



Dharmsinh Desai University , Nadiad
Faculty of Technology
Department Of Computer Engineering

A

Project Report On

Engineering Student Helping System

By

Name	Roll No	ID
ARJUN VAGHASIYA	(CE141)	(18CEUON019)
ANIL PARMAR	(CE090)	(18CEUBG001)
SAURAV KUKADIYA	(CE066)	(18CEUOS105)

B.Tech CE Semester-VI
Subject: SYSTEM DESIGN PRACTICE

Guided by:

Prof. Jigar M Pandya

DHARMSINH DESAI UNIVERSITY

NADIAD-387001, GUJARAT



CERTIFICATE

This is to certify that the practical / term work entitled "**Engineering Student Helping System**" is a bonafied documentation of the work carried out by

- 1) **Arjun Bharatbhai Vaghasiya** , Student ID No: **18CEUON019**
- 2) **Anil Dharamashibhai Parmar**, Student ID No: **18CEUBG001**
- 3) **Saurav Hiteshbhai Kukadiya**, Student ID No: **18CEUOS105**

of Department of Computer Engineering, semester VI, under the guidance and supervision for the subject System Design Practice.

Prof. Jigar M Pandya

Assistant Professor

Department of Computer
Engineering,

Faculty of Technology,

Dharmsinh Desai University,
Nadiad

Dr. C.K Bhensdadia

Prof. and Head,

Department of Computer
Engineering,

Faculty of Technology,

Dharmsinh Desai University,
Nadiad

Table of Contents

1 Abstract.....	4
2 Introduction.....	5
2.1 Project Details: Brief Introduction	5
2.2 Technology and Tools Used	5
3 Software Requirement Specifications	6
3.1 Types of user	6
3.2 System Functional Requirements	6
4.Design.....	10
4.1 Usecase.....	10
4.2 Class.....	11
4.3 Activity.....	12
4.4 E-R Diagram	14
4.5 State Diagram	15
4.6 Sequence Diagram	16
4.7 Data Dictionary.....	17
5 Implementation Details	22
6 Testing	24
7 Screen-shots of the System.....	26
8 Conclusion	34
9 Limitations and Future Extensions of System	35
10 Bibliography	36

1.

Abstract

- ★ The Engineering Student Helping System is an online community designed to make user more active.
- ★ The Aim To This Project Is To Help Engineering Students. User Can Access System Via Creating Account.
- ★ User Can Find Post Relevant To Their Skills. User Can Also Contact Company Via Messaging. Company Can Post Information So That User Read It.
- ★ User Can Also Search Connections And Add Connection Into Their Account. User Can Also Share Their.
- ★ This System Can Help To Maintain Existing Relationships With People And Share Pictures And Messages And Also Establish New Ones By Reaching Out To People One Has Never Met Before.

2. Introduction

2.1 Brief Introduction

Now In The Time Of Latest technology And Internet Any Student Can Get Knowledge From The Internet And Gain Their Skill Equivalents to Any Top Tier Institute. For The Show Up Their Skill And Idea To People We Have Created This Web Project. And Also Startup And MNC's Can Post Their Job And Also Post Of Their Achievements And Current Demanding Technology. And Also From Others Student's Achievements We Can Motivate Ourselves And Take Help With Communicating With him/her By Messaging.

2.2 Tools/Technologies Used

Technologies :

Asp.Net Ef Core , SQL ,Tiny Mce , Signal R

Tools :

Visual Studio 2019

GitHub Command Line And Inbuilt Of Visual Studio

3. Software Requirement Specifications

3.1 Types of User

1. Admin
2. Student/Regular User
3. Employer

3.3 System Functional Requirements

3.3.1 Manage Account

Description : The user can create/update/delete/search account.

3.3.1.1 Create Account

Description : The user will be asked some basic information about him/her and information will be saved and profile will be created.

Input : User oriented information

Output : Redirection to home page of the system

3.3.1.2 Update Account

Description : The user can update his/her account information.

Input : User information

Output : Confirmation message

3.3.1.2 Delete Account

Description : User can delete his/her account.

Input : Account password and confirmation button

Output : Confirmation message and redirection to Login Page.

3.3.1.3 Search Account

Description : User can search any user's account by name.

Input : Name or any input string in search bar

Output : Display That User's Account Page

3.3.1.4 Login

Description : User will have to login before using system.

Input : Username and Password/Google Authentication

Output : Home page of the system

3.3.1.5 Logout

Description : User can log out from the system after the completion of use.

Input : Button click

Output : Redirection to log in page

3.3.2 Manage Connection

Description : User Can Unfollow And Accept Reject Connection's

3.3.2.1 Create Connection

Description : User can create connection with any other user by sending connection request to that user.

Input : Button click

Output : Confirmation message

3.3.2.2 Reject Connection

Description : User can reject other user's connection request if he/she doesn't want to make connection.

Input : Reject button click

Output : Confirmation message

3.3.2.3 Delete Connection

Description : User can delete connection made with any user.

Input : Delete connection button selection

Output : Confirmation message

3.3.3 Manage Post

Description : User Can Manage Their Post(Edit, Update, Delete)

3.3.3.1 Create Post

Description : User can create post and represent his/her thought.

Input : Selection of "Create Post" and filling all the details

Output : Created post on timeline

3.3.3.2 Update Post

Description : User can edit created post

Input : Button click and edit information

Output : Edited post on timeline

3.3.3.3 Delete Post

Description : User can delete the post created by him/her'

Input : Choose delete post option

Output : Confirmation message

3.3.4 Manage Messaging

Description : User Can Message Their Connection

3.3.4.1 Send Message

Description : User can send message to his/her connection

Input : Message and button click

Output : Sent message

3.3.4.2 Delete Message

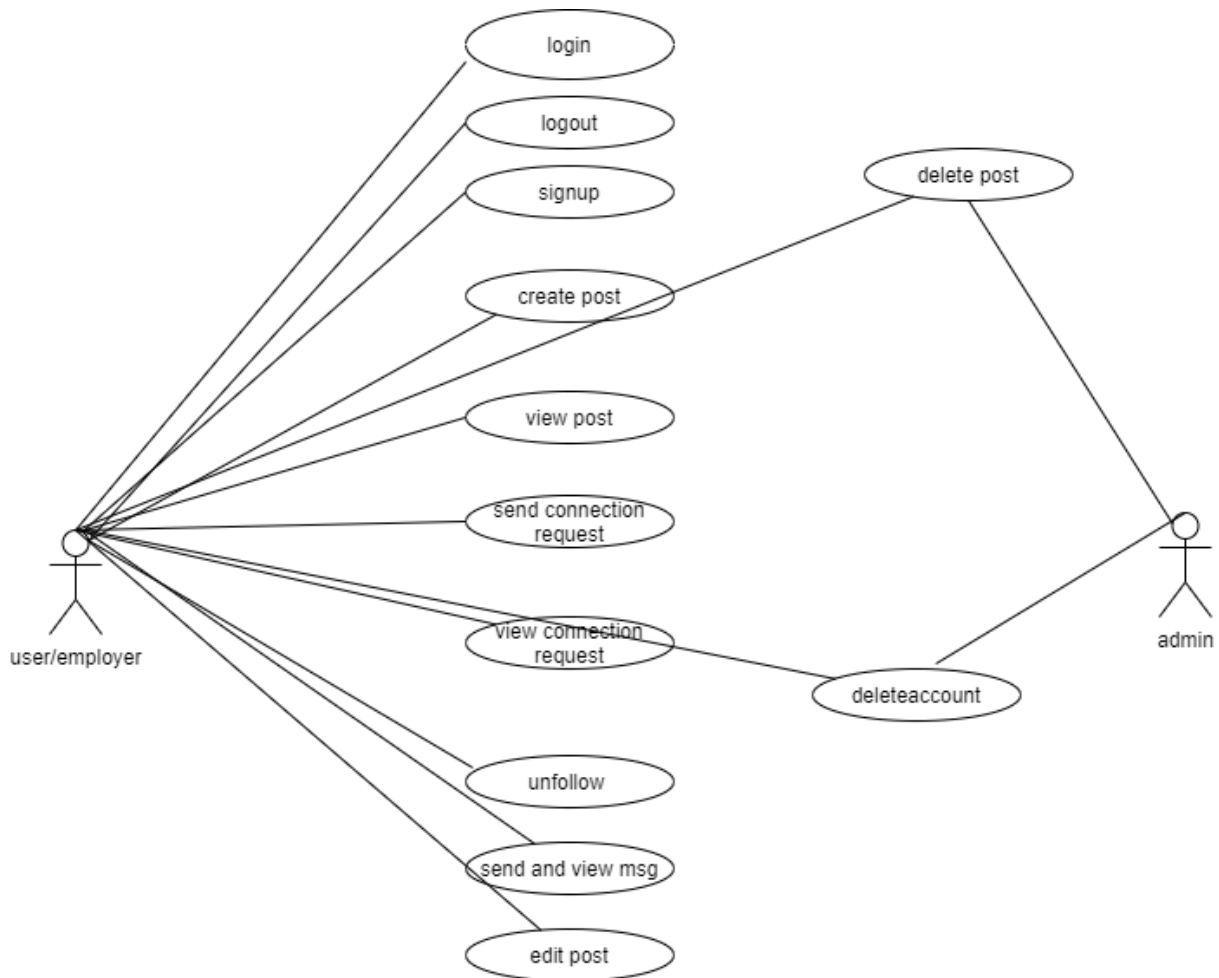
Description : User can delete sent message

