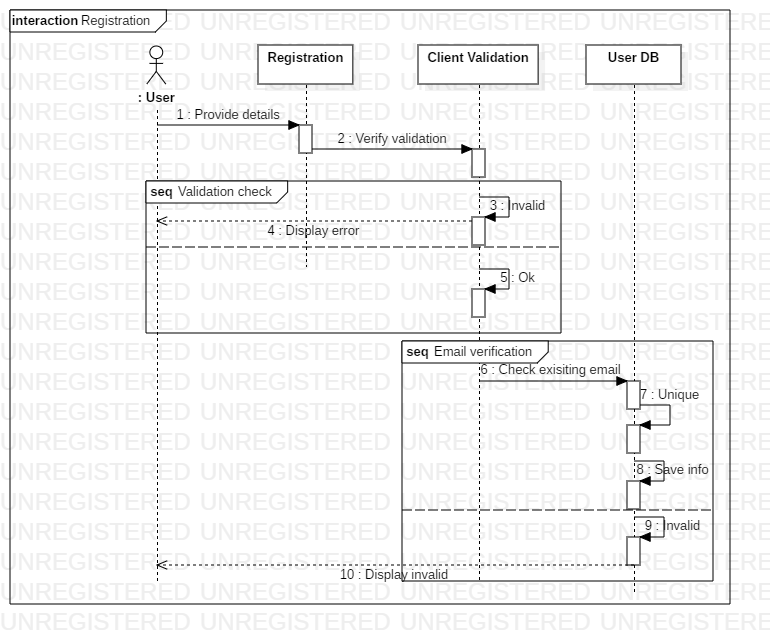


Fig: Registration Sequence diagram

Login Sequence Diagram



Fig: Login Sequence Diagram

Create Ad Sequence Diagram

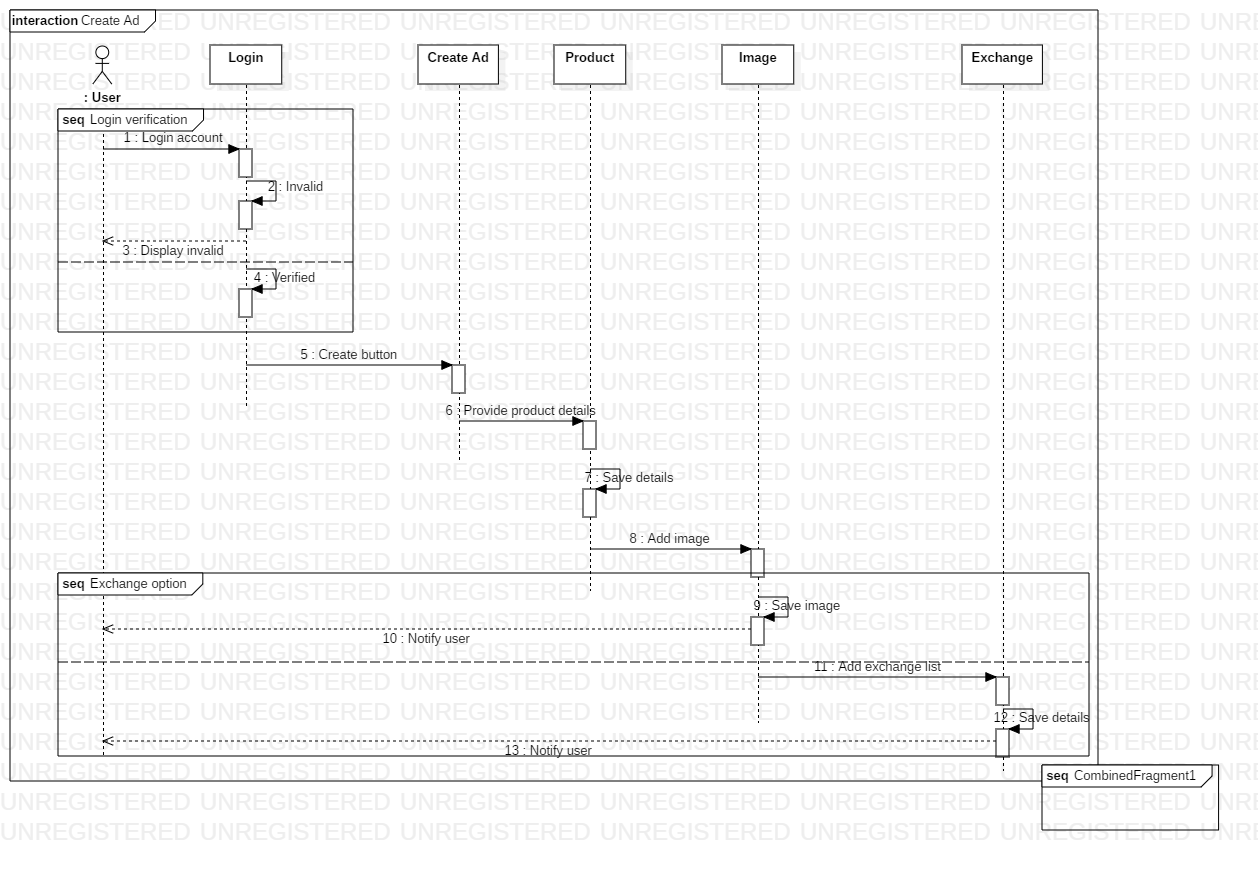


Fig: Create Ad Sequence diagram

Search Sequence Diagram

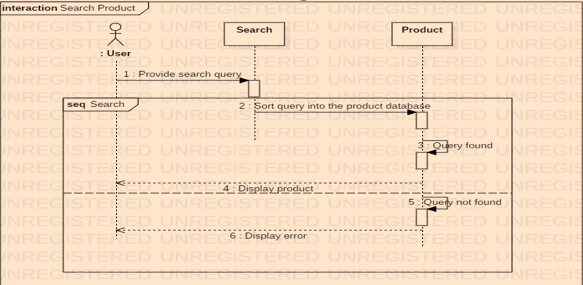


Fig: Search Sequence diagram

ER Diagram

The database design of our system is defined by the help of ER diagram. It shows the relations among tables and column of each table is defined. These tables hold all the data in the back-end server of our system. In our system we have properly normalized the table and made the ER diagram minimum to data redundancy. Finally, we have six table accordingly.

Why we need ER diagram in our design:

* Shows relationship among tables
* Defines column related in each table
* The database system is properly normalized
* There is less data redundancy in the system.
* Can avoid fan trap and chasm trap through diagram.



Fig: ER diagram of Classified Advertising Website

Data Dictionary

Data dictionary defines the structure of the database. It holds the extra definition of data i.e. shows data of the data. What kind of data, the type, character or length is explained in this dictionary in the form of table.

Uses of data dictionary:

* Details about the table such as column, constraints.
* Physical information about the tables.
* Type of data stored in column of the table.
* Constraints such as primary key, foreign key.

Category

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column Name | Data Type | Length | Null | Key | Constraint | Remarks |
| Category\_Id | int | - | No | Pk | Pk\_cat\_id | autoincrement |
| Title | varchar | 50 | No | - | - | - |

User

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column Name | Data Type | Length | Null | Key | Constraint | Remarks |
| User\_Id | int | - | No | Pk | Pk\_user\_id | autoincrement |
