**SOFTWARE REQUIREMENT SPECIFICATION**

**Of**

***“SNIPE COMMUNITY”***

**Team Size - 12**

**Software Requirements Specification**

**Document**

**Version: 1.0.0**

**Date: (-11-2017)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Document** | **Author** | **Reviewed By** | **Version** |
| SRS |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **TABLE OF CONTENTS** | | |
| **1.** | **INTRODUCTION** | |
|  | 1.1 Purpose | 3 |
|  | 1.2 Project scope | 3 |
|  | 1.3 Definition | 3 |
|  | 1.4 References | 3 |
|  | 1.5 Technologies | 3 |
|  | 1.6 Overview | 4 |
| **2** | **OVERALL DESCRIPTION** |  |
|  | 2.1 Project Perspective | 4 |
|  | 2.2 Use-Case Model survey | 4-6 |
|  | 2.3 Use case diagram | 6 |
|  |  | 6-7 |
| **3** | **REQUIREMENTS** |  |
|  | 3.1 Use case specification | 7-8 |
|  | 3.2 External interface requirements | 8-10 |
|  | 3.3 User characteristics | 10 |
|  | 3.4 Usability | 10-11 |
|  | 3.5 Reliability | 11 |
|  | 3.6 Performance | 11 |
|  |  |  |
|  |  |  |
| **4** | **USER INTERFACES** |  |
|  | * 1. Home Page | 12 |
|  | * 1. Login and signup page | 13 |
|  | * 1. Question form | 14 |
|  | * 1. Subscribe form | 15 |
| **5** | **INTERFACES** |  |
|  | * 1. User interface | 16 |
|  | * 1. Hardware interfaces | 16 |
|  | * 1. Communication interface | 16 |

1. **INTRODUCTION**
   1. **Purpose:**

An online community can act as an information system where members can post, comment on discussions, give advice or collaborate.

* 1. **Project Scope:**

It is mainly giving solutions for who having the questions and queries in development and coding. Having the details of snipe employs x employs.

* 1. **Definition:**

SNIPE COMMUNITY It features a variety of categories and services that aim to provide users with the necessary tool of effective solutions for their questions and blogs of respective languages.

* 1. **References:**

This web application has been prepared on the basis of discussion with Team members and also taken information from following websites,

Google.com

Wikipedia.org

* 1. **Overview:**

This SRS document contains many use case diagram and description about all use case diagrams have been included in this document. It also gives the functional requirement.

1. **SOFTWARE REQUIREMENT SPECIFICATION**

**User Interface:**

* IDE Tool: Atom 1.22.1
* Languages: HTML5 ,CSS 3 ,Javascript,JQuery
* Framework: Angular JS ,Bootstrap 3

**Back End:**

* Languages: JAVA 1.8
* IDE Tool: Eclipse
* Framework: hibernate, Springs, Spring boot, REST client.
* Server: Apachecxf

**Database:**

* IDE Tool: pgAdmin4
* Database: PostgresSql 9

**Hardware Requirements**

* Processor : Intel Pentium III
* Speed : 2.13 GHz
* Memory : 4GB
* Hard Disk : 40GB
* Peripherals : Keyboard, mouse,

1. **OVERALL DESCRIPTIONS**
   1. **Project Perspectives**

* Admin will create all employee registrations
* Admin will be having default user name and password
* Once admin logged in, he can view registration page
* Password field is a unique in which it is given for admin of type varchar 10 bytes.
* It contains the username of a admin of type varchar 10 bytes.
* Mention Whether he is existing employee or old employee
* If he is existing employee, Admin give access to modify/design/develop the webpage
* If employee left, Admin remove his access rights
  1. **Use-Case Model Survey**

The use case diagrams contain three actor Admin, Employee and User, and use cases Register, Authentication, Login, Upload questions of related topics, View posts.

**Use Cases:**

1. **Register:**

Here admin is adding employee of snipe community, by filling details of the employee in the registration form.

1. **Signup:**

User needs to sign up by filling some mandatory fields.

1. **Authenticate:**

Email and phone numbers of user are authenticated before login.

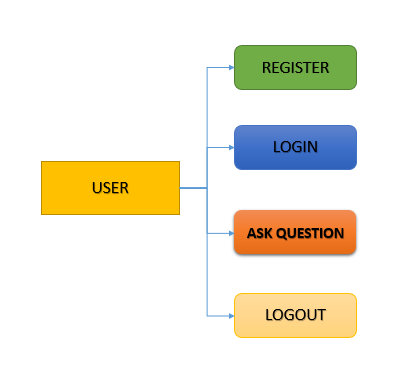
1. **Login:**

User and employee should login here.

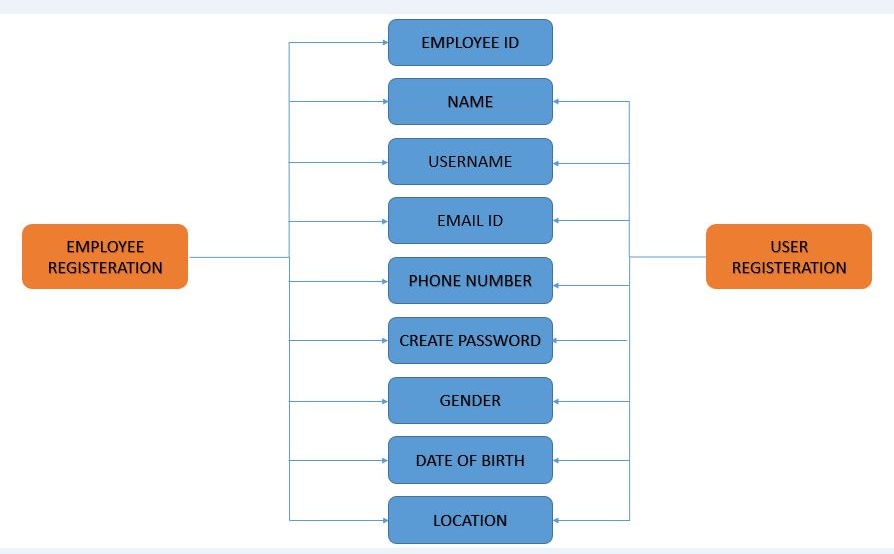
1. **Asking questions:**

`User can post questions.

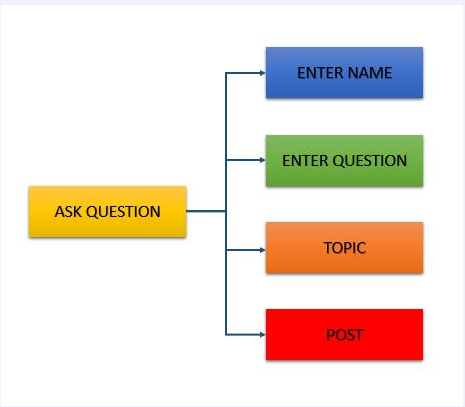
1. **SYSTEM DESIGN**
   1. **Use case diagrams:**
2. **USER**



