



DESIGN DOCUMENT

Product Name: Social Networking Web Application

Date: 03/21/2016 Modified: 03/25/2016

Version: 1.1

Project Group 2:

Chris Farris, Jake Fenton, Maaz Syed, Sandeep Kathulla and Venkat Sambandhan

Submitted To: XYZ Social Networking Company



INDEX

1. Introduction.....	3
1.1 Purpose.....	3
1.2 Scope.....	3
1.3 Definitions and Abbreviations.....	3
1.4 Overview.....	4
1.5 Standard Conventions.....	5
2. General Description.....	5
2.1 Tools and Technology.....	5
2.2 Working Environment for the product.....	5
2.3 High Level Block Diagram.....	5
2.4 Assumptions.....	6
3. Architecture Details.....	6
3.1 Top Level Architecture.....	6
3.2 Database Architecture.....	7
3.3 Use Cases.....	8-23
3.4 Class Diagrams.....	24
3.5 Sequence Diagrams.....	25-38
3.6 User Interface	39-46
3.7 Error Handling.....	47
3.8 References.....	47



1. Introduction

1. Introduction

1.1 Purpose

Communication is essential to the human experience and has become ever more important in the information age. Just as the telephone replaced the letter, the internet has dramatically altered the communication landscape of social networking. The central focus of this project is to create a social networking application which allows users to connect with one another in an intuitive and natural fashion. User interactions are text based, with such actions as direct communication (chat) or posting to a public board type structure (Newsfeed).

This document is intended to give a thorough and detailed outline of the project's scope, requirements, and specifications in order to be approved or altered by the client. This includes all functional and non-functional requirements of the application. The intended audience of the document includes the client, developers, testers, and users, but may be useful to any party which desires to understand the application's functionality and purpose.

With this web application people will be able to connect with friends and family by utilizing an application that provides ease of use with private and/or public messages. The web application will allow for people to contact friends and family no matter how near or far they are from that friend or family member.

1.2 Project Scope

The scope of this web application is to allow for a user to establish a friend's list with at least 100 friends and be able to message any one of those friends privately, or via their own page. Also, either of the friends can be in any region of the world, permitted they have internet and electricity in order to connect to their device.

1.3 Definitions and Abbreviations

Client	The client consists of the Spring 2016 CSCI-P465 Instructor and Co-Instructor at Indiana University.
User	A user will have a registered identity in the application which allows them to use the application and connect with other users.
Stakeholder	Any party involved with the development process of the project



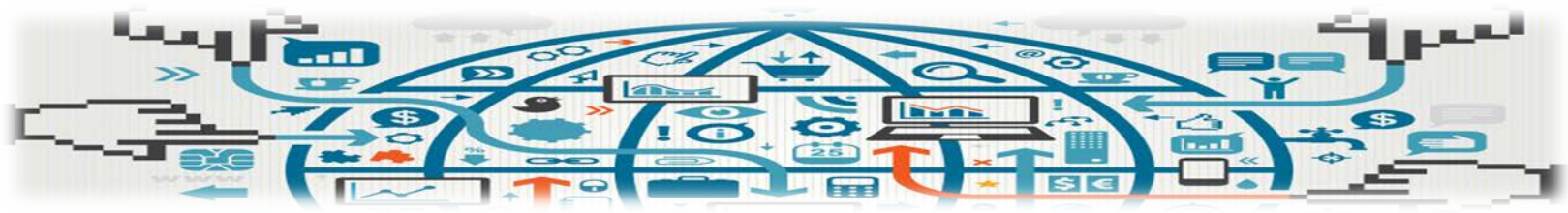
Friend	A designation given to a second party from the user which allows communication. Friend status must be requested by one party and accepted by the recipient of the request. After acceptance, the two parties are designated as friends and additional communication functionality is made available.
Chat	A private system of correspondence between friends. This communication can only be read or received by the parties involved.
Inbox	A system which stores text communication sent by the user and received by the user. The inbox will store the data transmitted by the chat system.
Newsfeed	A page which displays public posts made by the friends of the user. These posts are displayed in order of occurrence, with the most recent post being displayed first.
Group	Groups may be created and joined by users. A group has its own page which consists of a Newsfeed specific to group members. Only group members may post to or see the group specific Newsfeed. Additionally, groups may be searched for by users to join.