Input : Selection of delete option on the message

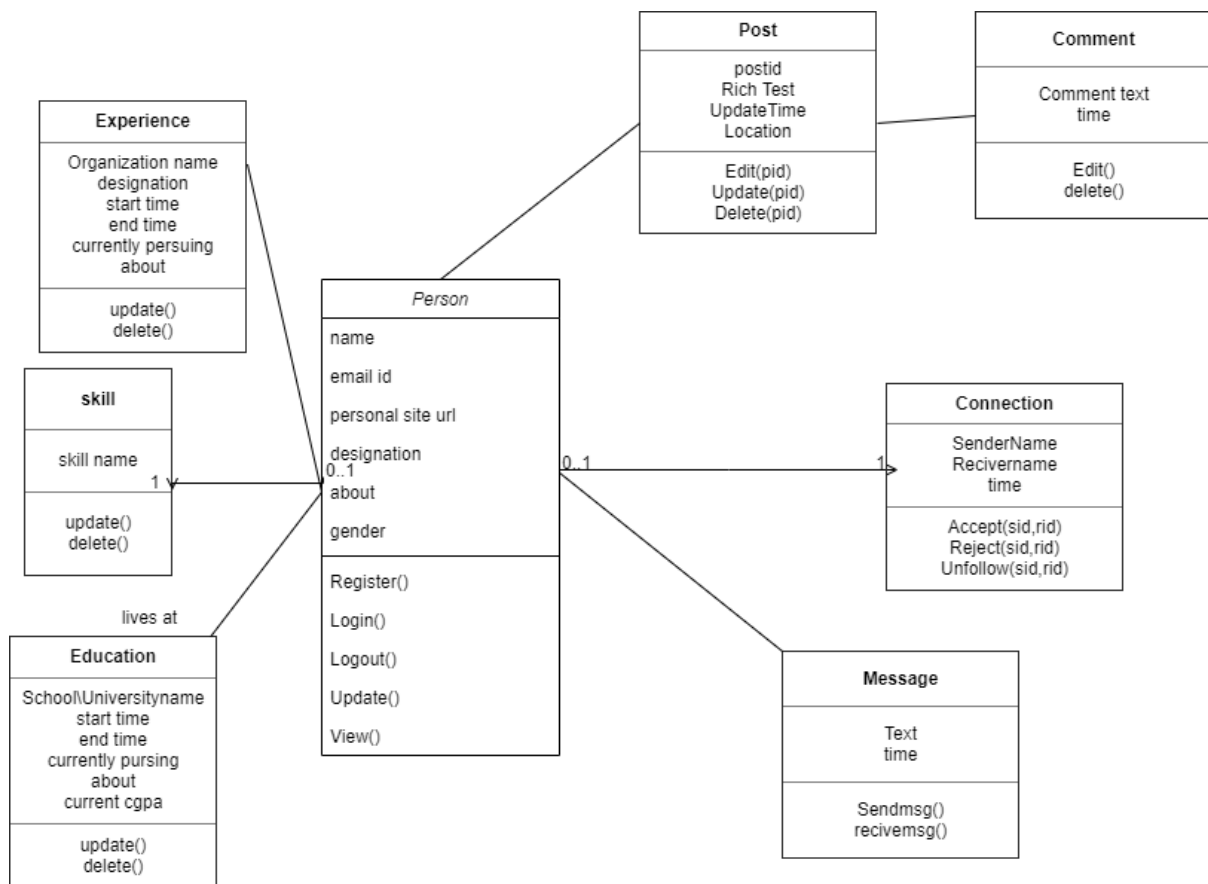
Output : Confirmation message

4. Design

4.1 USECASE DIAGRAM

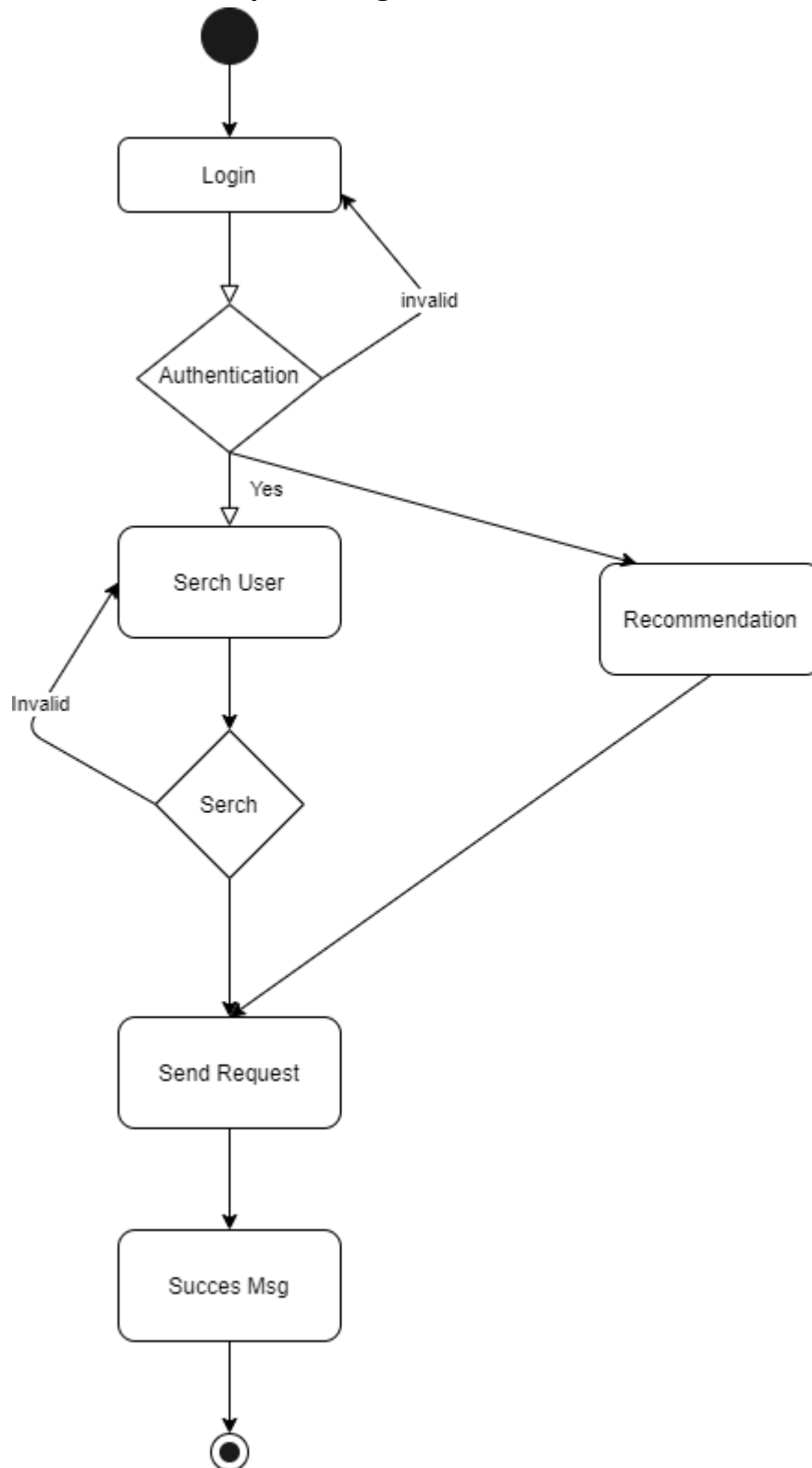


4.2 CLASS DIAGRAM

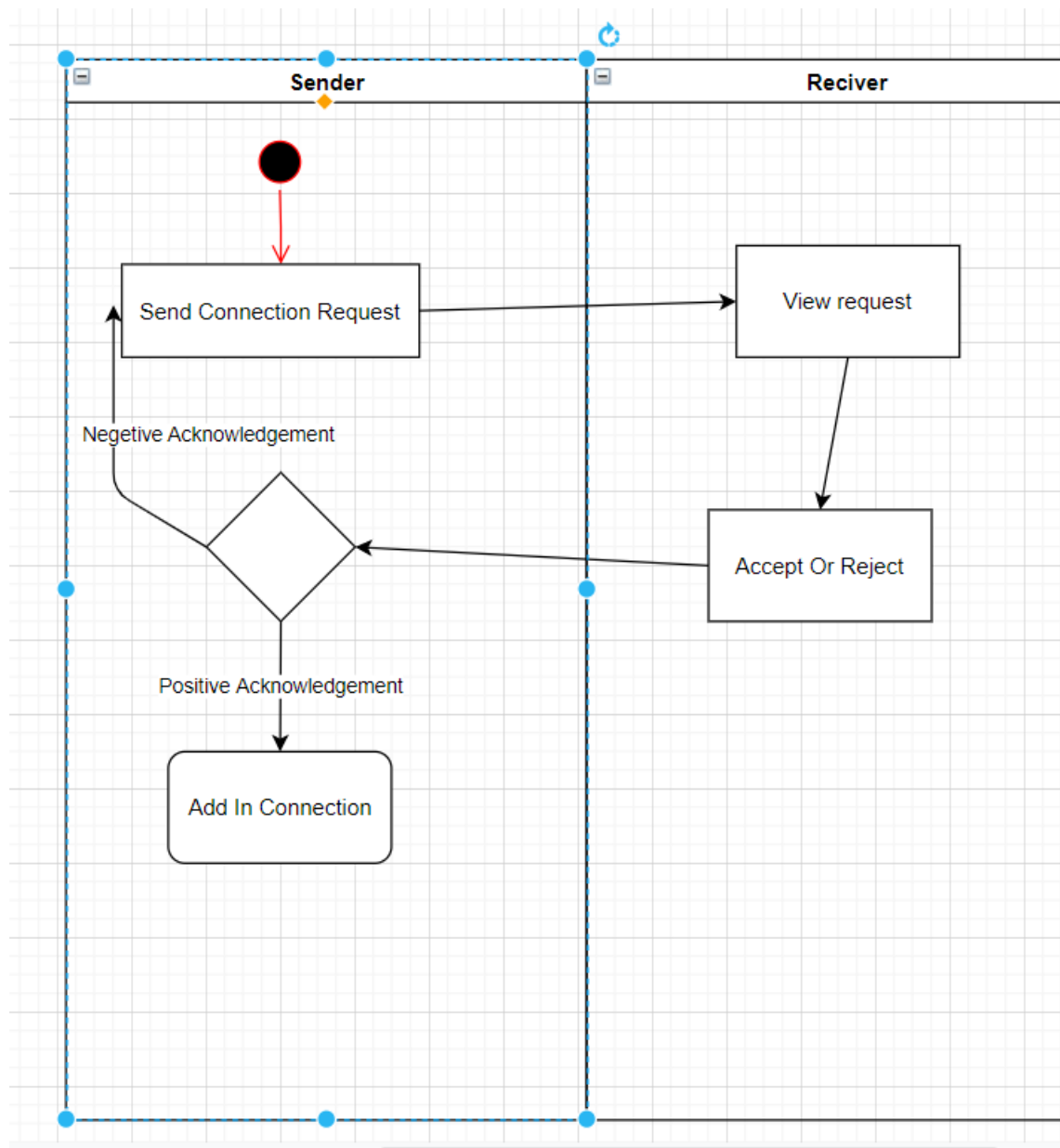


4.3 ACTIVITY

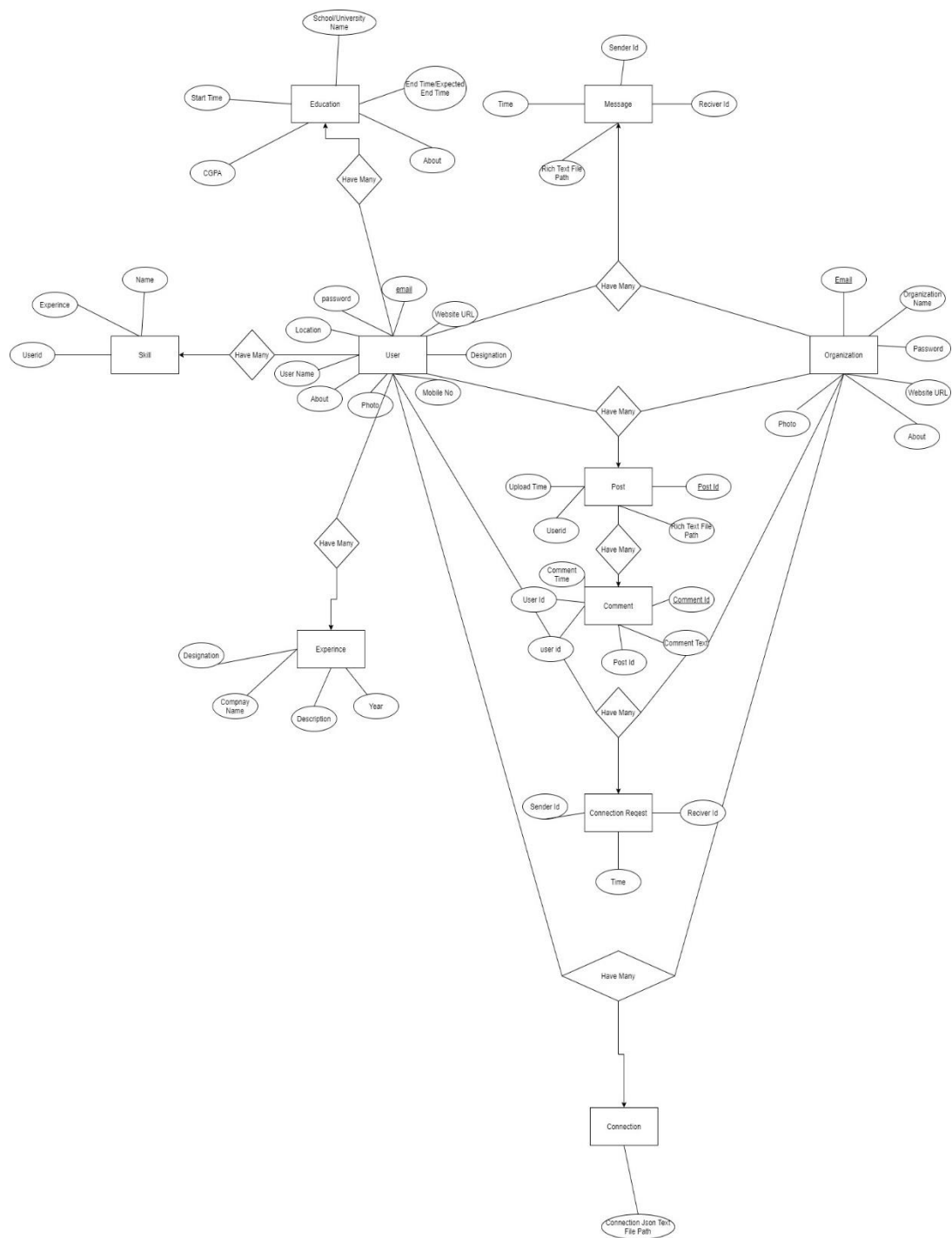
1. Connection Request Diagram



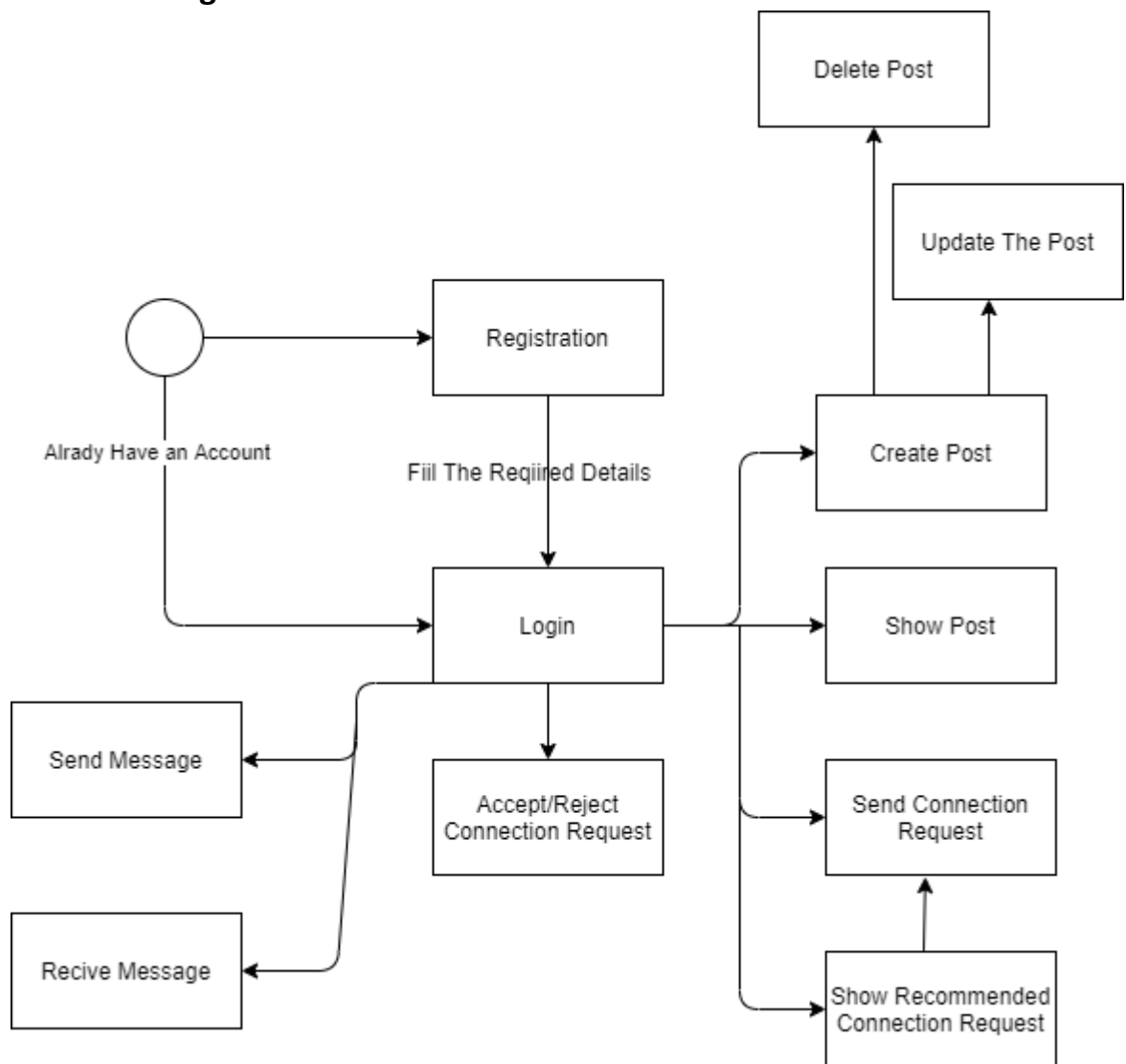
2.Connection Request Activity



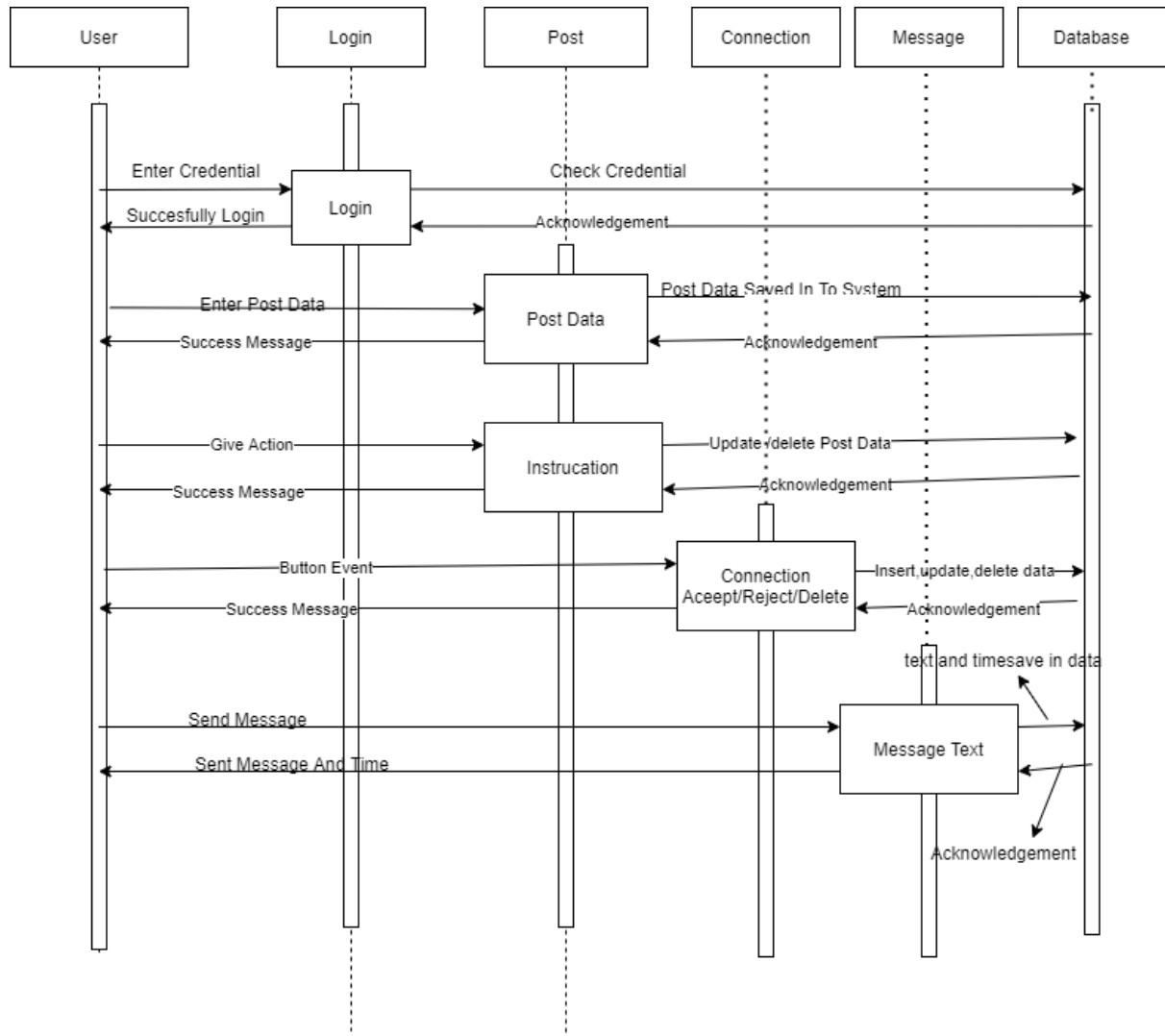
4.4 ER Diagram



4.5 State Diagram



4.6 Sequence Diagram



4.7 Data Dictionary

1.User

Field Name	Data Type	Description	Example
email	string	Primary key	abc@mail.com
name	string	Name of user	anil
url	string	url of personal website	www.abc.com
designation	string	designation of user	Frontend engineer ,student
mobilenno	string	Mobile no	99999999
photopath	string	Photo path where photo is store	/abc.com/media/a.png
about	string	Description about person	I am software engineer at xyz.
location	string	Location of user	nadiad
password	string	Password of user	123456789

2.Organization

Field Name	Data Type	Description	Example
Email	string	Email and primary key	abc@domain.com
Name	string	Name of organization	google
password	string	Password of user	123abc#.
Website URL	string	URL Of Organization	Google.com