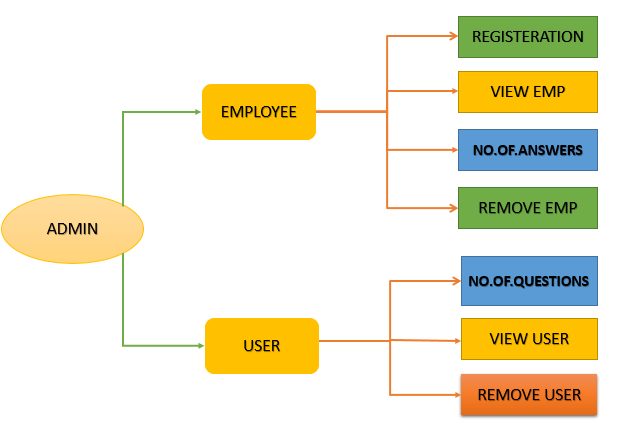
1. **REGISTRATION**



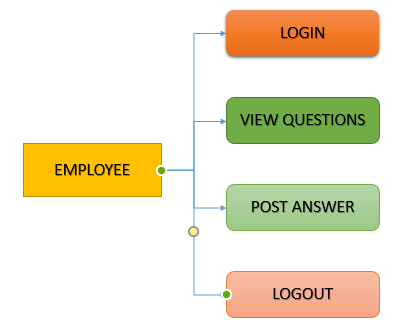
1. **Ask question**



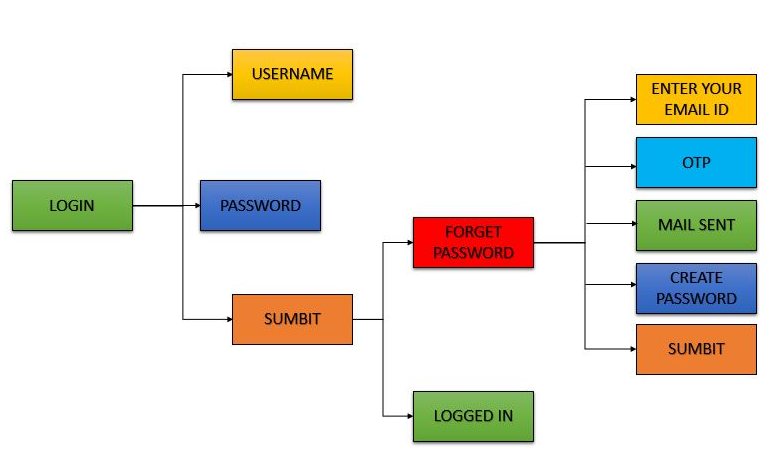
1. **Admin**



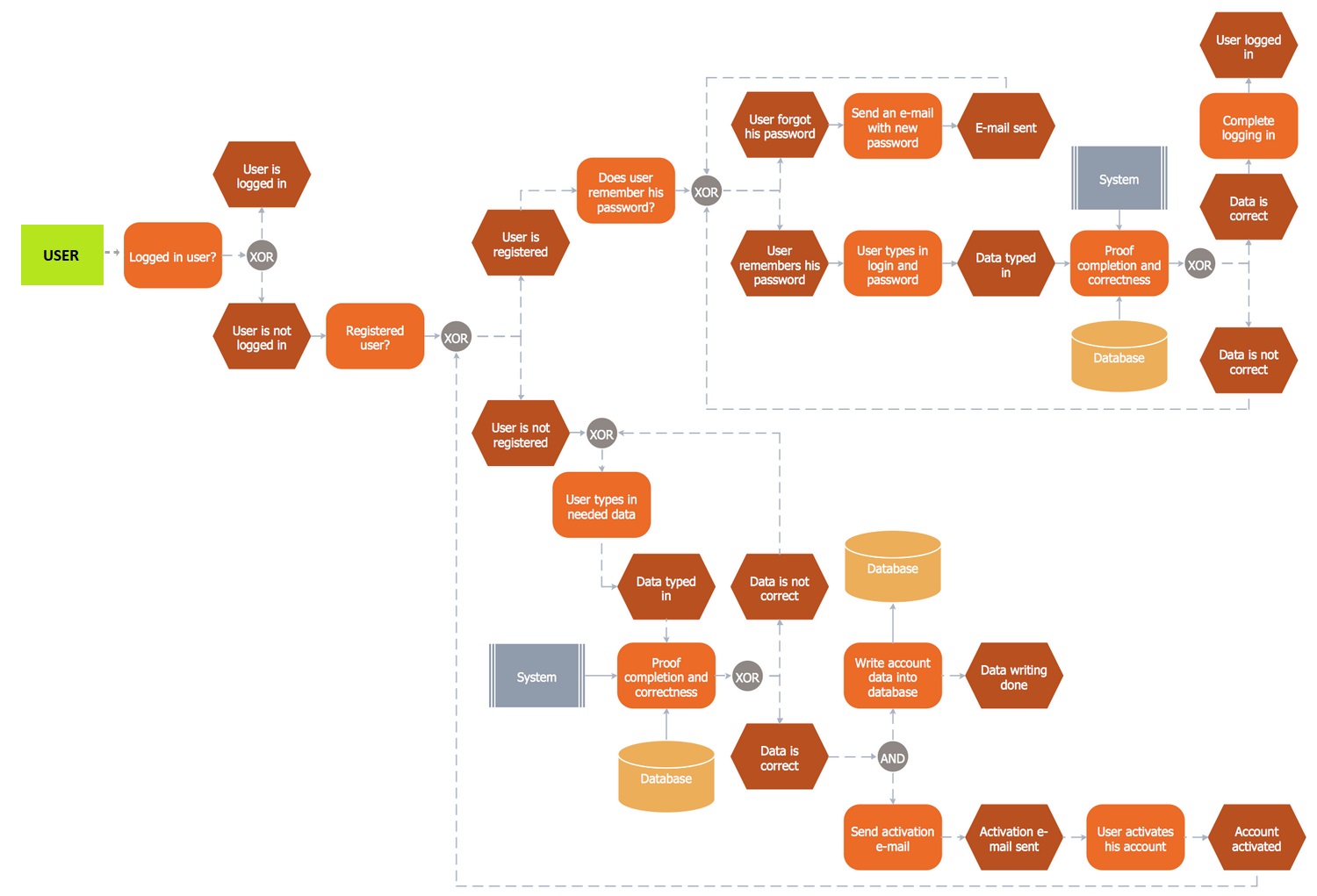
1. **Employee**



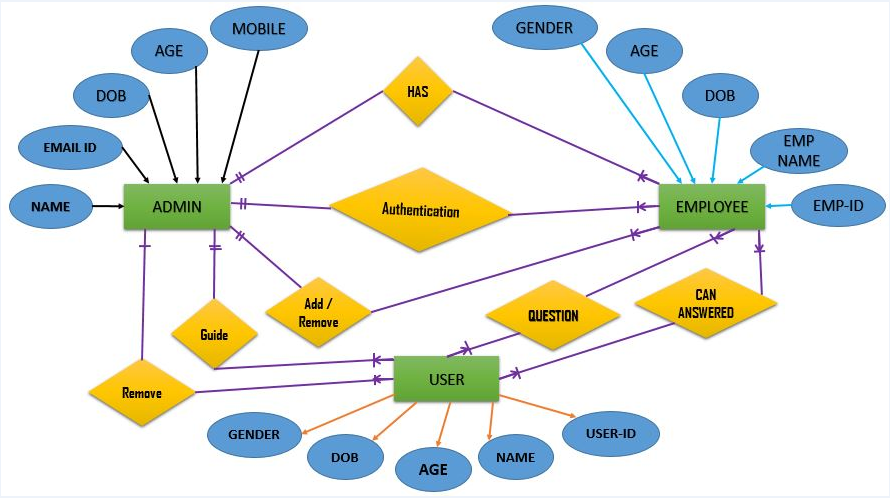
1. **Login**



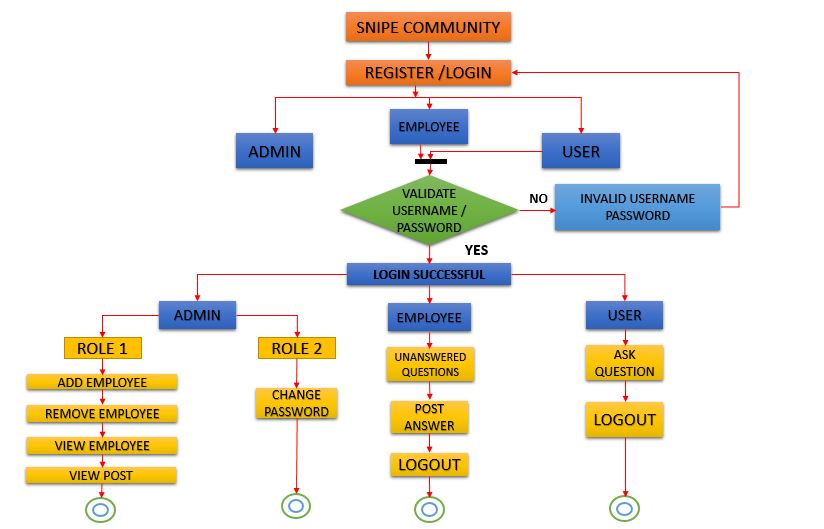
1. **Login process**

****

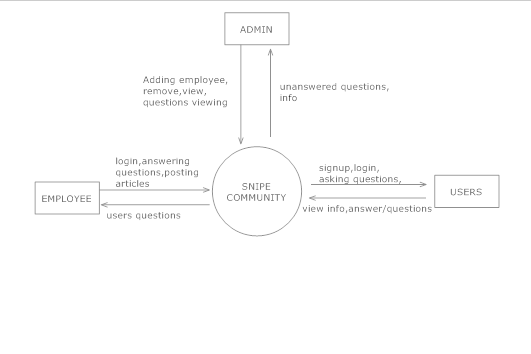
* 1. **ER diagram:**



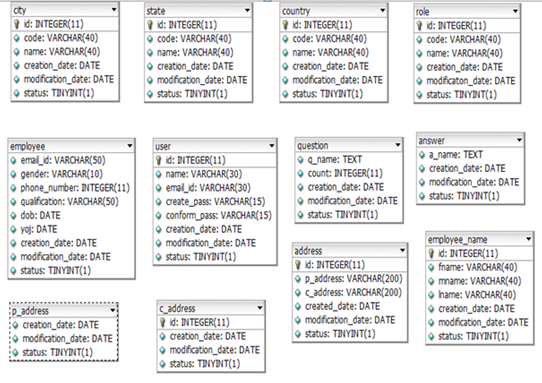
* 1. **Activity diagram:**

****

* 1. **CONTEXT DIAGRAM:**

****

* 1. **ER diagram:**



Application consists of Objects such as role, admin, employee, user, city, state, country, address, current address, permanent address, employee name, question, answer.

