# Chapter 3 Design

Design is the process of designing all the elements of system architecture, modules and component of the system. All the requirements specification collected from the analysis phase is studied in this phase and then design on the basic of requirement. It helps to determine architecture of the system.

In the design phase, it includes four modelling:

* Structural modeling
* Behavioral modeling
* Database modeling
* UI (user interface) modeling

## 3.1 Structural modeling

Structural modeling is static view of system which helps to emphasizes the structure of the objects and their classifiers, attributes, relationships and operations.

### 3.1.1 Final class diagram

#### 3.1.1.1 Definition of class diagram

Class diagram is static diagram that represent static view of the system, and visualized the relationship between the classes and objects. It is a structural diagram which shows a collection of classes, objects, attributes, associations, interfaces, collaborations and constraints.

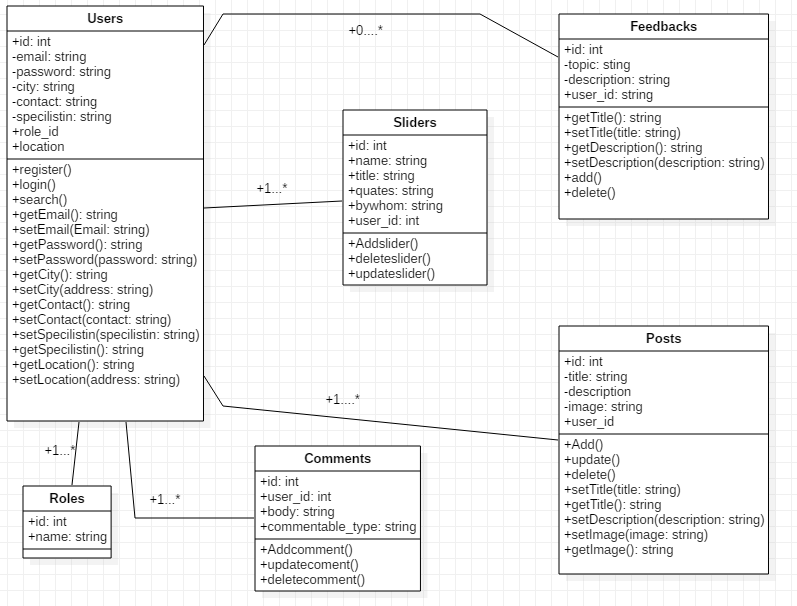
#### 3.1.1.2 Justification for using class diagram

* Provides static view of the application
* Describes the responsibility of application
* Shows the relationship of classes
* Provides flexibility in object-oriented modeling/ structure

#### 3.1.1.3 Notation used in class diagram

|  |  |  |
| --- | --- | --- |
| Symbols | Meaning | Descriptions |
| + | Public | Able to access from outside |
| - | Private | Access modifier which can be accessible only within class |
| # | Protected | Access modifier which can be accessible within and child class |
|  | Aggregation | Show the special relationship of classes with another which specifies whole-part relationship between the aggregated(whole) and component part |
|  | Composition | Connect contained classes. |
|  | Association | It shows simplest type of relationships. |
|  | Generalization | Shows inheritance which specifies child classes inherit the parents class attributes. |
|  | Dependency | Shows one class is dependent to another. |
| \* | Many | Specifies many, generally used to show relationship with other class |

#### 3.1.1.4 Actual Class Diagram



#### 3.1.1.5 Explanation of Class Diagram used

Here I have make main classes named: roles, comments, users, feedbacks, posts. There are actually 3 roles in the system Admin, doctor, normal user. According to the role, different user has different authority.

Class user consist of different attributes like email, password, role-id, city, location, specialist-in, etc. it has functionality like register, delete user, update profile, search.

Class roles consist of two attributes name and id.

Class comments consist of four attributes are id, user-id, body and comment-able-type. With functions Add-comment, update-comment and delete-comment.

Class feedback has attributes id, topic, description and user-id with functionality like add, delete, and view feedback.

Class posts consists of id, title, description, user-id attributes and add, update, delete post functionality.

### 3.1.2 Flow chart

Flow chart is a visualization of the workflow or process or representation of an algorithm, step by step approach for solving the task. It traces the information and work through it, and even highlight the process and decision making of system. So, it shows overall structure of system process.