| Email | varchar | 50 | No | - | - | - |
| Number | Varchar | 50 | No | - | - | - |
| Location | Varchar | 50 | No | - | - | - |

Product

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column Name | Data Type | Length | Null | Key | Constraint | Remarks |
| Product\_Id | int | - | No | Pk | Pk\_product\_id | autoincrement |
| Title | varchar | 50 | Yes | - | - | - |
| Description | Varchar | 50 | Yes | - | - | - |
| Price | int | - | Yes | - | - | - |
| Category\_Id | Int | - | - | Fk | Fk\_product\_cat\_id | - |
| User\_Id | Int | - | - | Fk | Fk\_product\_user\_id | - |

Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column Name | Data Type | Length | Null | Key | Constraint | Remarks |
| Comment\_Id | int | - | No | Pk | Pk\_comment\_id | autoincrement |
| Comment | varchar | 50 | Yes | - | - | - |
| User\_Id | Int | - | Yes | Fk | Fk\_comment\_user\_id | - |
| Product\_Id | Int | - | Yes | Fk | Fk\_comment\_product\_id | - |

Image

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column Name | Data Type | Length | Null | Key | Constraint | Remarks |
| Image\_Id | int | - | No | Pk | Pk\_user\_id | autoincrement |
| Image\_Path | varchar | 50 | Yes | - | - | - |
| Product\_Id | Int | - | Yes | Fk | Fk\_image\_prodcut\_id | - |

Exchange

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column Name | Data Type | Length | Null | Key | Constraint | Remarks |
| Exchange\_Id | int | - | No | Pk | Pk\_exchange\_id | autoincrement |
| Title | varchar | 50 | Yes | - | - | - |
| Product\_Id | Int | - | Yes | Fk | Fk\_exchange\_product\_id | - |

Prototyping

A prototype is an early release of a product which defines the future designing or interface of our product. All the rough works can be displayed on prototyping so that everyone has understanding in the interface of the product and frequent changes can be made before finalized product is done.