These objects are converted in tables and all tables are represented in one schema, generally tables are categorized in to two as master table and transaction tables.

**Master tables** related to an application are tables whose data store the master data related to the application. It's data are almost fixed and rarely change. It could hold data like your application logs location area, application version, application constants

Master tables are as followed…

Role

Admin

Employee\_name

User

Address

Permanent address(p\_address)

Current address(c\_address)

City

State

Country

**Transaction Tables** here Data which frequently changes. For example, the company is selling some materials to one of the customer.So they will prepare a sales order for the customer. When they will generate a sales order means they are doing some sales transactions.Those transactional data will be stored in Transactional table.

This is really required to maintain database normalization.

Transaction tables are as followed…

Employee

Question

Answer

There are two ways to design Entity Relationship diagrams,

1. Before designing the database
2. After designing the database

Here generating ER diagram for an application after designing database is explained.

**Process of generating ER/EER**

Requirements

Dbdesigner 4(tool for reverse enginerring process)

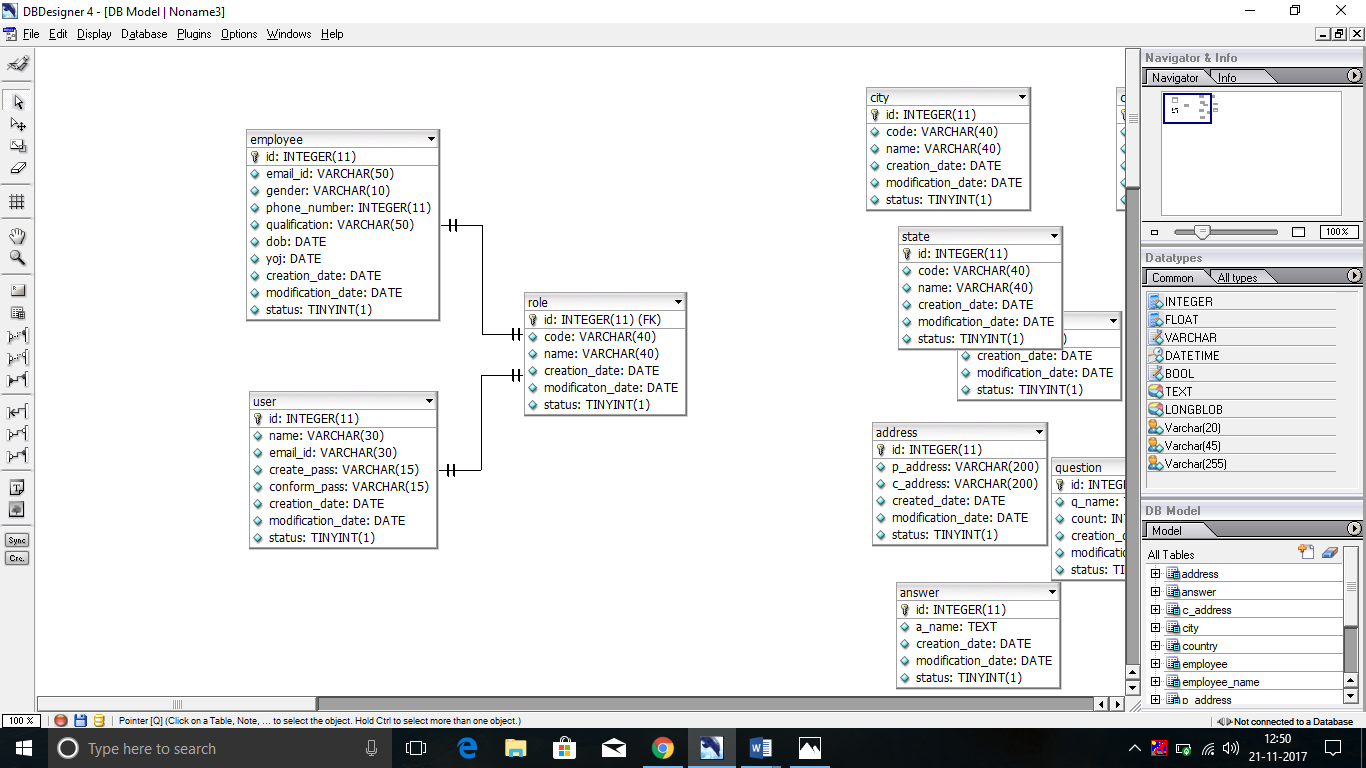
mysql(any database is suggested, here used mysql 5.5)

Step 1) create a database in mysql, create schema, create tables with specific fields.

Step 2) In dbdesigner, go to file and choose open from database, there network hosts and database connecton window appears, make a proper connection to mysql and dbdesigner.

Step 3) check the localhost, all the tables of database are shown here, and check all boxes to select them and connect.

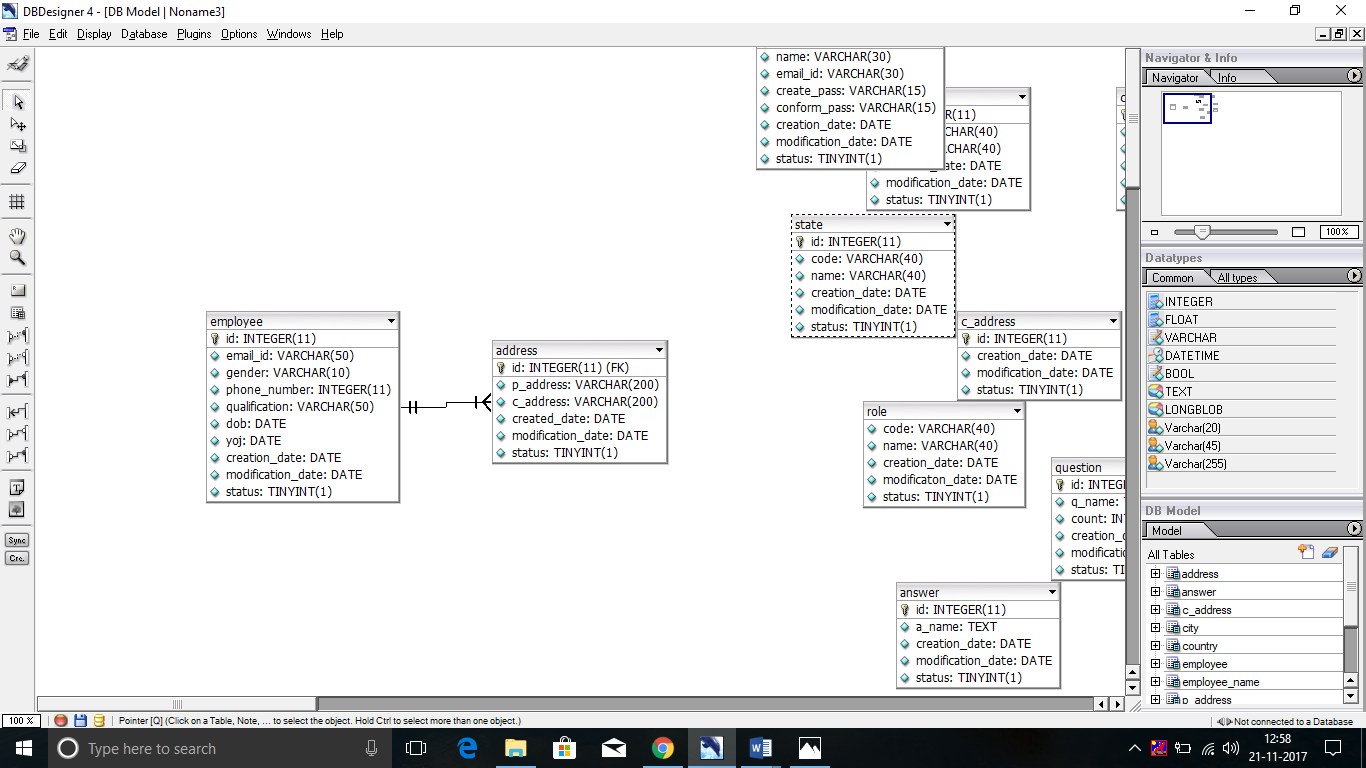
Step 4) select ‘reverse engineering process’ option in database in menu bar. EER is automatically generated for all objects in the database. And here how each object is related with another object is shown by pictorial representation.