About	string	About organization	We are world wide
Photo	string	Photo path url	/google/media/image.png

3.Skill

Field Name	Data Type	Description	Example
Userid	String	Foreign key Of User	Abc@mail.com
Name	String	Skill Name	ASP.Net
Experience	string	Experience of Person	2 years

4.Education

Field Name	Data Type	Description	Example
School name	String	School Or University Name	DDU
Start time	Date	When start this Degree or school	2018
End time	Date	Expected End Time Or End Time	2022
CGPA	String	CGPA or grade	99.99
About	String	About That Education	I Have Studying Computer Engineering

5.Message

Field Name	Data Type	Description	Example
Sender Id	String	Sender Id Of User	sender@mail.com
Reciver Id	String	Reciver Id Of User	reciver@mail.com
Date And Time	DateTime	Time When User Send Message	12:12,14-01-2021
Rich Text File Path	String	Path Of Rich Text File Path	/abc.com/msg/12m.txt

6.Post

Field Name	Data Type	Description	Example
User Id	String	Id Of Author	postcreator@mail.com
Upload Time	DateTime	Upload Date And Time	12:12,14-01-2021
Post Id	String	Primary Key	P15414484
Filepath	String	RichText File Path	/abc.com/Post/p15414484.txt

7.Comment

Field Name	Data Type	Description	Example
Postid	String	Id Of Post	postcreator@mail.com
Comment Time	DateTime	Comment Time	12:12,14-01-2021
Comment Id	String	Primary Key	C454444
Comment Text	String	Comment Text	Comment

8.Connection Request

Field Name	Data Type	Description	Example
Sender Id	String	Id Of Sender	sender@mail.com
Reciver Id	String	Id Of Reciver	reciver@mail.com
RequestTime	DateTime	Time When User Send Connection Request	12:12,14-01-2021

9.Connection

Field Name	Data Type	Description	Example
User Id	String	Userid	abc@mail.com
Filepath	String	Jsonfile Path where Connection Data Are Store	/abc.com/connection/p1.json

10.Experience

Field Name	Data Type	Description	Example
Organization Name	String	Organization Where User Gain This Experience	Google
Designation	String	Designation Of User	Frontend Engineer
Description	String	Description About This Role.	I Have Work in Angular Framework.
Year	Int	How many Year's Of Experience User Have.	2

5. Implementation Details

Registration Module:

For Registration We Use Google Authentication And Also Our Normal Registration and This Module Store User's Data. Which Will Enable User To Login

Login Module :

This module takes users credentials and then verifies it with registered users, if user is not registered the invalid credentials is shown else if they match with database then login user. And User Can Use Google Authentication.