Here we use paper prototyping in our design as it is the easily available, cheapest and quickest way to design and share a prototype.

Reasons of using paper prototype in our design:

* All the rough works can be displayed on prototyping.
* Early design can be shared to the development team.
* Frequent changes can be made easily.
* Paper prototyping is easy, cheap and fast to design.

Home Page

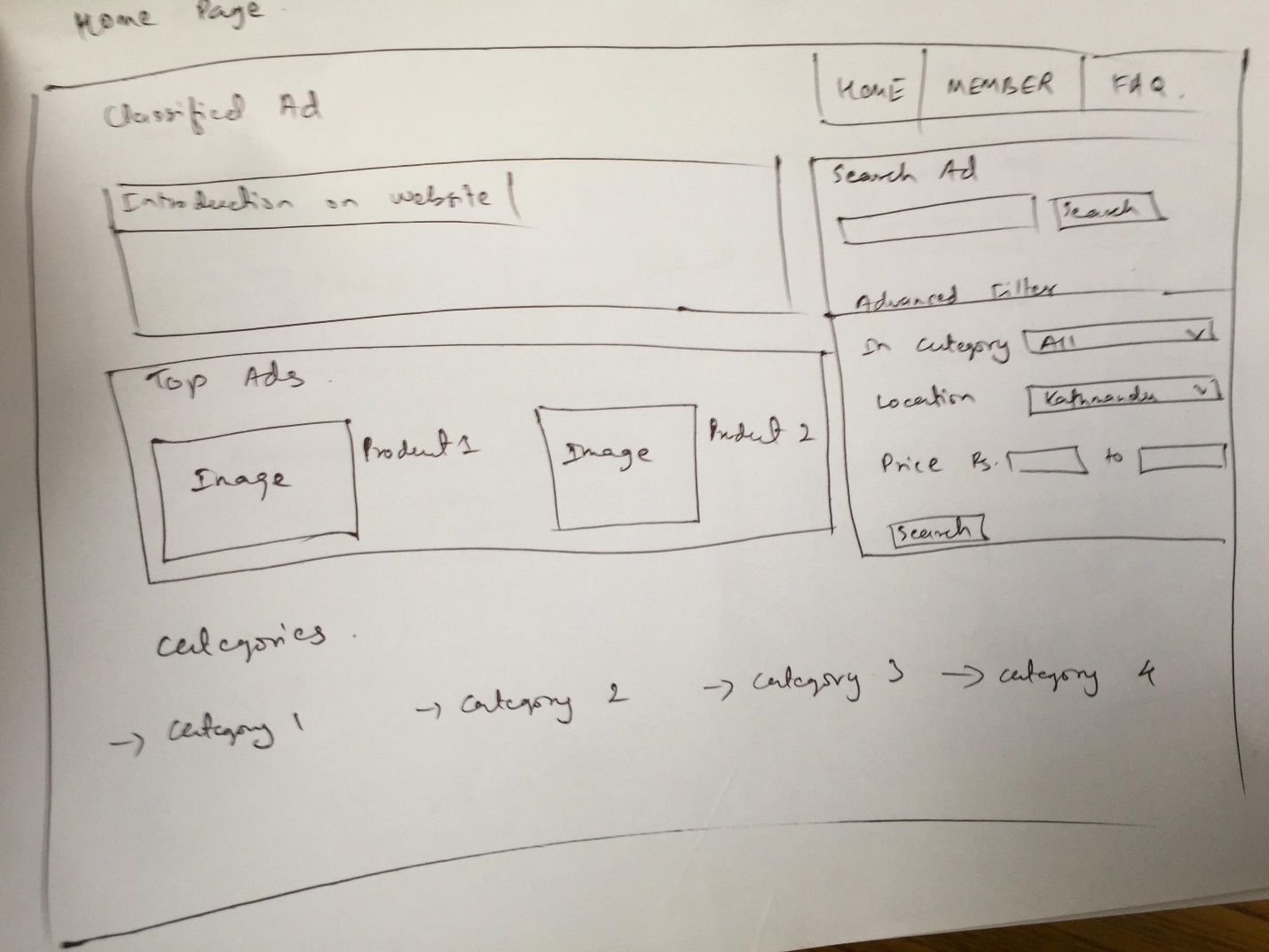


Fig: Home page

Home page consists of a short introduction of the website of what it does and provides services to the users. Top ads are display on every page, and search ad with advanced filter is viewed.