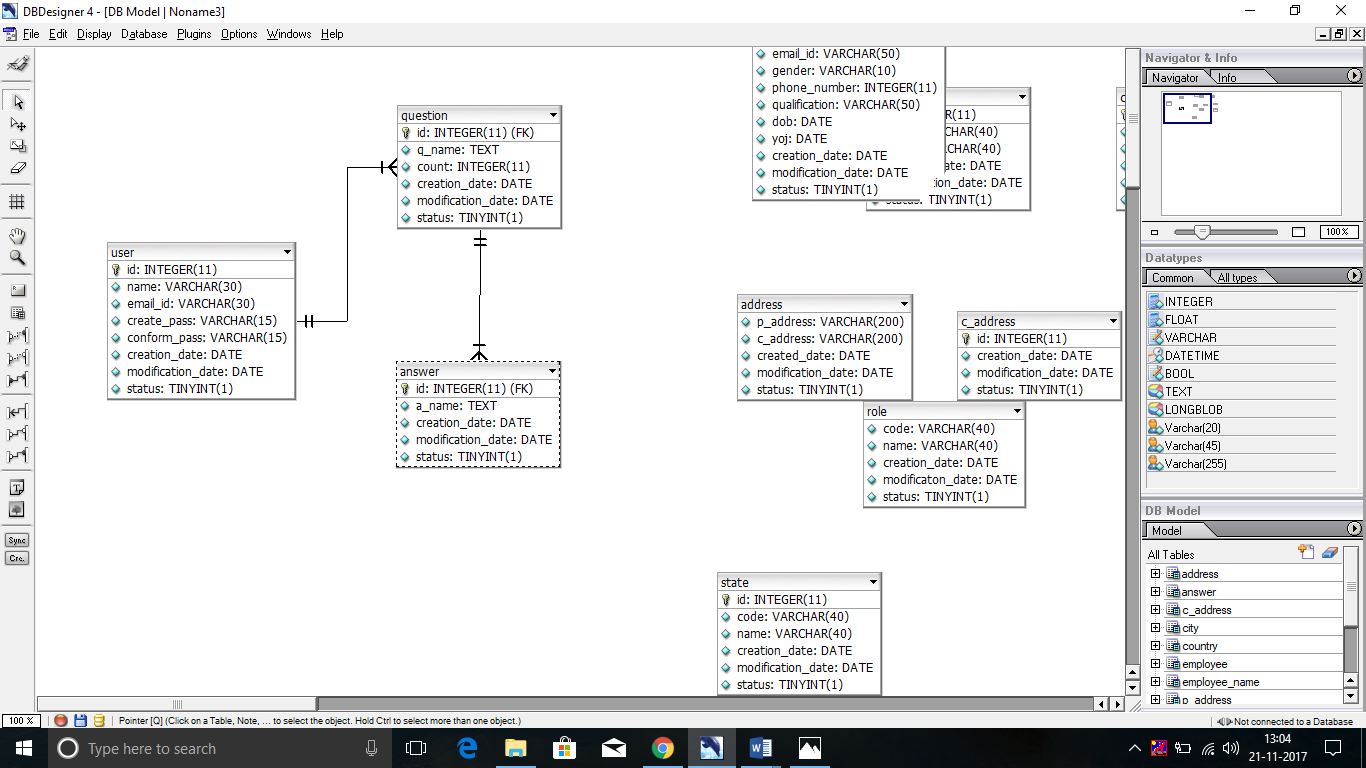
Employee has 1 role

User has 1 role

Both have 1:1 relationship

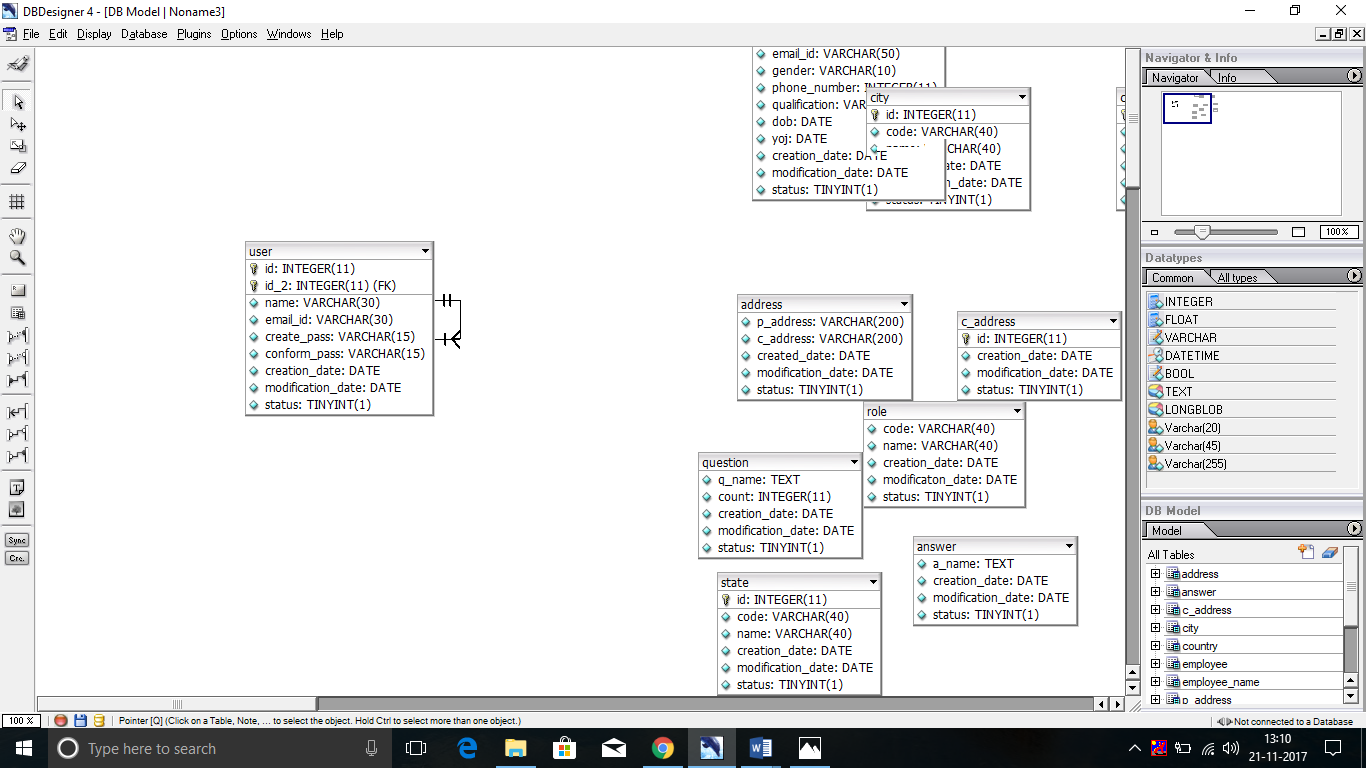


One employee can have many address.(1:n relationship)