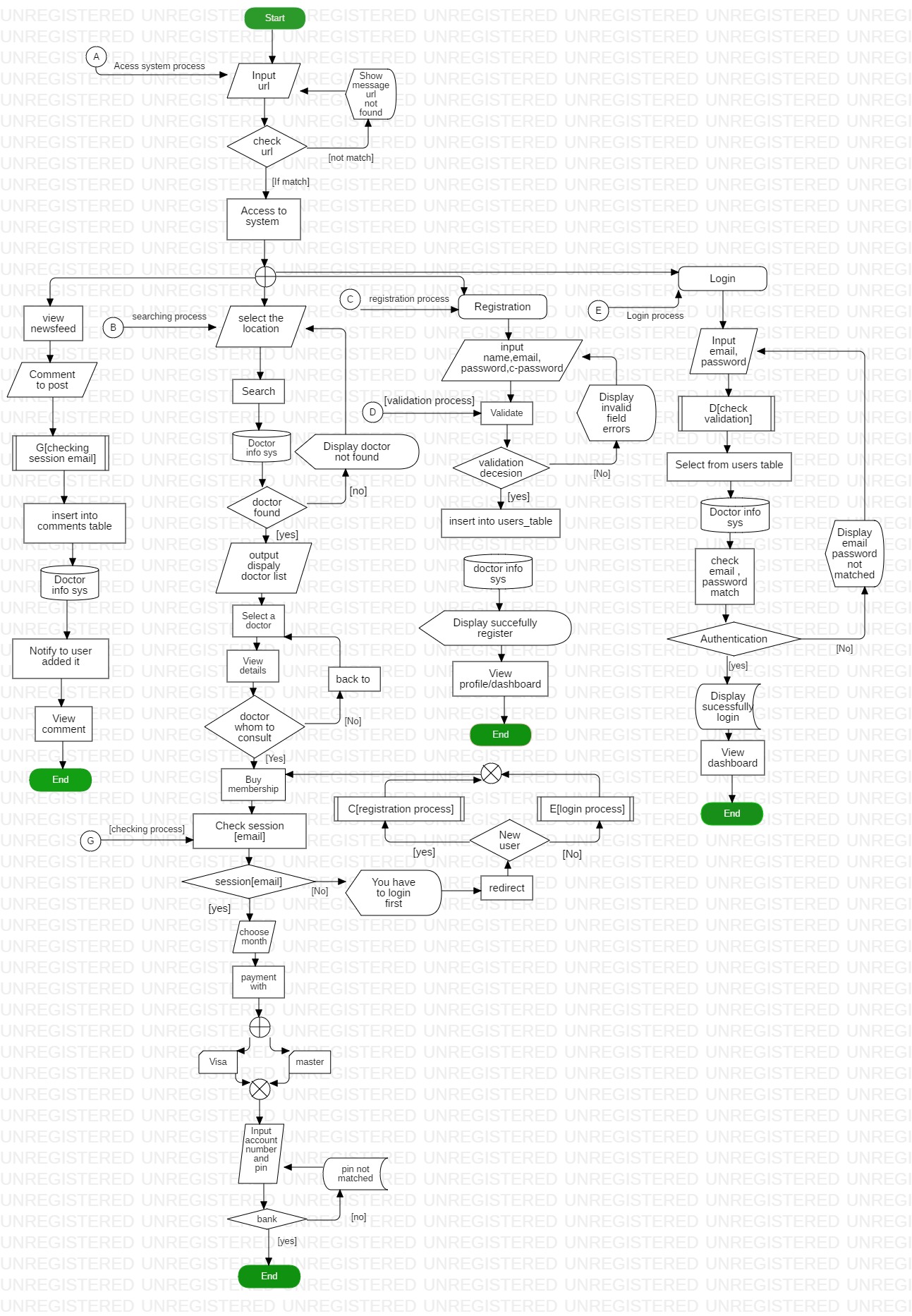
#### 3.1.2.1 Justification for Flowchart diagram

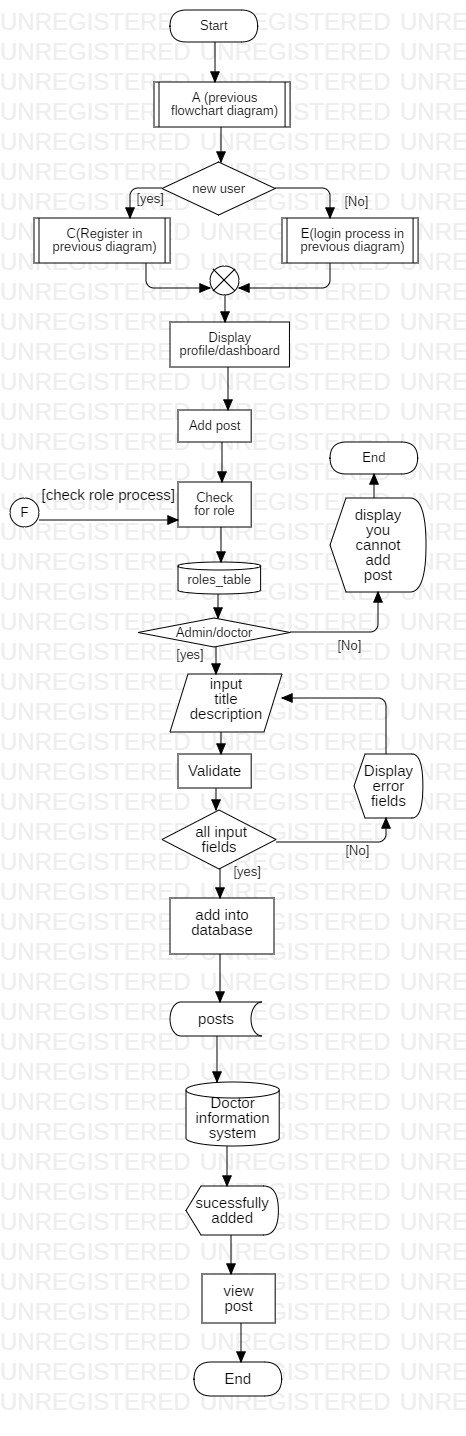
* Helps to identify different element of the process
* Helps to understand various interrelationship among steps
* Helps to gather data and information of the process
* Helps to improve the process

#### 3.1.2.2 Symbols and notations used while drawing the flow diagram

|  |  |  |
| --- | --- | --- |
| Notations | Meaning | Description |
|  | Flowline | It shows the direction of the process flow |
|  | Terminal | It shows start and end point of the process. |
|  | Process | Rectangle in shape represent the set of operations or process that changes the value, form |
|  | Decision | used in position of conditional statement like true or false, yes or no, it is diamond in shape |
|  | Input/ Output | Represent as parallelogram, used to display input and output data process |
|  | On page connector | Represent as small circle which has letter inside, shows the matching jumping point. |
|  | Off-page connector | Represent as pentagon, used for when the connector is another page |
|  | Pre-define process | Represented as rectangle which has also two vertical line inside, it is marked up process or set of process that are defined elsewhere |
|  | Single document | Represented by rectangle with wavy base, process step for producing the document |
|  | Multi-document | Represented as multiple rectangle with wavy bases which are stacked, process step for producing the multiple document |
|  | Data-store | Used for any process that stores data |
|  | Datafile/ database | Represented as cylinder |
|  | Manual operation | Represented as trapezoid shape, operation to process which can only be manual |
|  | Preparation or Initialization | Represented as elongated hexagon, used for the setting a route |
|  | Summing junction | Represented as circle with cross inside, shows when multiple branches converge into single process |
|  | Collate | Represents a process step which requires managing data, information or materials accordingly in standard format |
|  | Manual input | Keyboard like quadrilateral top irregularly slopping up from left to right corner, shows the process where the operator is promoted for the information that must be manually inputted |
|  | Sort | Represented as two triangle whose base are merge top corner seems opposite to each other, used to show the shorting of data, information, materials |
|  | Annotation | Represented as open rectangle with a solid or dash line connecting to it, indicating additional information about a step in the system |