Below the page, categories of the products are displayed through which user can click on any one of their choice and find many products related to that category.

Member Page

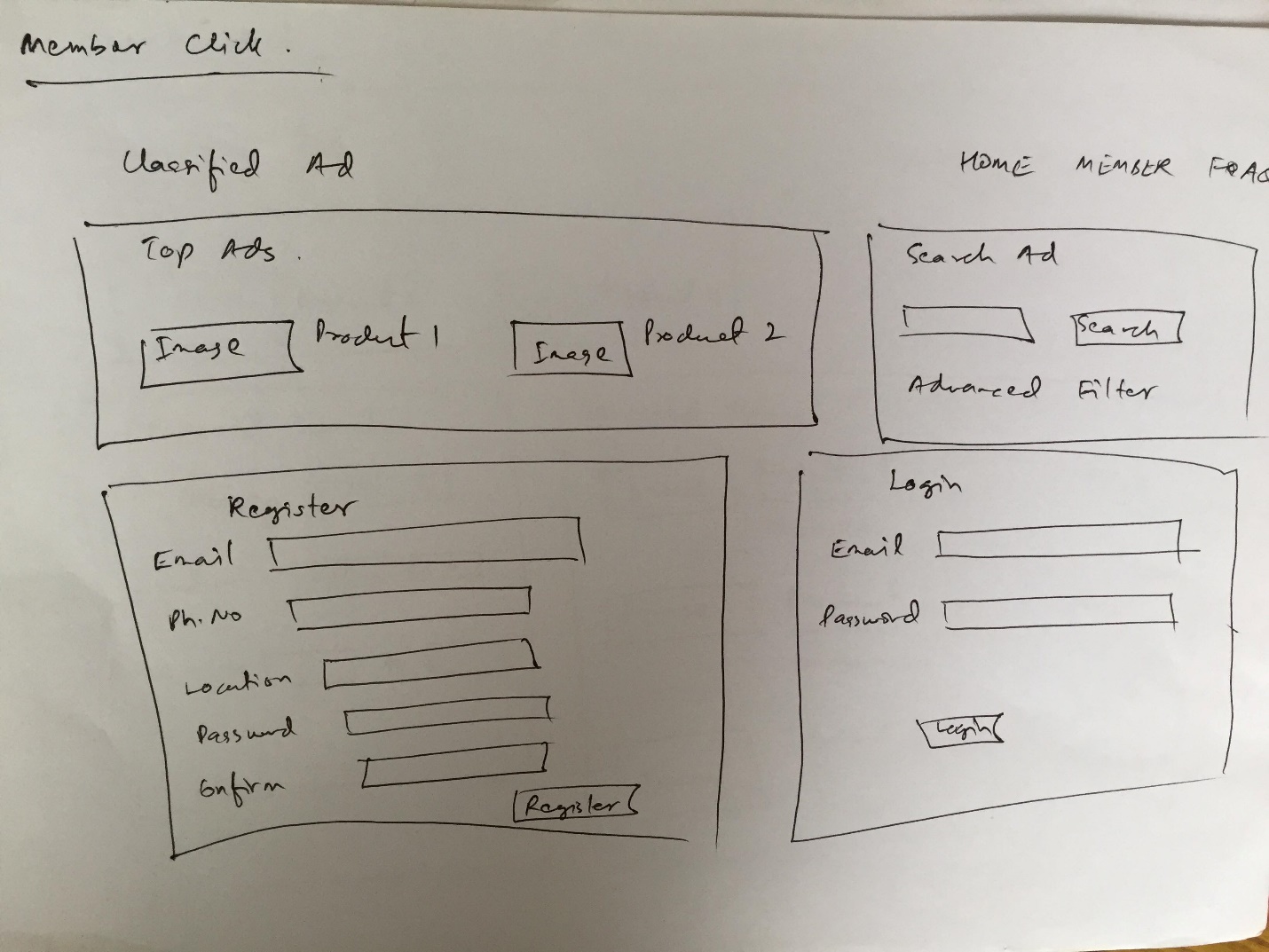


Fig: Member area

Member page consists of field to register user for the system or login into the system if user is already registered. For registration user needs to provide detailed information about their email, number, location and enter a password for their security.