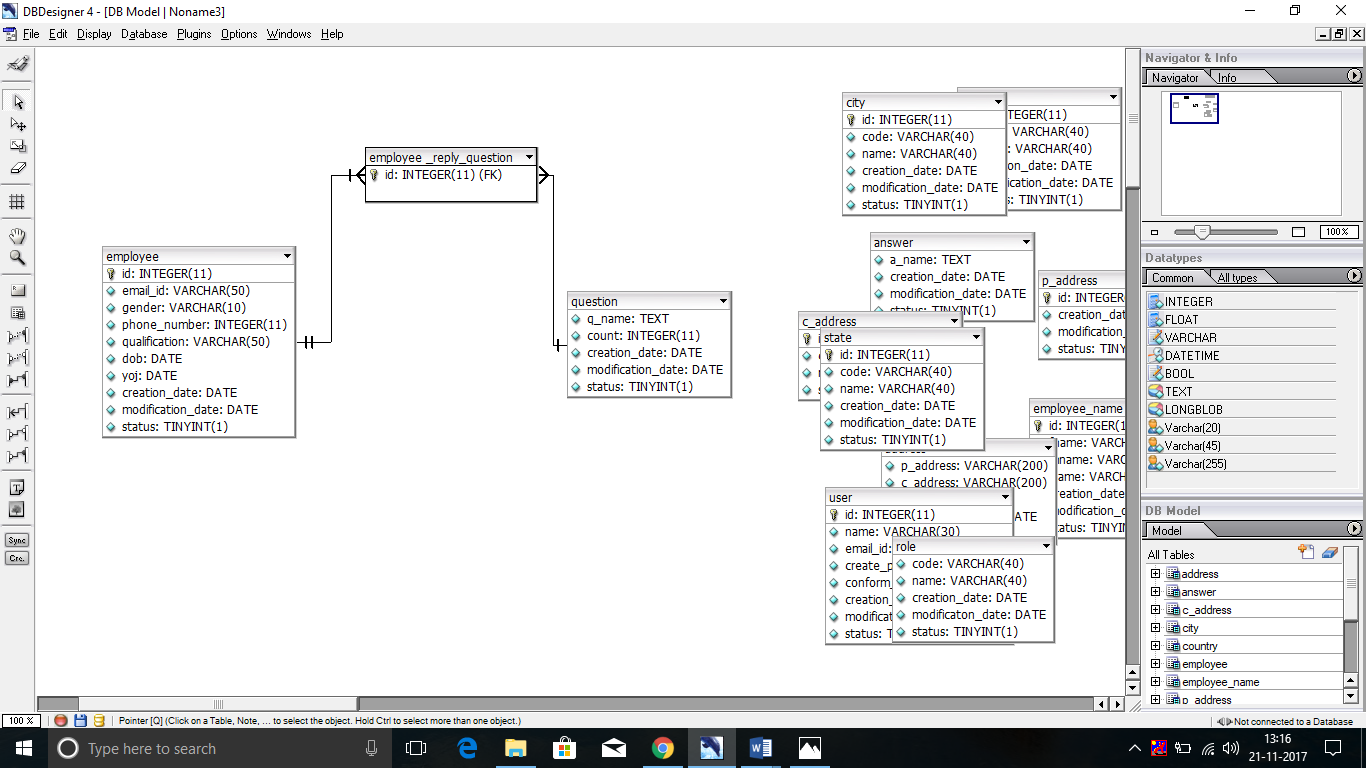


One user asks many questions (1: n relationship)

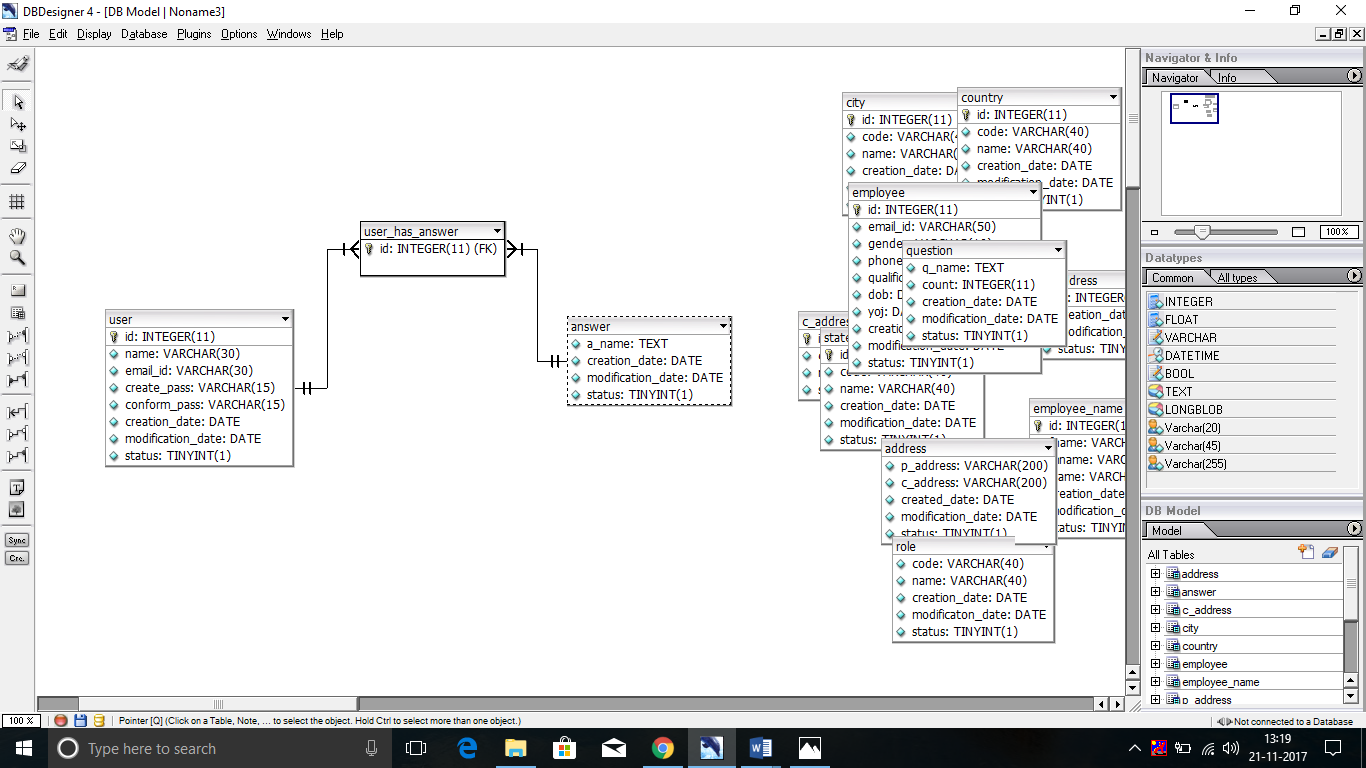
One question may have multiple answers (1: n relationship)



One user can register only once, relationship indicates 1:1relationship to its own entity.

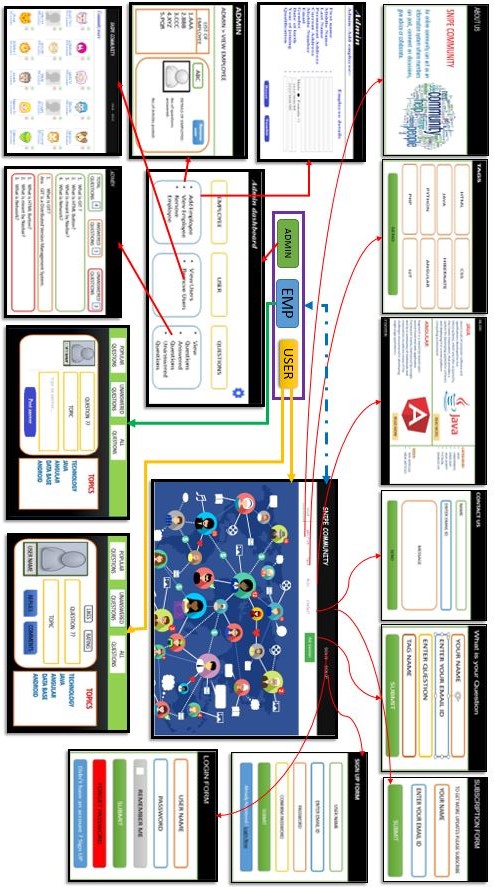


Many employees reply many questions, so relation is considered as n:m.

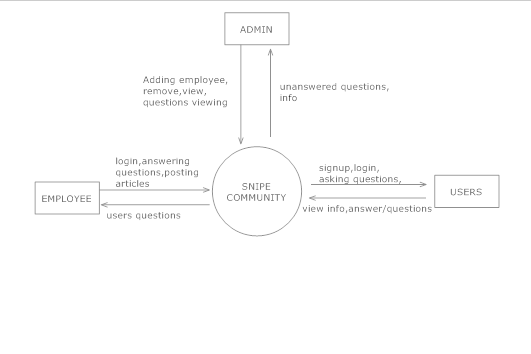


Many user can view many answers, here both entities invlolves in n:m relatiopnships.