#### 3.1.2.3 Flowchart diagram





#### 3.1.2.4 Explanation of the flowchart diagram

Here in those two diagrams is shows how the user input data how it converts into output? how the data changes with the process? what are the activity perform by the users? It shows how the system validate the given input of user and if not matched then send back to login form showing the field does not match requirement error message, if matched send to database to check whether the user exit, if exit redirect dashboard showing message successfully login and if not exit send back to login form displaying error email , password does not matched . Similarly, while creating the new user we can send input and system validate similar as login then send to database to check whether user having same email exit or not if exit send back to register page and display error email address already exit. If not exit then register new user and send to dashboard and display you are successful to register to system.

## 3.2 Behavioral modeling

It describes the dynamic nature of the system. It is used to model interaction between objects. Activity diagram and sequence diagram are the example of the behavioral modeling. It is time dependent. It shows run time activity process.

### 3.2.1 Activity Diagram

Activity diagram is the behavioral modeling diagram which represent dynamic view of the system and describes how the activity are coordinated. It represents the workflow of system.

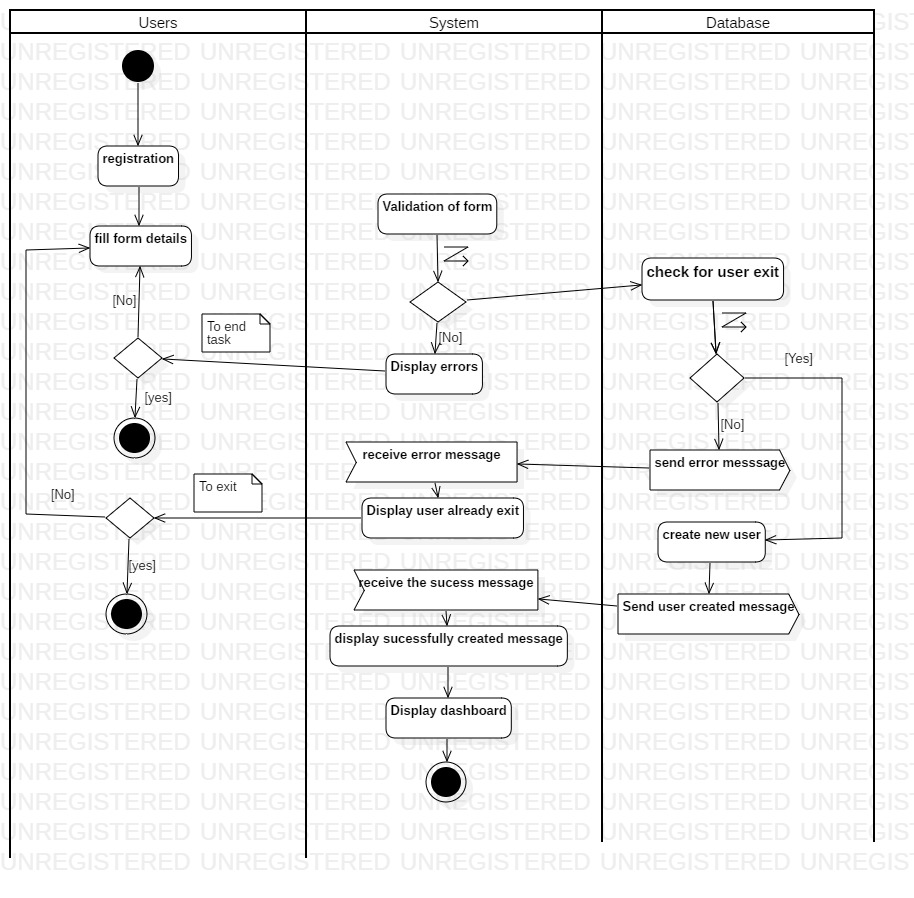
#### 3.2.1.1 Justification for the use of Activity Diagram

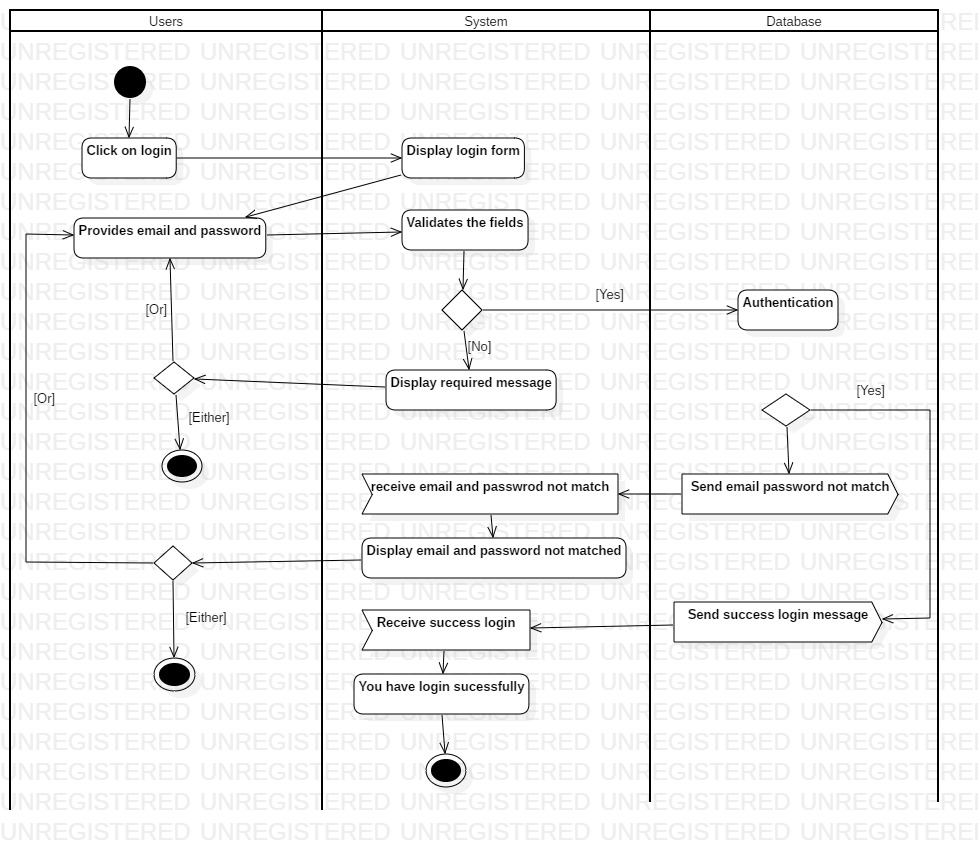
* Used to visualized the activity flow of the system
* Used to describe the sequences from one activity to another
* Shows the parallel, concurrent and branched flow of the system