Search Ad page

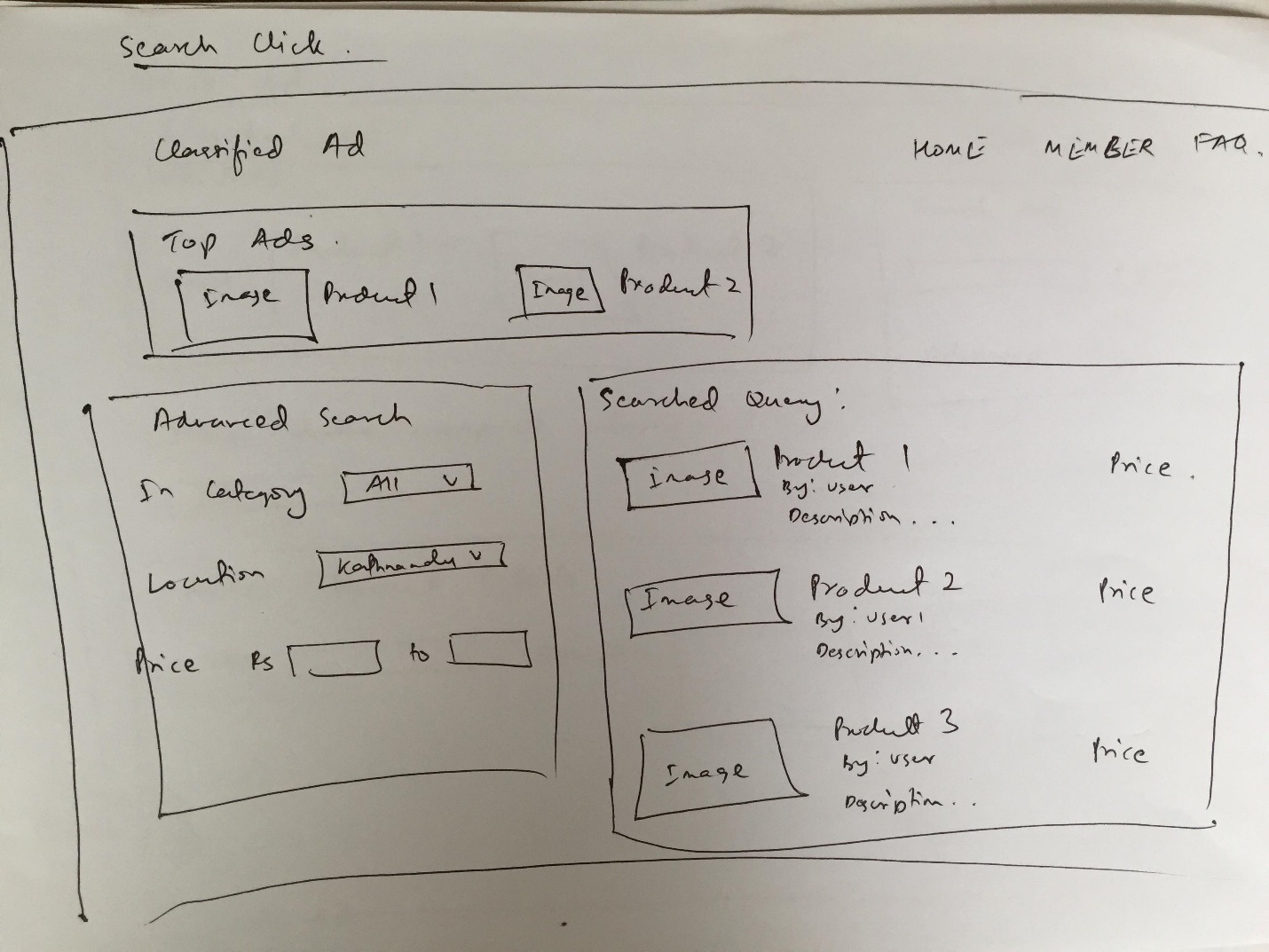


Fig: Search ad page

User can search by typing their query in the search filed button or they can use advanced filter search option to filter out the category, price or location which they find suitable. After their successful query, products related to their query are display vertically with the description, seller and price of the product. User can click to any one of these and find more detailed information about the product and seller.