Recommended Connection :

In This Module We Have Show User To Recommended Connection. Logic Behind This Module Is That When User Enter In To Our System Then They Provide Their School Name And Work Experience's Name By This Name We Have Recommended Other Member Of The Same School.

Search User :

In This Module User Type Other User's User Id And Then If That User Present In Our System Then We Show That User's Details. And Also We Show That This User Is Friend Or Not? And If User Type His Name Then We Have Redirect Him To Home Page. If User Is Not Present In Our System Then We Give's Him Message.

Send Connection Request:

In This Module User Can Search Other User Or From Recommended Connection User Can Send connection Request To Other User And We Store Sender Id And Receiver Id And Time In To The Database.

Accept/Reject Request :

In This Module User Can Accept Or reject The Connection Request And Base On The Choice We Change Our Database.

Create/Edit/delete Post:

In This Module User Can Create New Post. For Creating Post We Have Use Tiny MCE Editor . Using It We Can Create Beautiful Post. After Creating Post We Have Store That Rich Text In Our Local Storage With .txt File And Then Store It's Location To Our Database. And Then We Can Display The Post.

Messaging :

For Messaging We Have Use Real Time Open Source Signal-R Messaging. In This Module User Can Only Send Message To Their Connection.

6. Testing

Testcases

Test Case ID	SRS Requirement ID	Test Case Description	Expected Result	Actual Result	Tested By	Date Tested	Test Case Status (Pass/Fail)
1	4.1.1	User Registration detail for creating account	Pass	Data Inserted And Home Page	Anil	27-3	Pass
2	4.1.2	Updated User Information	Pass	Data Updated	Anil	27-3	Pass
3	4.1.3	Username For Searching User	Pass	Search Result Displayed	Anil	27-3	Pass
4	4.1.4	User Credentials For Login	Pass	Successfully Login	Anil	27-3	Pass
5	4.1.5	Logout Button Is Clicked	Pass	Successfully Logout	Anil	27-3	Pass

Test Case ID	SRS Requirement ID	Test Case Description	Expected Result	Actual Result	Tested By	Date Tested	Test Case Status (Pass/Fail)
7	4.2.1	Send Connection Request	Pass	Data Added In Connection Request Table	Anil	27-3	Pass
8	4.2.2	Reject Received Connection Request	Pass	Received Connection Request And Data Update	Anil	27-3	Pass
9	4.2.3	Delete Existing Connection	Pass	Data Deleted From Database	Anil	27-3	Pass

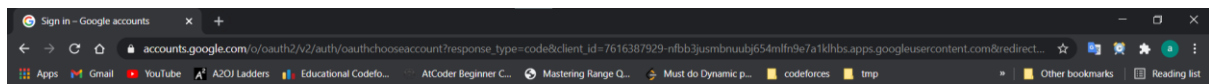
Test Case ID	SRS Requirement ID	Test Case Description	Expected Result	Actual Result	Tested By	Date Tested	Test Case Status (Pass/Fail)
10	4.3.1	Create Post	Pass	Post Text Added In Storage And Other Data Inserted In Database	Arjun	25-3	Pass
11	4.3.2	Update Post Information	Pass	Post Data Updated	Arjun	25-3	Pass
12	4.3.3	Delete Post	Pass	Post Data Deleted	Arjun	25-3	Pass

Test Case ID	SRS Requirement ID	Test Case Description	Expected Result	Actual Result	Tested By	Date Tested	Test Case Status (Pass/Fail)
13	4.4.1	Send Messages	Pass	Message Sent	Saurav	25-3	Pass
14	4.4.2	Delete Messages	Pass	Future Extension			

7. Screenshots

Register

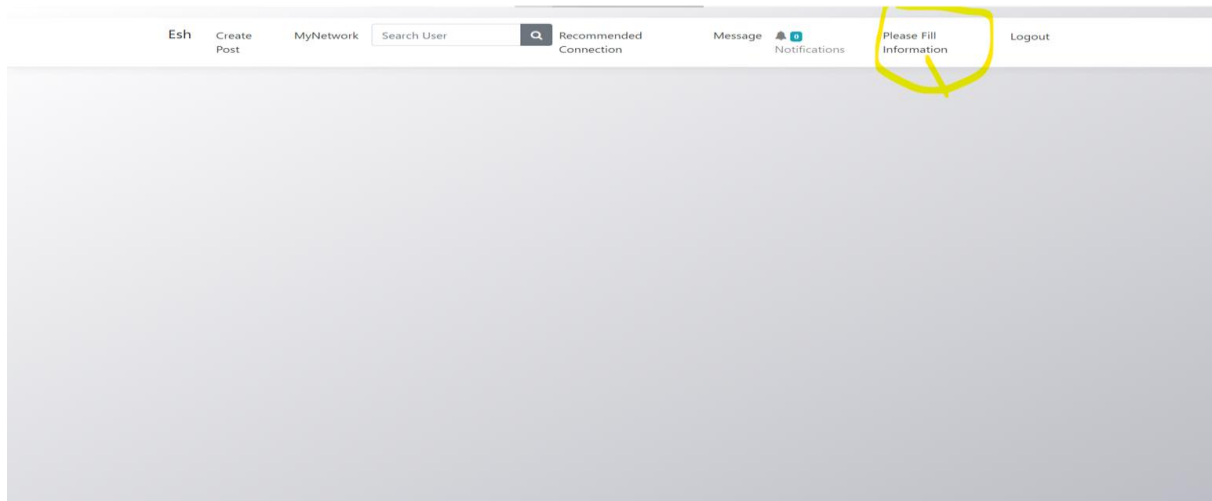
The screenshot shows the 'Register' page of the Esh website. The page has a header with the user's name 'Esh', navigation links like 'Create Post', 'MyNetwork', and a search bar. The main content area is titled 'Register' and is split into two columns. The left column, 'Create a new account.', contains input fields for 'Email', 'Password', and 'Confirm password', followed by a 'Register' button. The right column, 'Use another service to register.', features a 'Google' button.



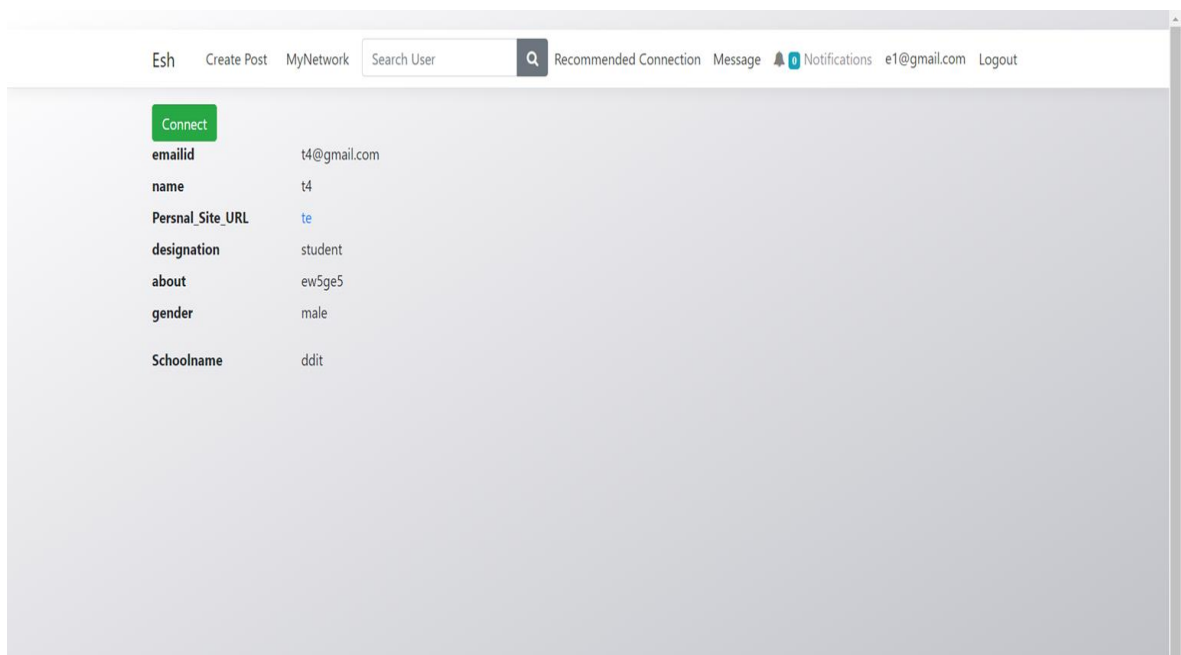
This screenshot shows a Google account selection dialog box. It prompts the user to 'Choose an account to continue to Esh'. A list of accounts is displayed, including 'anil parmar' (anilparmar76983@gmail.com), 'CEO90 - Anil Parmar' (18ceubg001@ddu.ac.in), 'Anil Parmar' (apnoticeboard@gmail.com), 'anil parmar' (newanilparmar@gmail.com), and 'Kesav Studio' (kesavstudio100@gmail.com). At the bottom, there is a 'Use another account' option and a disclaimer about data sharing.