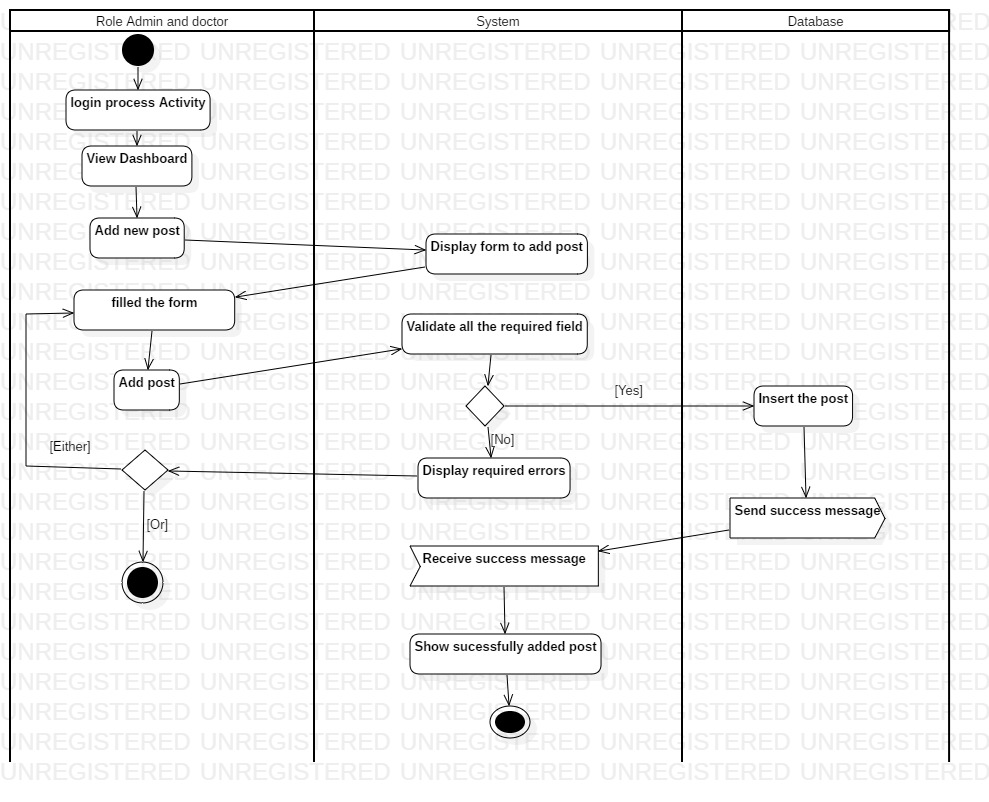
#### 3.2.1.2 Symbol or notation used in Activity diagram

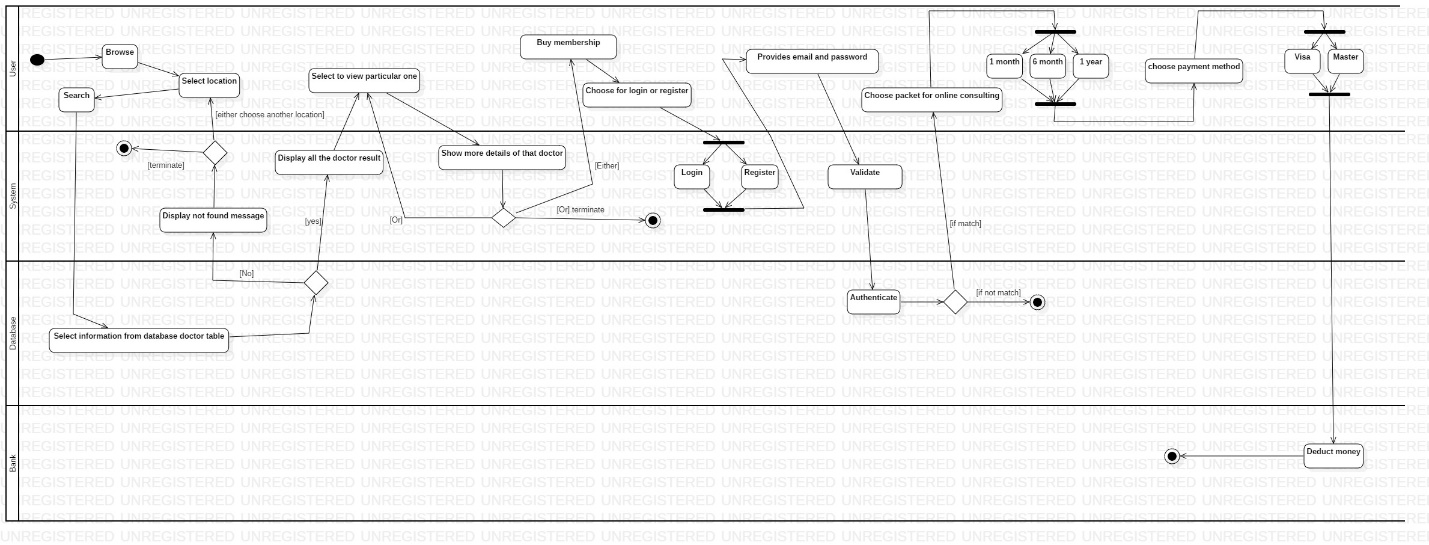
|  |  |  |
| --- | --- | --- |
| Symbol | Name | Description |
|  | Start | Represent the beginning of a process or workflow in an activity diagram, represented as a dark circle |
|  | End | Represent the completion/ end of the process, represented by a circle containing a dark circle in middle |
|  | Activity/ process | Main component, indicates the activities that make up a modeled process |
|  | The Connector | Represented by arrow line, shows the control flow or directional flow of the activity |
|  | Join or Synchronization bar | Represented by thick horizontal line or vertical line, combines two concurrent activities and re-introduce them to flow where only one activity occurs at a time |
|  | The decision | Represented by diamond shape, represent the branching of various flow with a symbol acting as frame or container |
|  | The Fork | Symbolized with multiple arrowed line from join, splits a single activity flow into two concurrent activities |
|  | Note | Allows the diagram creator to communicate additional message that do not fit within the diagram itself |
|  | Send signal | Means that a signal is being sent to a receive activity |
|  | Receive signal | Demonstrate the acceptance of an event after event is received |
|  | The flow final | Shows the ending point of a process flow |
|  | Option Loop | Allows to model a repetitive sequence |
| **H** | Shallow history pseudo-state | Represents a transition that invokes the last active state |
|  | Swimlane | Can be horizontal or vertical, represent the activity objects |
|  | Merge | Represent by diamond shape having an arrow ahead, two or more non concurrent |