Create Ad Page

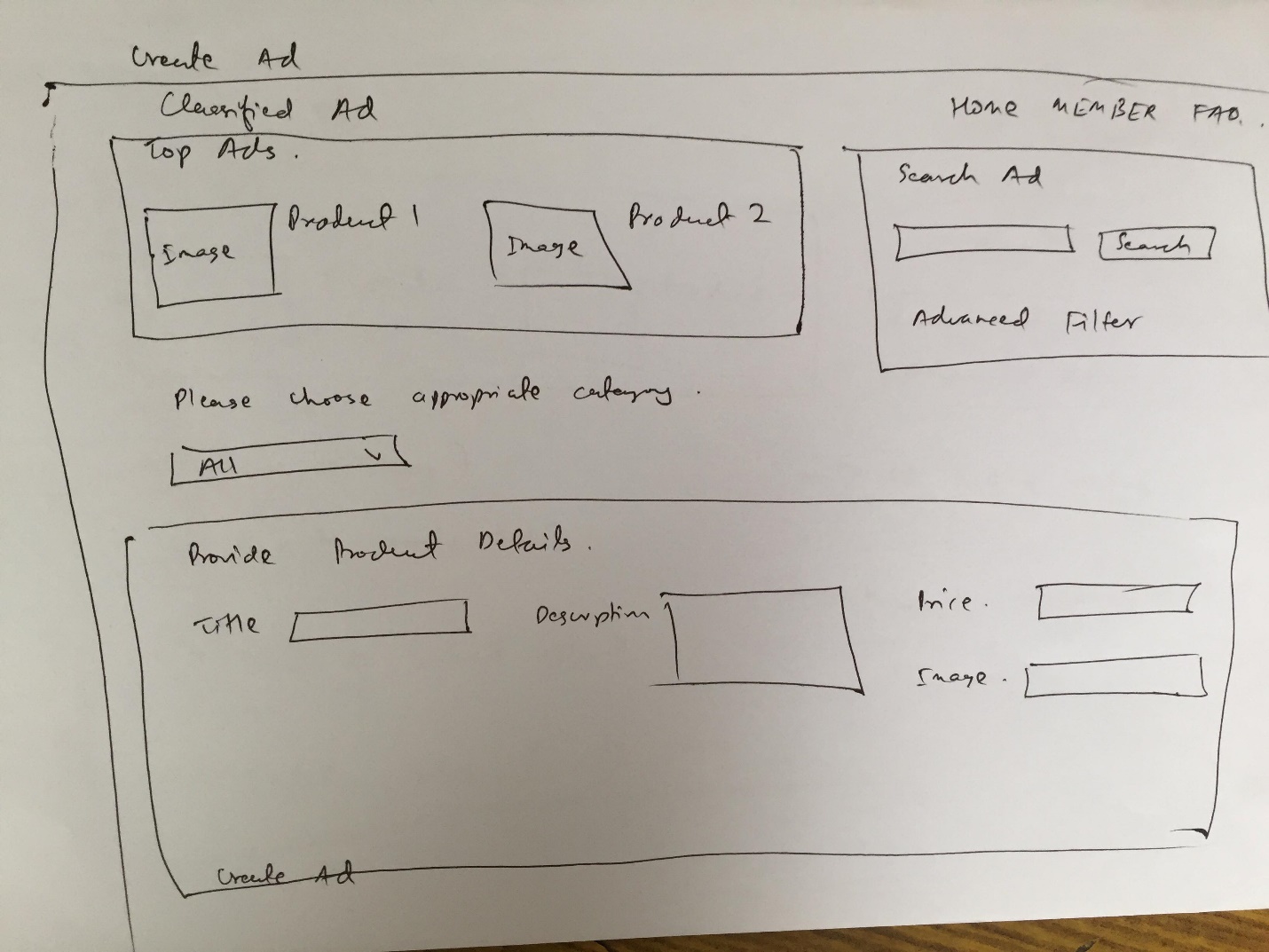


Fig: Create Ad Page

If a user wishes to create an ad, create ad option can be used to enter into this page. First the user needs to select an appropriate category under which their product fall in. After detailed information about the product such as title, description, price and at least one image of the product must be uploaded to create their own ad.