The screenshot shows the 'Register' page of the Esh website after successful Google authentication. The heading 'Register' is followed by the sub-heading 'Associate your Google account.'. A message states: 'You've successfully authenticated with Google. Please enter an email address for this site below and click the Register button to finish logging in.' Below this, there is an 'Email' input field containing 'anilparmar76983@gmail.com' and a 'Register' button.

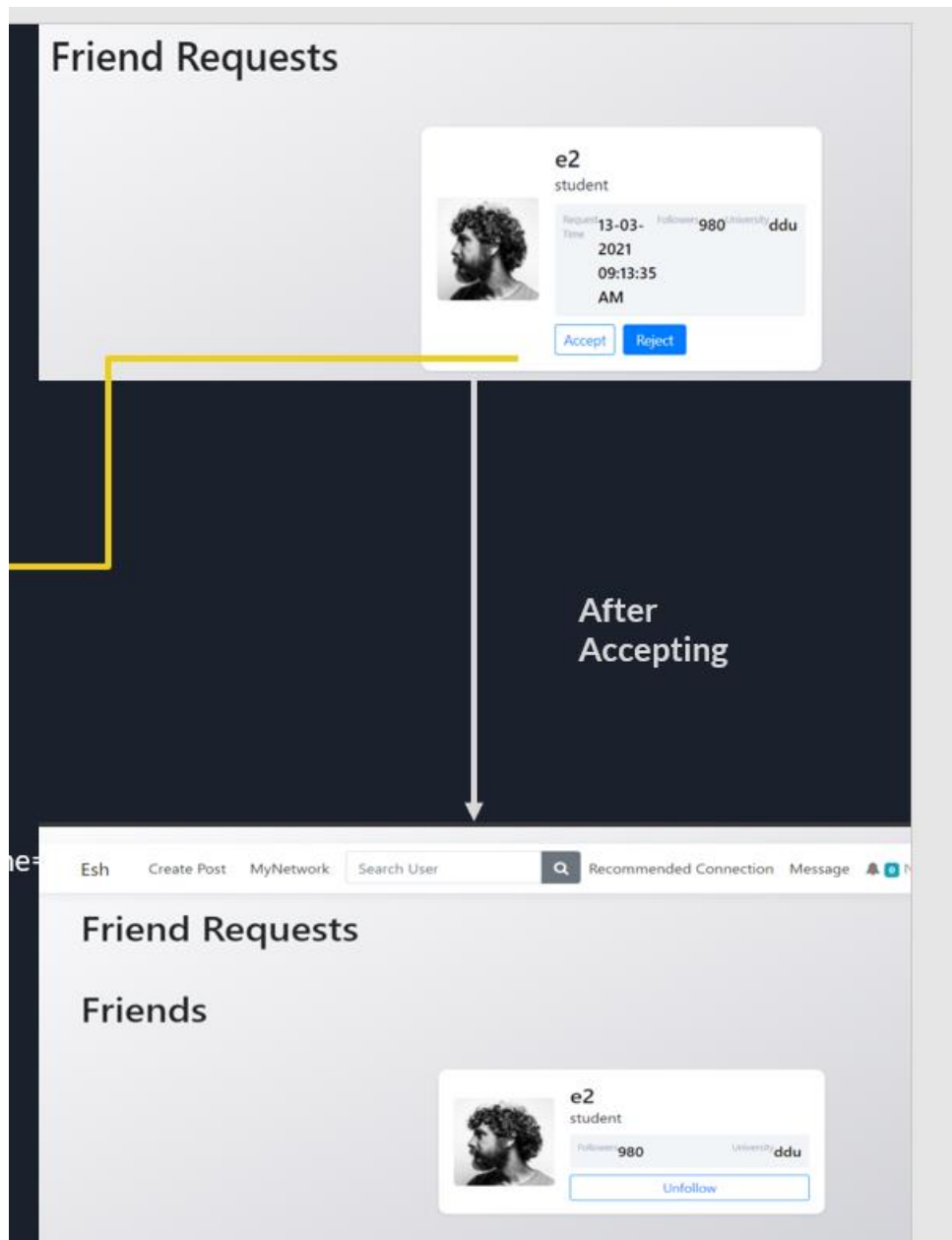
User Must Have To Provide Their Details



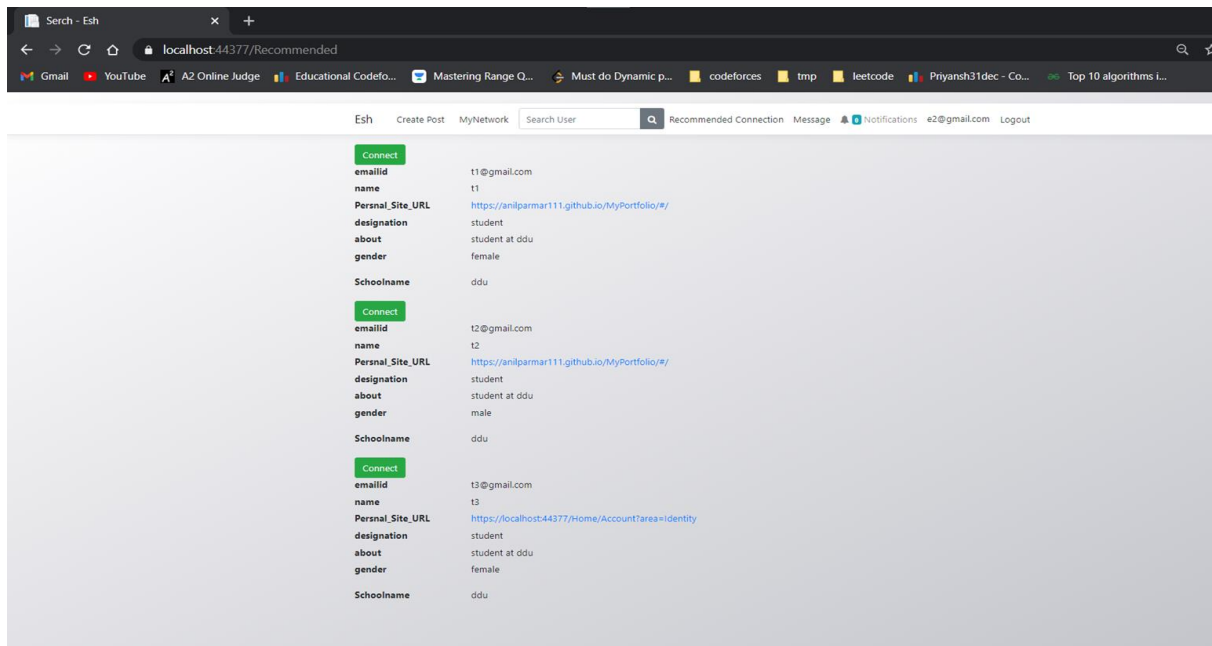
After Registration User Get Recommended Connection From Same University Or School



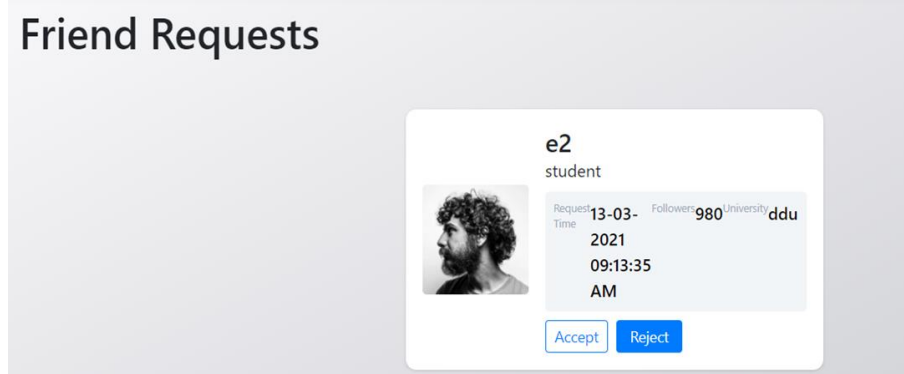
Now User Have Send Connection request To Other User