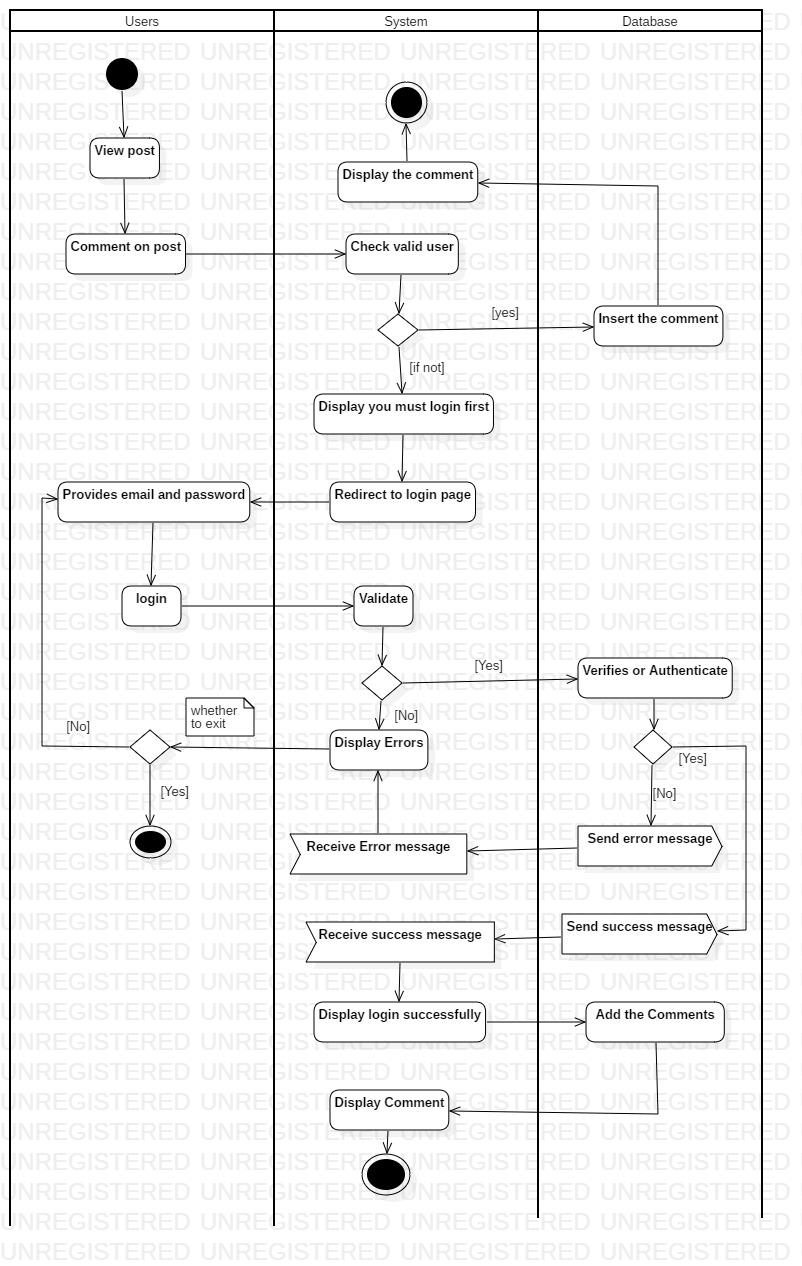
#### 3.2.1.3 Activity Diagrams

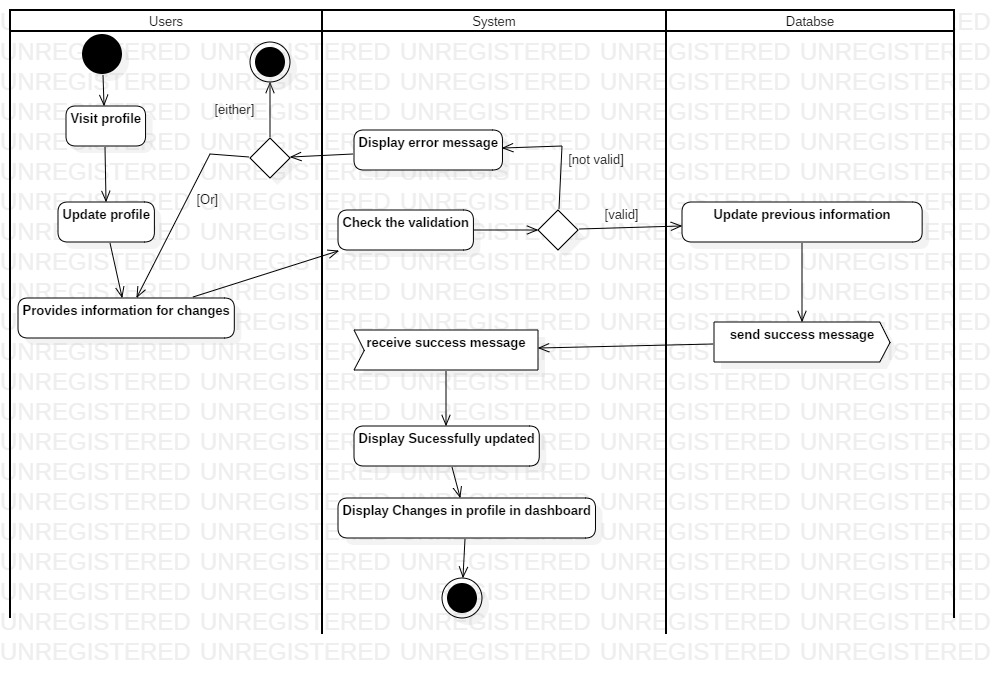












#### 3.2.1.4 Explanation of diagram

Here it show the interaction of functionality with the user activity.

### 3.2.2 Sequence Diagram

It is the behavioral diagram which shows how the processes