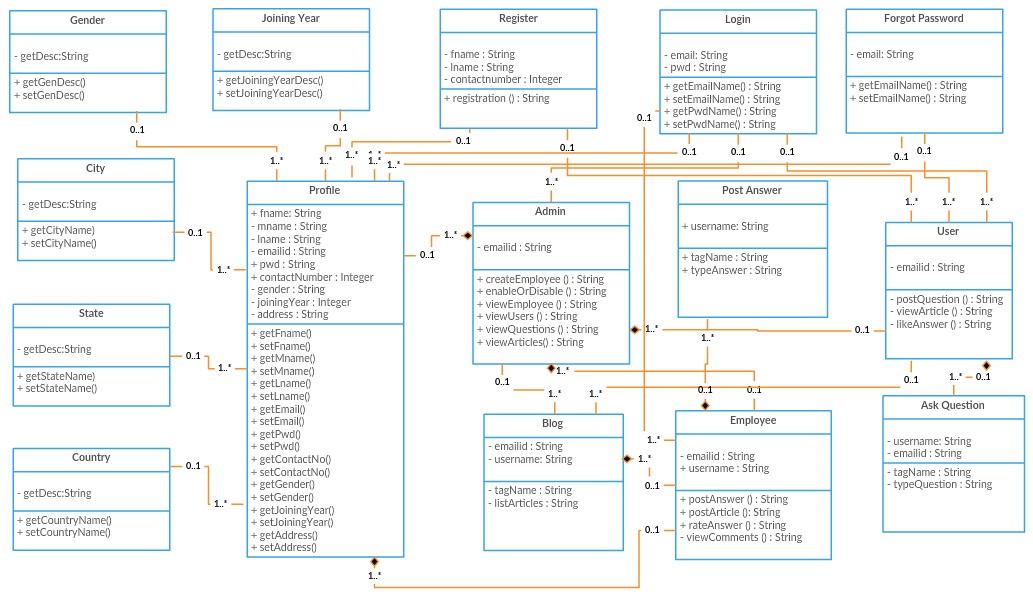
* 1. **Wireframe diagram:**

****

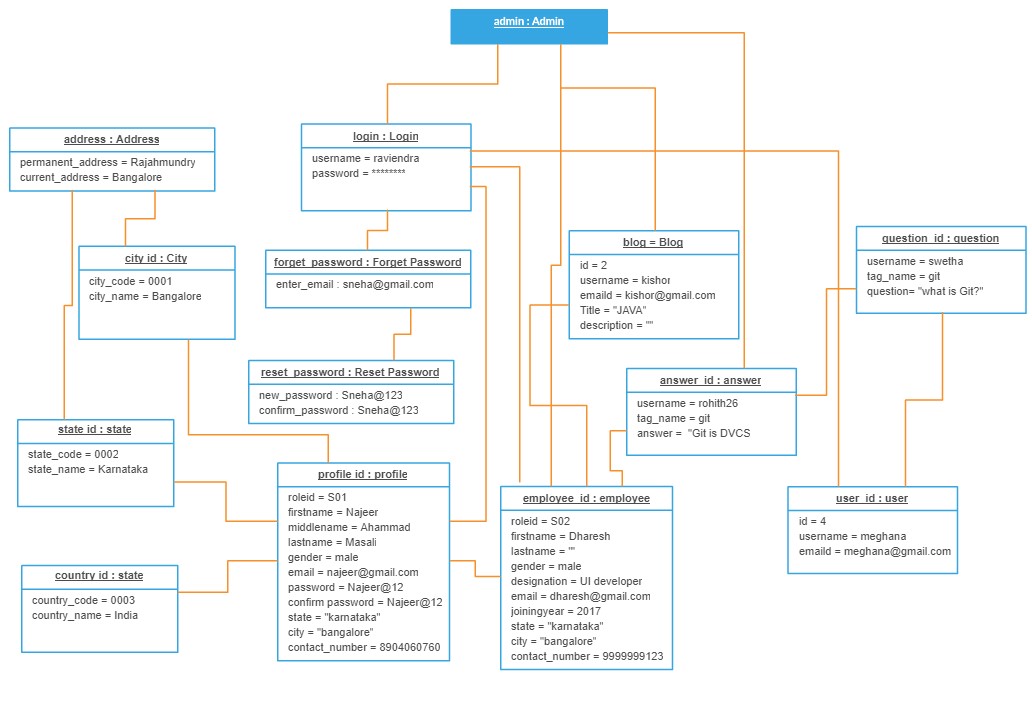
* 1. **Context Diagram**

****

* 1. **Class Diagram**

****

* 1. **Object Diagram**

****

1. **Actors Documentation**
2. **Admin:**

Admin logins authenticates with the employees and he can able to add or remove the employees. He can upload & view the details of the posts posted by the user and employees and he verify whether employees are responding or not, if no then admin can take actions on employees.

1. **Employee:**

Admin adds the employees, after login and he is the responsible for answering for the users questions.

1. **User:**

User registers, authenticates, logins with giving some details of email id or mobile number for security purpose. After creating his account in the snipe community he needs to login to his respective account, then he can ask the questions.

1. **REQUIRMENTS**

This section describes in detail all the functional and non-functional requirements.

* 1. **Use-Case Specifications**

1. **Sign up**

User needs to register to get all the details.

**Pre-Condition:** All mandatory fields need to be filled.

**Post-Condition:** Registration successful.

1. **Authenticate:**

The email and phone numbers are authenticated for security purpose.

**Pre-Condition**: Proper email and phone number should be provided.

**Post-Condition:** Authentication successful.

1. **Login**

All the actors will login to the community.

**Pre-Condition**: Proper user name and password must be provided.

**Post-Condition:** Login successful.

1. **Upload the questions.**

User needs to upload the questions.

**Pre-condition:** Complete the question need to be uploaded and all the mandatory fields’ needs to be filled.

**Post-Condition:** Upload successful.

1. **View Post**

Employee is responding to the users questions and posting the answers.

**Pre-Condition**: Uploads or information needs to be available.

**Post-Condition:** Answers uploaded successful.

1. **Subscription**

If user wants notifications from the snipe community he needs to subscribe to our community.

**Pre-Condition**: All the mandatory fields must be filled by the user by giving email id to get notification.

**Post-Condition:** User will get notifications.

* 1. **User Characteristics**

The Users one who have problems and queries in coding or development or related to jobs they can use it and ask the questions.

* + 1. **Usability**
* The web application shall allow the users to access this community from the internet.
* Since all users are familiar with the general usage webapps, no specific training is required and many the students and graduates confused about the job and about their subjects this app can give effective solution for that.
* The community is user friendly and self-explanatory.
  + 1. **Reliability**

The application process all work completely and successfully without being terminated.

* + **Availability**

The application will be available 365 days a year and 24 hours a day.

* + **Accuracy**

The application works accurately in all the scenarios such as huge data, and also in the same speed.

* + **Response Time**

The application will respond to the queries in fractions of seconds.

* + **Throughput**

The application will work accurately for all kind of inputs.

* + 1. **CAPACITY**

The application is capable of handling any number of users at a time.