FAQ Page

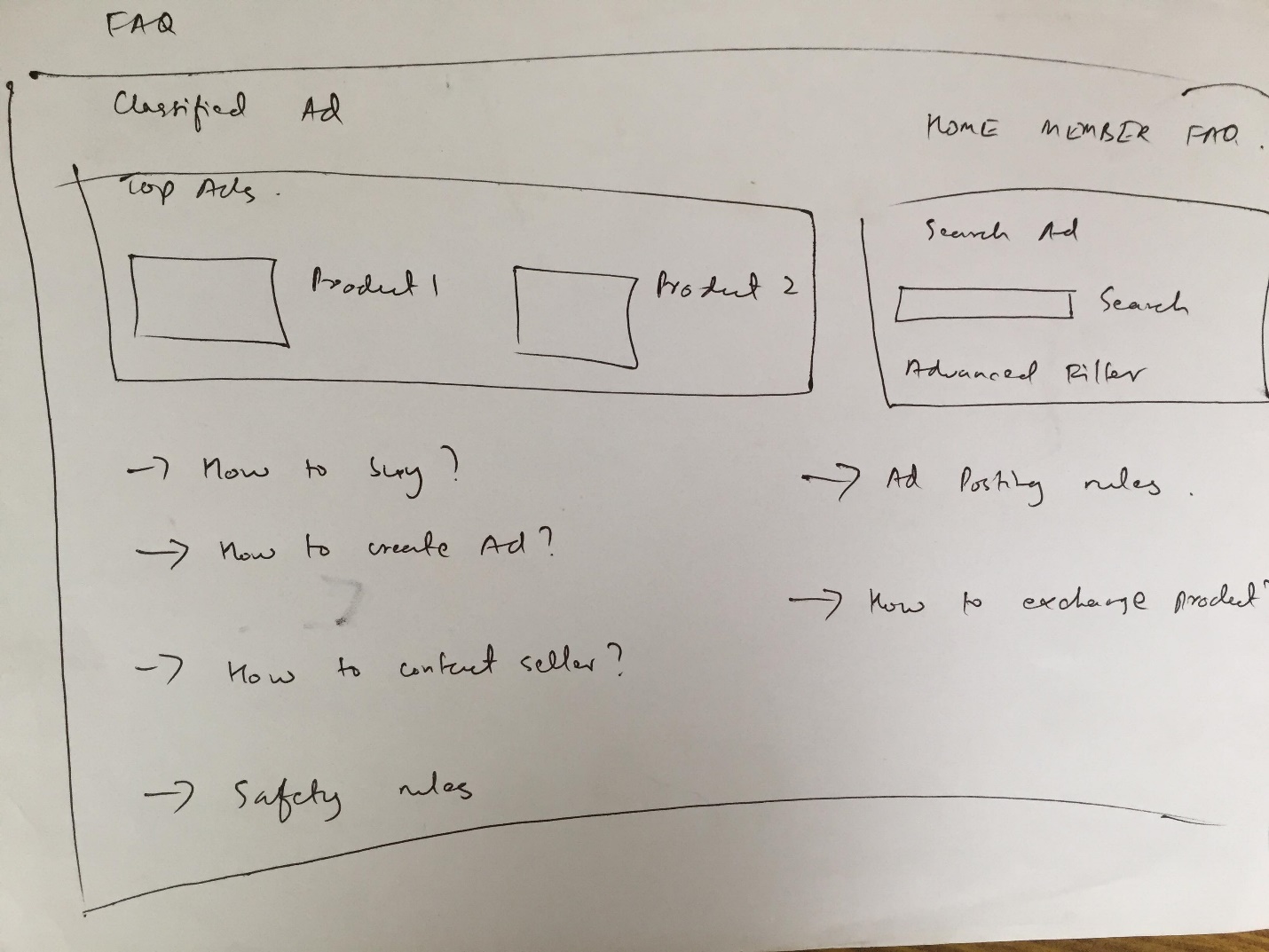


Fig: FAQ page

Users who need help about the website and its process can clear their query by viewing this page. This page consists of frequently asked questions about the services of this system with its corresponding clarified answers.