operate with one another and in what order. It represents the object interaction arranged in time sequence. It is just like the event trace diagram. It describes the flow of the messages, actions or process between objects.

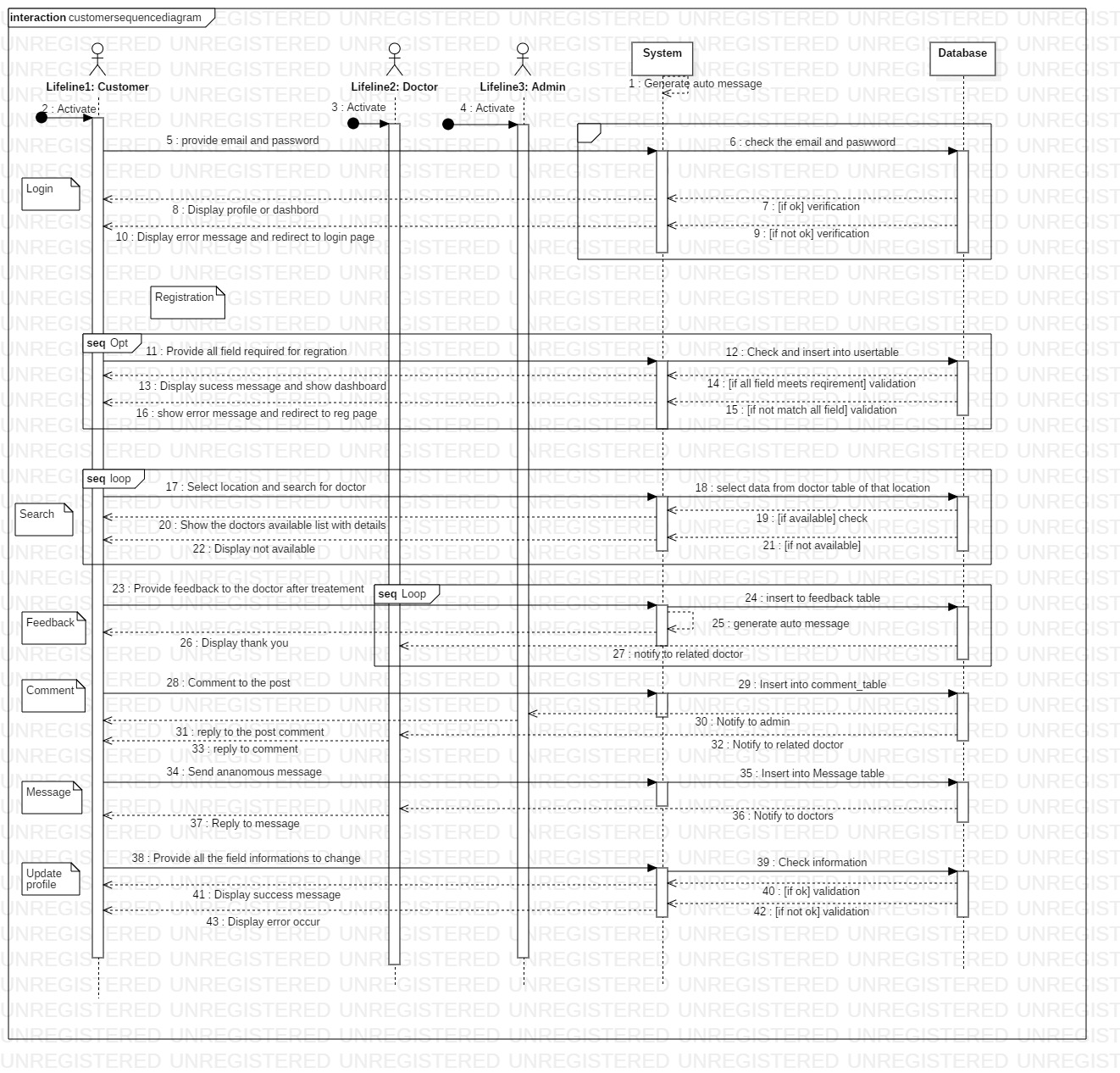
#### 3.2.2.1 Justification for using Sequence Diagram