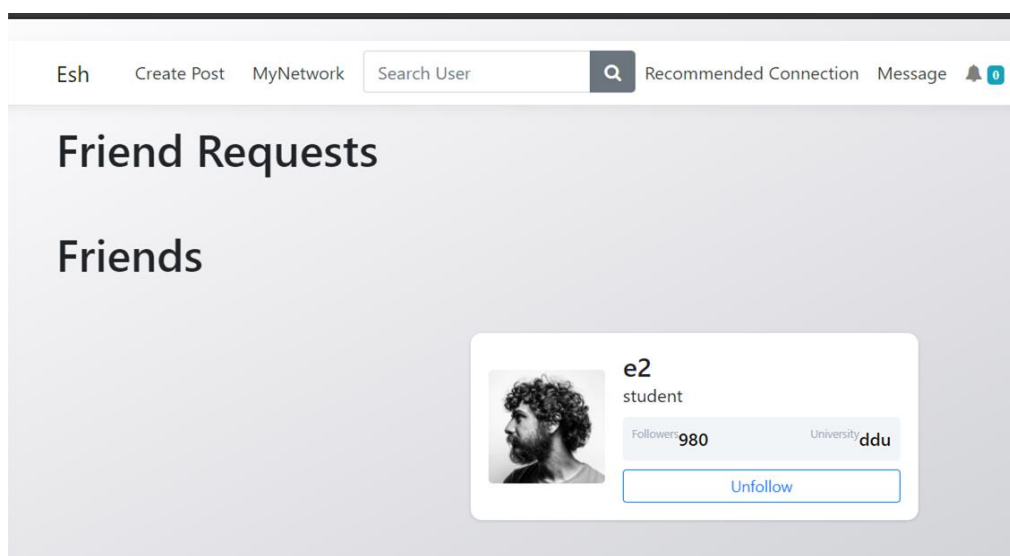
Now User Can See Recommended Connection From Their Friend's Friend



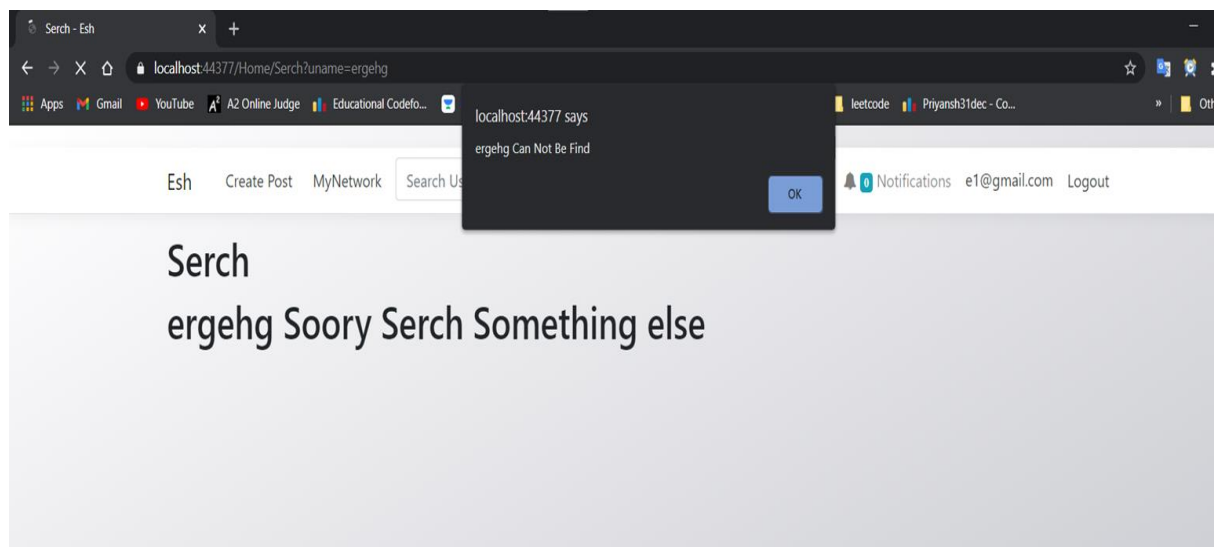
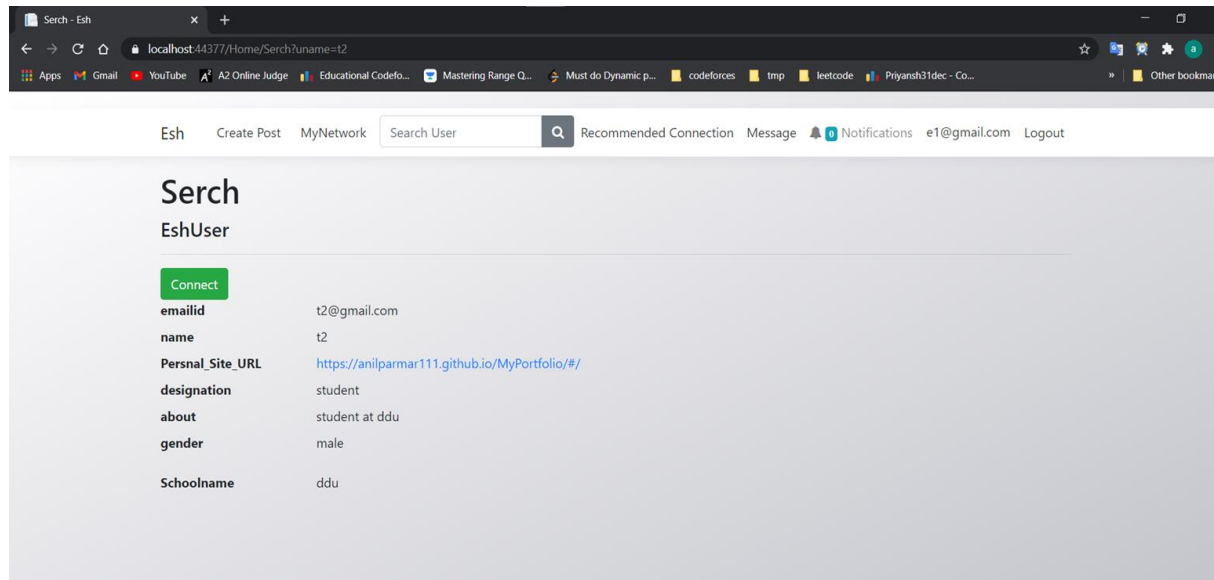
Reject Connection Request



Unfollow




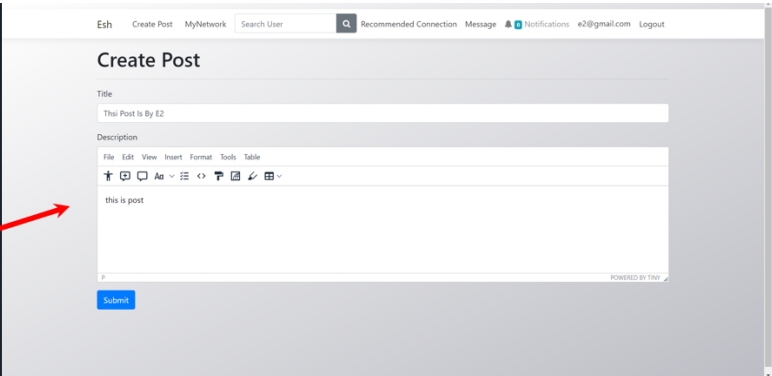
Search



Create Post

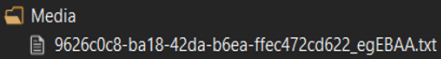
Create Post





- ★ TinyMCE Editor Is Used To Create Post
- ★ Data Of Post Is Stored In File Which Has Unique Name Generated By System


```
public static string RandomFileName(int length)
{
    const string chars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
    return new string(Enumerable.Repeat(chars, length)
        .Select(s => s[random.Next(s.Length)]).ToArray());
}
```