* 1. **REQUIREMENTS**

**Homepage**

* Home
* Registration
* Login
* About us
* Tags
* Blog
* Question
* Contact

|  |  |  |  |
| --- | --- | --- | --- |
| **Snipe\_FI\_Req\_Ids** | **Functionality** | **Description** | **Comments** |
| Snipe\_FI\_Req\_01 | Home | This specifies the information about the snipe community project. |  |
| Snipe\_FI\_Req\_02 | Registration | User will fill the relevant details in registration page |  |
| Snipe\_FI\_Req\_03 | Login | Everyone has to login in to their account in order to post. |  |
| Snipe\_FI\_Req\_04 | About us | This specifies information of the community. |  |
| Snipe\_FI\_Req\_05 | Tags | It specifies the topics/tags related to technology |  |
| Snipe\_FI\_Req\_06 | Blog | This specifies the articles on technology |  |
| Snipe\_FI\_Req\_07 | Question | User can add the questions and mention their tag. |  |
| Snipe\_FI\_Req\_08 | Contact | If there is any queries related to the answer contact the team. |  |

**LOGIN**

* Login
* Forgot password
* Reset password

**Login:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Snipe\_FI\_Req\_Ids** | **Functionality** | **Description** | **Comments** |
| Snipe\_FI\_Req\_09 | Email or Username | This has to be filled in order to login in to their account. |  |
| Snipe\_FI\_ Req\_10 | Password | The password relevant to their Username which was created at registration time has to be entered. |  |
| Snipe\_FI\_ Req\_11 | Button: Submit  Forgot password: | After entering the details, it will be checked if valid then they can access the account.  It will help to recreate a new password. |  |

**Forgot Password:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Snipe\_FI\_ Req\_ids** | **Functionality** | **Description** | **Comments** |
| Snipe\_FI\_Req\_12 | Email | User has to provide their email which is registered with the Snipe Community. |  |
| Snipe\_FI\_Req\_13 | Submit: | After entering email if we click on submit then it will validate the email with database and will provide the link to reset for that email. |  |

**Reset Password**

|  |  |  |  |
| --- | --- | --- | --- |
| **Snipe\_FI\_ Req\_ids** | **Functionality** | **Description** | **Comments** |
| Snipe\_FI\_Req\_14 | New password | After getting an link user has to click on it and generate a new password. |  |
| Snipe\_FI\_Req\_15 | Confirm password | This field is used to renter the password. |  |
| Snipe\_FI\_Req\_16 | Button: Submit | This field is used Submit after entering password. |  |

**Registration**

* User registration
* Employee registration

User Registration:

|  |  |  |  |
| --- | --- | --- | --- |
| **Snipe\_FI\_Req\_Ids** | **Functionality** | **Description** | **Comments** |
| Snipe\_FI\_Req\_01 | First name | User will enter their first name. |  |
| Snipe\_FI\_Req\_02 | Middle name. | User will enter their middle name. |  |
| Snipe\_FI\_Req\_03 | Last name | User will enter their last name. |  |
| Snipe\_FI\_Req\_04 | Gender | User will enter their gender. |  |
| Snipe\_FI\_Req\_05 | Date of Birth | User will enter the DOB. |  |
| Snipe\_FI\_Req\_06 | Address | User will enter their address. |  |
| Snipe\_FI\_Req\_07 | Country | User will enter their respective country. |  |
| Snipe\_FI\_Req\_08 | State | User will enter their respective state. |  |
| Snipe\_FI\_Req\_09 | City | User will enter their respective city. |  |
| Snipe\_FI\_Req\_10 | Contact Number | User has to enter their contact no of 10 numbers. |  |
| Snipe\_FI\_Req\_11 | E-mail | User has to enter correct email of them. |  |
| Snipe\_FI\_Req\_12 | Password | They have to create a new password with minimum of 8 characters. |  |
| Snipe\_FI\_Req\_13 | Confirm Password | User has to retype the password in order to confirm it. |  |
| Snipe\_FI\_Req\_14 | Button: Submit | After entering all the details if we submit it will be sent to the database. |  |

**Employee Registration:**

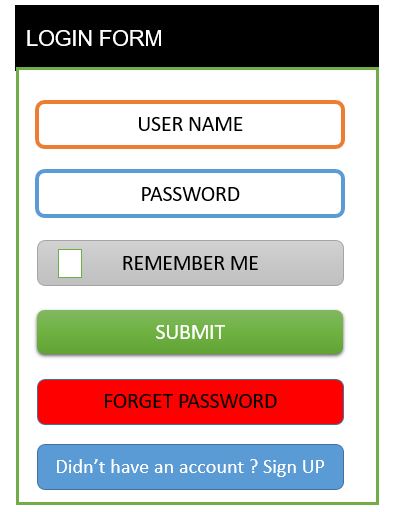
|  |  |  |  |
| --- | --- | --- | --- |
| **Snipe\_FI\_Req\_Ids** | **Functionality** | **Description** | **Comments** |
| Snipe\_FI\_Req\_01 | First name | Admin will enter their first name. |  |
| Snipe\_FI\_Req\_02 | Middle name. | Admin will enter their middle name. |  |
| Snipe\_FI\_Req\_03 | Last name | Admin will enter their last name. |  |
| Snipe\_FI\_Req\_04 | Gender | Admin will enter their gender. |  |
| Snipe\_FI\_Req\_05 | Date of Birth | Admin will enter the DOB |  |
| Snipe\_FI\_Req\_06 | Joining Year | Admin will enter the year |  |
| Snipe\_FI\_Req\_07 | Address | Admin will enter their address. |  |
| Snipe\_FI\_Req\_08 | Country | Admin will enter their respective country. |  |
| Snipe\_FI\_Req\_09 | State | Admin will enter their respective state. |  |
| Snipe\_FI\_Req\_10 | City | Guide will enter their respective city. |  |
| Snipe\_FI\_Req\_11 | E-mail | Guide has to enter correct email of them. |  |
| Snipe\_FI\_Req\_12 | Password | They have to create a new password with minimum of 8 characters. |  |
| Snipe\_FI\_Req\_13 | Confirm Password | Guide has to retype the password in order to confirm it. |  |
| Snipe\_FI\_Req\_14 | Button:  Submit | After entering all the details if we submit it will be sent to the admin. |  |

1. **USER INTERFACE:**
   1. **Home Page:**



|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * Sign in | Links to the sign in page |
| * Sign up | Links to the sign up page |
| * About Us | Information about the company |
| * Tags | Links to the tags page |
| * Blog | Links to the Blogs page |
| * Contact | Links to the contact page |
| * Ask Question | Links to the Ask question page |
| * Members | Links to the members page of the community. |

* 1. **Login**

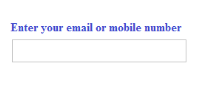
****

|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * User name | Enter the user name , it should be less than 16 char |
| * Password | Enter the password, password should be alphanumeric means containing number with the characters. |
| * Login | Giving the required username and password users can login. A message box will popup “login successful” and returns to the home page. If login is successful user can post the questions. |

**Forgot password:**

* If user forgets his password then he needs click on forgot password.
* It will ask registered email id or phone number.
* Entering mobile number or email id OTP will send.
* Once again needs to enter the OTP.
* Then user can able to reset the password.

**1.**



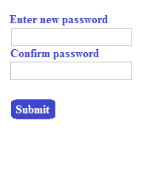
|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * mobile number or email id | Enter the emailed and a verification code will be sent to the email or mobile number. |

**2.**



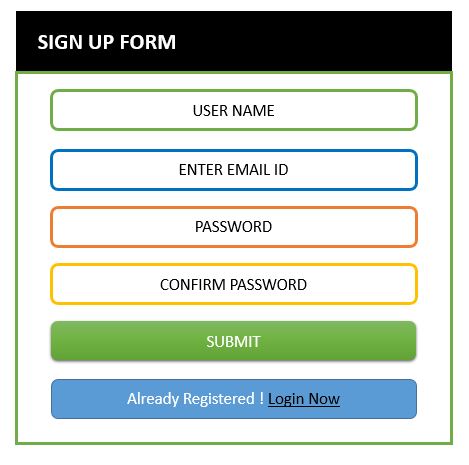
|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * OTP | Have to enter the verification code sent to the mail id and giving continue it links to the change password page |

**3.**



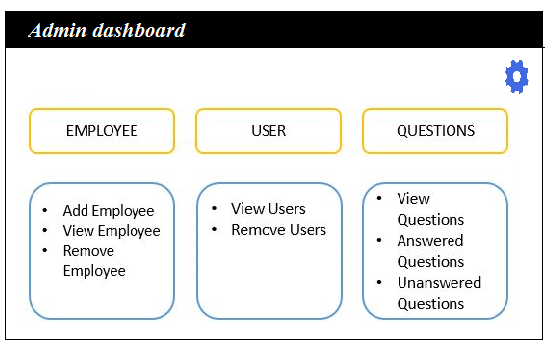
|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * Enter new password | Enter the new password containing alpha and numeric characteristics. |
| * Confirm password | Once again reenter the password to confirm. |
| * Submit | After entering password then click submit. “Password changed successfully” Giving ok returns to the home page. |

* 1. **Sign up:**

****

|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * User name | Enter the user name for community account. User name should be within 16 char. |
| * Email id | Enter the email id to authenticate with user. |
| * Password | Enter strong password for login purpose and password contain alpha numeric characteristics. |
| * Confirm password | Once again enter the password to confirm. |
| * Already registered | If already having account for community then user can login with user name password. |
| * Sign Up | After entering all mandatory fields then sign up. |
| Login | Login to the community by entering valid password and user name. |