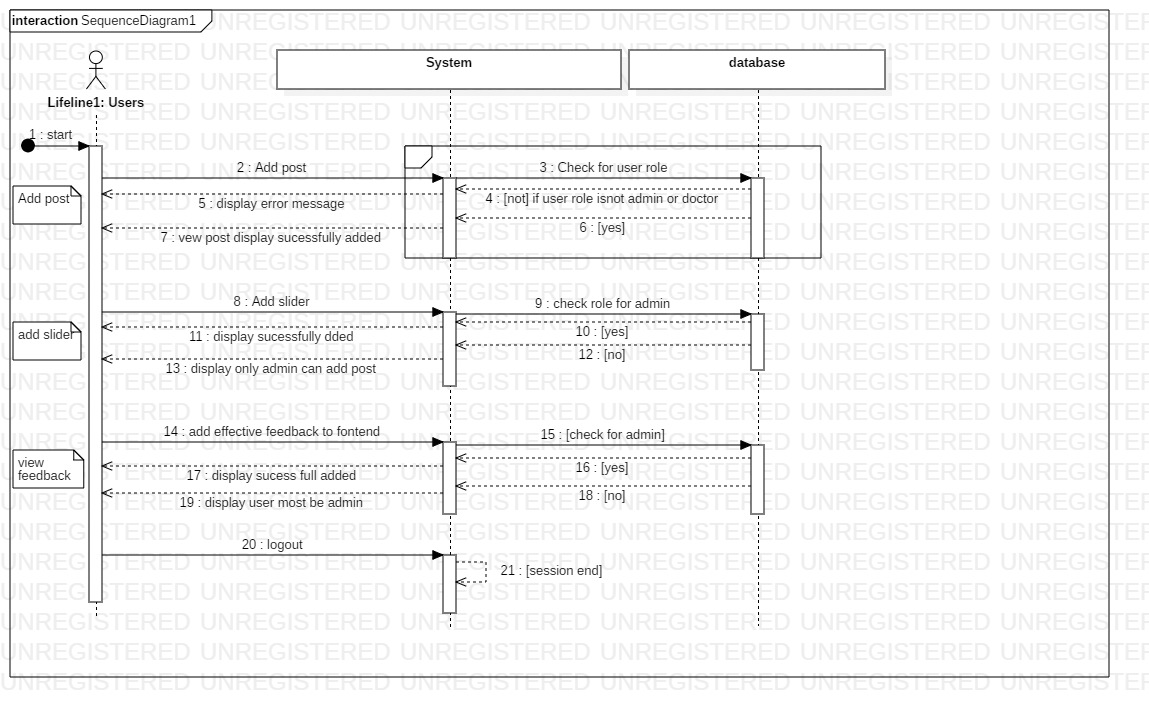
* Shows the logical flow of the system
* Shows the set of events and objects interaction
* While implementing the codes it makes lot easier to understand and implement the logic
* Makes other person to understand the system event interaction with objects and even the actors

#### 3.2.2.2 Symbol used in Sequence diagram

|  |  |  |
| --- | --- | --- |
| Symbol | Title | Description |
|  | Object | Represented as rectangular box at top with object name inside and followed by vertical line or dotted line, represent the different objects interact with each other |
|  | Actor | Represent the external user interact with system |
|  | Boundary | Indicates system boundary |
|  | Entity | Represents the system data |
|  | Control | Serves as intermediate between boundaries and entities, schedules and manage interaction between them |
|  | Activation bars | It is like box placed at lifeline, indicate an object is active |
|  | Synchronous | Dark arrow head like structure, used to send the message |
| **<**- - - - - - - - - - - - - | Receive message | Used to reply message |
| < | Asynchronous | Used to send the message, does not wait for receiver to process the message |
| Loop | Loop | Used when there looping condition |
| Alt | Alternative | Used when event have some condition alternative to first one |
| opt | Optional | Used when the event process has optional statement to choose either one of them |
|  | Comment or note | Used to comment the notes |

#### 3.2.2.3 Actual Sequence diagrams





#### 3.2.2.4 Explanation of Sequence diagram used

Users: role -normal user/ customer can able to search, comment to post and reply to comment, login, register, update profile, provide feedback

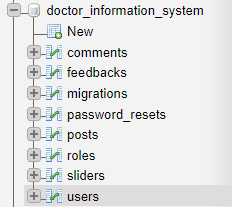
Users: role -doctor can able to add the new post, update their post, update their profile, reply to message, reply to post comment, login, register

Users: role -admin Can able to view users list able delete them, login, add post, update post, add sliders, update slider, delete sliders, delete post, reply to comment, maintain to view the best review

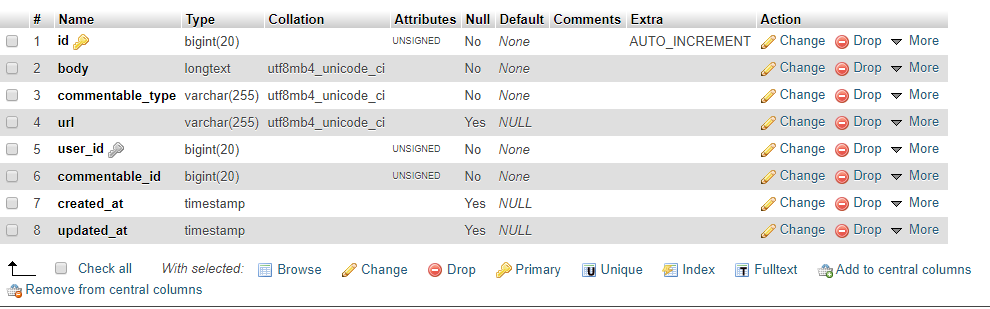
## 3.3 Database modeling

It deals with database design and modeling.