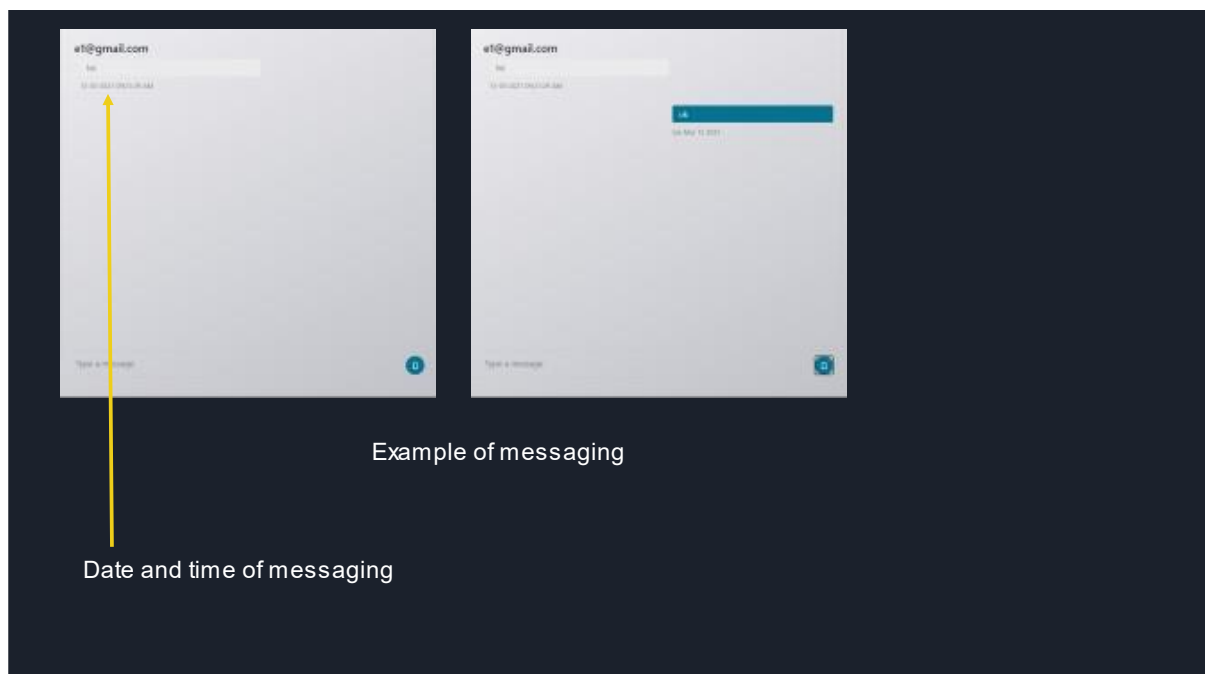
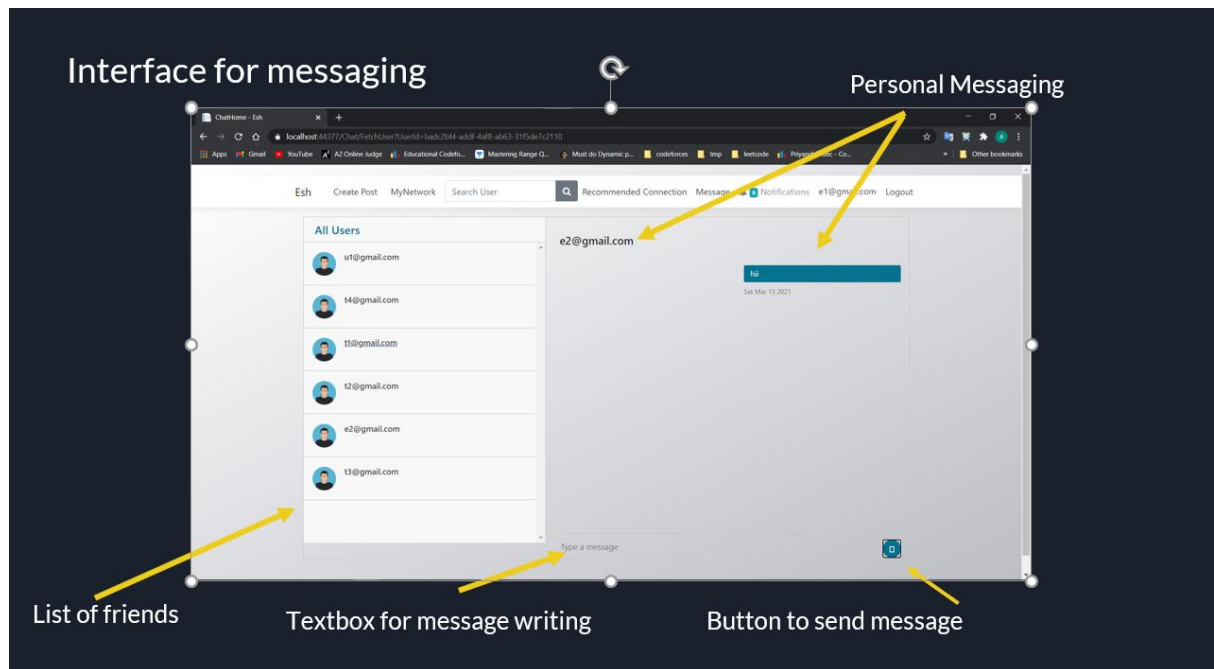
View Post

Esh Create Post MyNetwork Search User Recommended Connection Message Notifications e2 Logout

post by t1



Messaging



Logging

Logging

```
2021-03-04T13:40:24.9479378+05:30 0000001a-0004-ff00-b63f-84718c7967bb [INF] User logged in. (fdce0db4)
2021-03-04T13:40:25.0676634+05:30 0000001c-0004-ff00-b63f-84718c7967bb [INF] Index Page Of Home Has Been Accessed (37096ea7)
2021-03-04T13:40:34.9851410+05:30 0000001a-0004-ff00-b63f-84718c7967bb [INF] ChatHome Page Of Chat Has Been Accessed (f4102ebb)
2021-03-04T13:40:38.8686597+05:30 0000000a-000a-fd00-b63f-84718c7967bb [INF] FetchUser Page Of Chat Has Been Accessed (da625228)
2021-03-04T13:42:03.1058376+05:30 00000011-000a-fd00-b63f-84718c7967bb [INF] User logged in. (fdce0db4)
2021-03-04T13:42:03.1150039+05:30 00000026-0004-ff00-b63f-84718c7967bb [INF] Index Page Of Home Has Been Accessed (37096ea7)
2021-03-04T13:42:09.6433296+05:30 0000000f-0000-ff00-b63f-84718c7967bb [INF] ChatHome Page Of Chat Has Been Accessed (f4102ebb)
2021-03-04T13:42:27.0859009+05:30 00000015-000a-fd00-b63f-84718c7967bb [INF] ChatHome Page Of Chat Has Been Accessed (f4102ebb)
2021-03-04T13:42:29.9890316+05:30 0000000c-0007-ff00-b63f-84718c7967bb [INF] FetchUser Page Of Chat Has Been Accessed (da625228)
2021-03-04T13:42:40.1137060+05:30 0000000c-0001-ff00-b63f-84718c7967bb [INF] FetchUser Page Of Chat Has Been Accessed (da625228)
2021-03-04T13:42:47.2183725+05:30 00000003-0004-fd00-b63f-84718c7967bb [INF] Account Page Of Home Has Been Accessed (57f8185d)
2021-03-04T13:42:56.0737221+05:30 00000007-0006-fd00-b63f-84718c7967bb [INF] Index Page Of Home Has Been Accessed (37096ea7)
2021-03-04T13:42:58.4274135+05:30 00000010-0001-ff00-b63f-84718c7967bb [INF] Index Page Of MyNetwork Has Been Accessed (0b1a74e2)
2021-03-04T13:43:16.5752707+05:30 00000012-0001-ff00-b63f-84718c7967bb [INF] User logged out. (f9c27186)
2021-03-04T13:43:22.3732015+05:30 00000017-0000-fd00-b63f-84718c7967bb [INF] User logged out. (f9c27186)
```

- ★ LOG File Is Generated By Dependency Injection

- ★ NuGet Package :

using Microsoft.Extensions.Logging;



- ★ Dependency Injection :

```
public void Configure(IAApplicationBuilder app, IWebHostEnvironment env, ILoggerFactory loggerFactory)
```

8. Conclusion

The functionalities implemented in system after understanding all the system modules according to the requirements.

Functionalities That are successfully implemented in the system are :

- User registration containing all the necessary validation on field
- login
- user authentication
- logout
- User Can Send Connection Request
- Accept Or Reject Connection Request
- Unfollow User
- Post Editor With Tiny MCE.
- Real Time Messaging
- Recommended User
- Search User

9. Limitations and Future Enhancement

- In Our Project For Finding The Friend We Have Use Normal Search From Database But In Future For Better Time Complexity We May Extend This Functionality By Adding Staging Area And Store Friend's To The Database.
- In Post , Functionalities Like Comments , Likes , Polls Will Be Implemented.
- In messaging application functionalities for sending files and photos can be implemented

10. Reference / Bibliography

Following links and websites were referred during the development of this project.

<https://www.google.co.in>

<http://www.wikipedia.org>

<http://www.w3schools.com>

<http://www.stackoverflow.com>

[For Google Authentication](#)

<https://dotnet.microsoft.com/apps/aspn>

[et/signalr](#)