* 1. **Admin page**

****

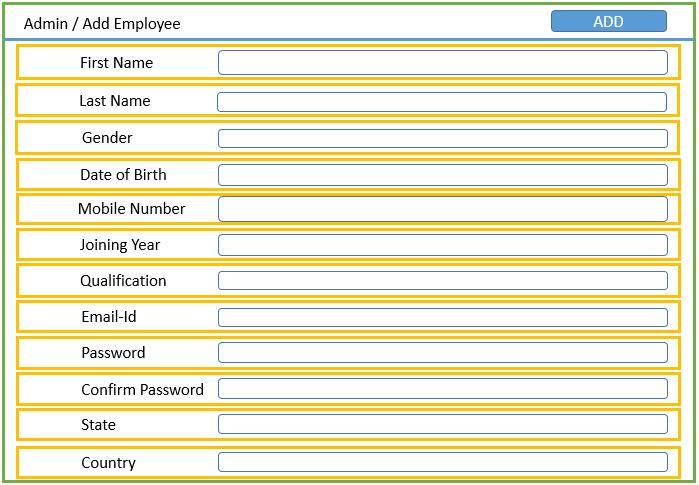
|  |  |  |
| --- | --- | --- |
| **Functionality** | **Description** | |
| * Employee | On clicking add employee, it containing some options for the admin to add, view and remove the employee. | |
| * User | Here also an admin can able to delete and he can view users. | |
| * Questions | | On clicking questions it opens how many questions are answered and unanswered questions of an user. |

* 1. **Settings**



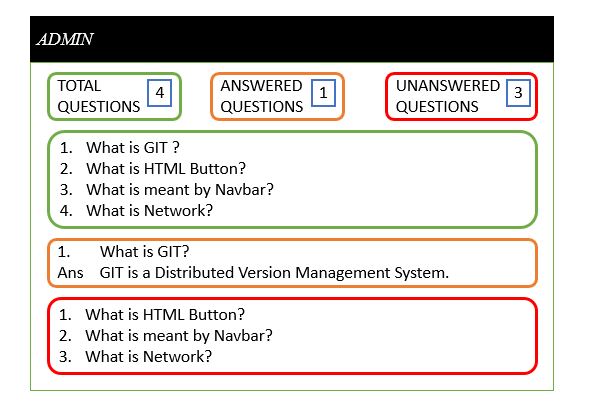
|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * Change password | If admin wants to change the password he can able to change his password here. |
| * Logout | by clicking on the logout it takes outs from the admin page to login page. |

* 1. **Admin add employee page**

****

|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * First name | Enter first name of the employee(it should contain only 10 char ). |
| * Last name | Enter last name of the employee(it should contain only 10 char ). |
| * mobile number | Enter mobile number to authenticate. |
| * Email id | Enter email id of an employee |
| * Gender | Click on the radio button to select the gender. |
| * Date of birth | Enter date of birth with the format of mm/dd/yyyy. |
| * Year of joining | Enter year of joining of an employee. |
| * Qualification | Enter the qualification of an employee. |
| * State | Enter the State |
| * Country | Enter the Country |
| * Submit | Submit button saves all data of an employee. |

**Questions:**

**y**

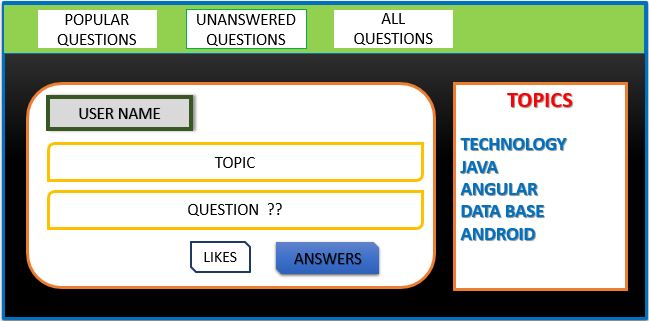
|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * Total questions answered | .Displaying total number of questions by an users. |
| * Answered questions | Showing the details of employees and users how much they answered for the questions |
| * Unanswered questions | Total number of unanswered questions by the community employees. |

* 1. **Ask question page:**

****

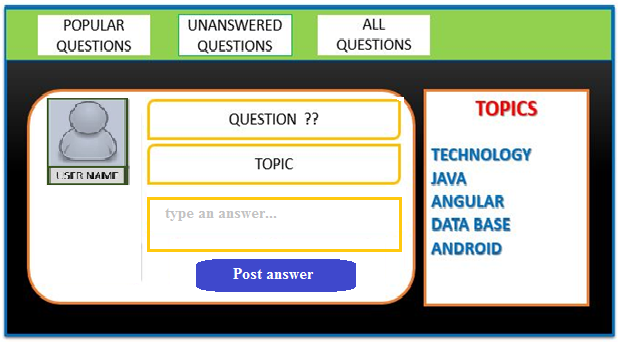
|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * Your name | Enter user name, who is going to ask and posting the questions. |
| * Enter your Email id | Email id of the user. |
| * Type your Question | Type the question which one user wants to ask. |
| * Tag name | Tag the name on which topic or sub asking question that helps to answer quickly. |
| * Submit | After filling all details then click on submit. |

* 1. **User page:**

****

|  |  |  |
| --- | --- | --- |
| **Functionality** | | **Description** |
| * Popular questions | | Links to the popular asked questions. |
| * Unanswered Questions | | Links to unanswered questions by the employees. |
| * All questions | | Opens all questions popular and asked questions. |
| * Topics | | Containing topic related questions |
| * Displaying answers | Displaying answers those which are posted by the employees with the questions and topic. | |

* 1. **Employee page:**

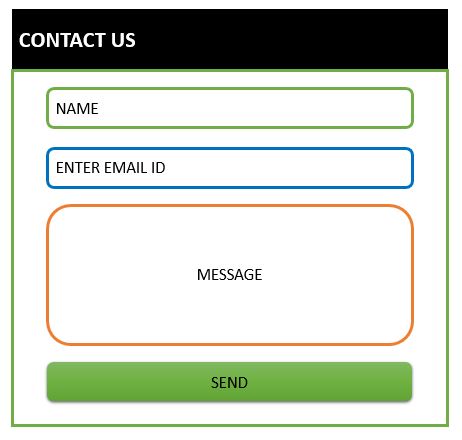
****

|  |  |  |
| --- | --- | --- |
| **Functionality** | | **Description** |
| * Popular questions | | Links to the popular asked questions. |
| * Unanswered Questions | | Links to unanswered questions by the employees. |
| * All questions | | Opens all questions popular and asked questions. |
| * Topics | | Containing topic related questions |
| * Type an answer | Employee is responding for the users questions and he typing an answers in the box. | |
| * Post answer | After typing answer for the particular question then he can post the answer. | |

* 1. **About us**

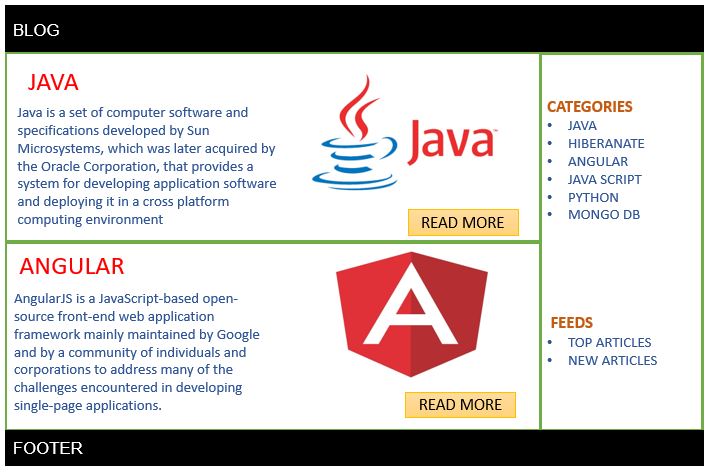
****

* 1. **Contact us:**

****

|  |  |
| --- | --- |
| **Functionality** | **Description** |
| * Name | Enter the name of an user to contact with community. |
| * Email | Enter email id to interact. |
| * Message | Type a message to ask to the community. |
| * Send | Click on send button to send or contact with the community |

* 1. **Blogs**

****

**INTERFACES**

* 1. **User Interfaces**

Our application provides interaction with web browser such as Microsoft internet explorer or Mozilla and etc.

* 1. **Hardware Interfaces**

Since the application must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable.

* 1. **Communication Interfaces**

The e-store system shall use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.