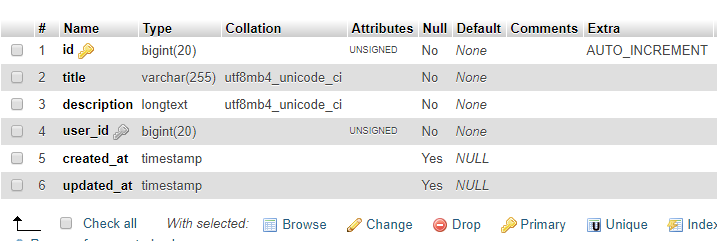
### 3.3.1 Data Dictionary



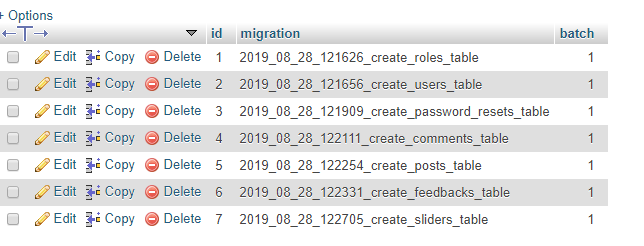
Comments table



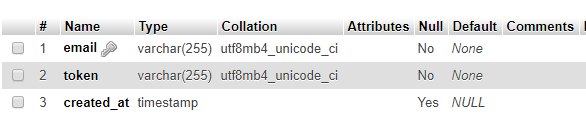
Feedbacks table



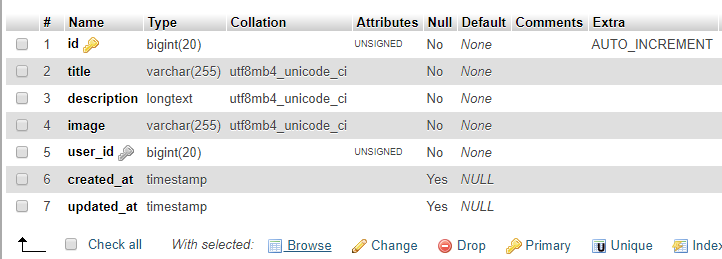
Migrations table



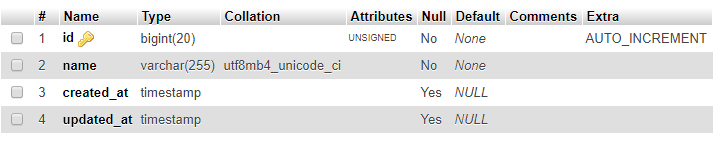
password\_resets table



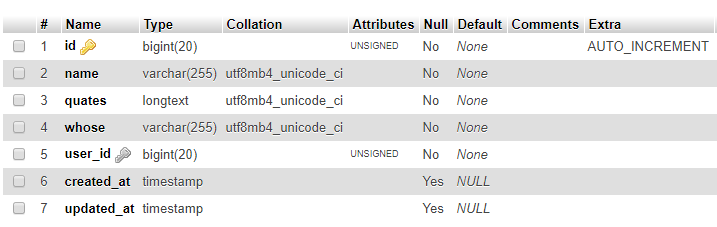
posts table



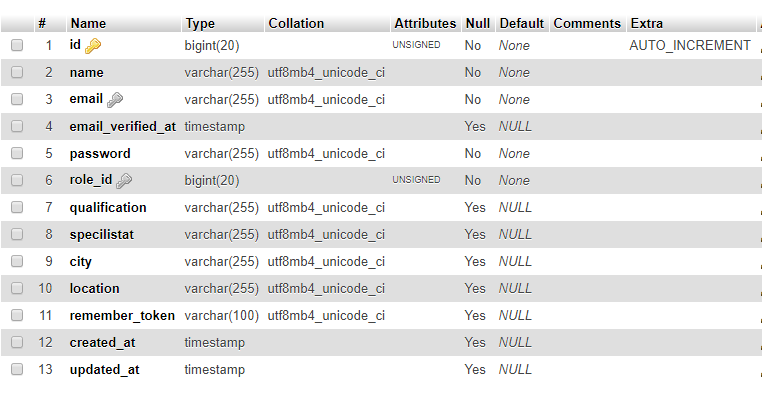
roles table



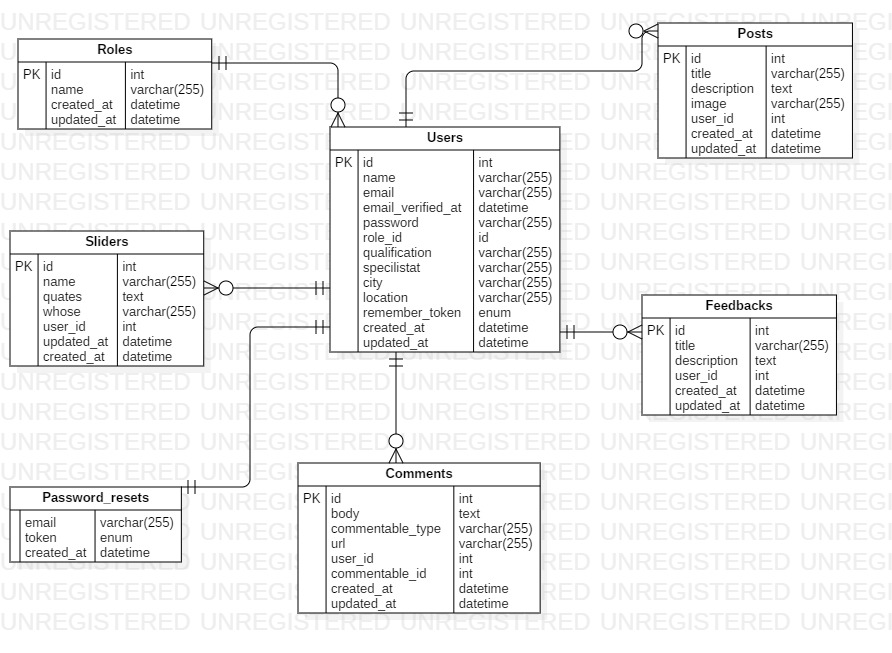
sliders table



Users table



### 3.3.2 Entity Relationship Diagram



3.4 User Interface Modeling

### 3.4.1 Prototypes

I have used digital prototyping tool named balsamiq for making the prototype. Digital Prototype is creating the sample how the system will look like using some digital tool (software). It is part of UI modeling.

Reason for choosing balsamiq

* Open source
* Easy to use
* Available
* Reliable
* Provides icons quicker from the search