1.4 Overview

The overall idea of this web application is to allow for a user to stay connected with friends and family that are not nearby to the user. This will allow for the user to share information with their friends and family, without having to travel to them in order to keep them up to date. Also, this application allows for the user to send the information to multiple people at once, via a group page or group message.

- The user logs into their account and the UI is displayed.
- The user is given a news feed of information from the friends.
- The user is given the opportunity to click on any of the pages from the newsfeed, or on their own page.
- The user is also able to send a message to any of their friends, or post a message via inbox or in a group page.

After the user is finished with their tasks on their account, the user can logout in order to attempt to keep the account secure.



1.5 Standard Conventions

- SDK – Software development kit.
- JDK – Java development kit.
- JRE – Java runtime environment.
- OpenGL – Open Graphics Library.
- API – Application Program Interface.
- SQL – Structured query language.

2. General Description

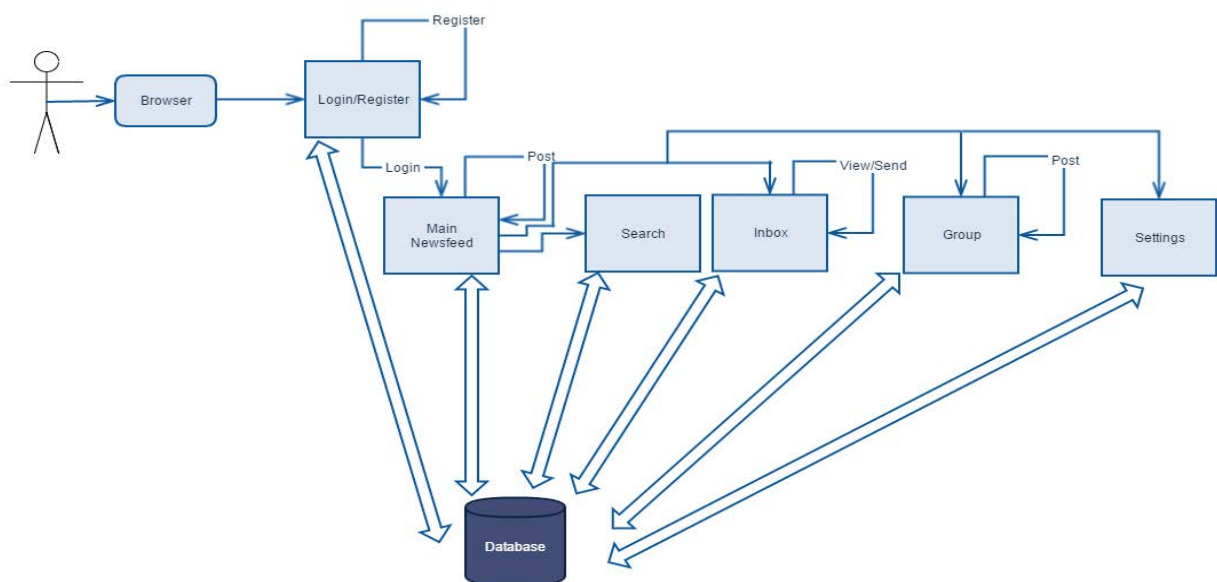
2.1 Tools and Technology

- Java
- SQL

2.2 Working Environment for the product

The project is designed to function in a web browser format. Two standard browsers are targeted in the design: Internet Explorer 11.0.29 and Chrome 49.0.2623. The user must also have internet access to the project servers and must have input devices such as a keyboard and mouse.

2.3 High Level Block Diagram



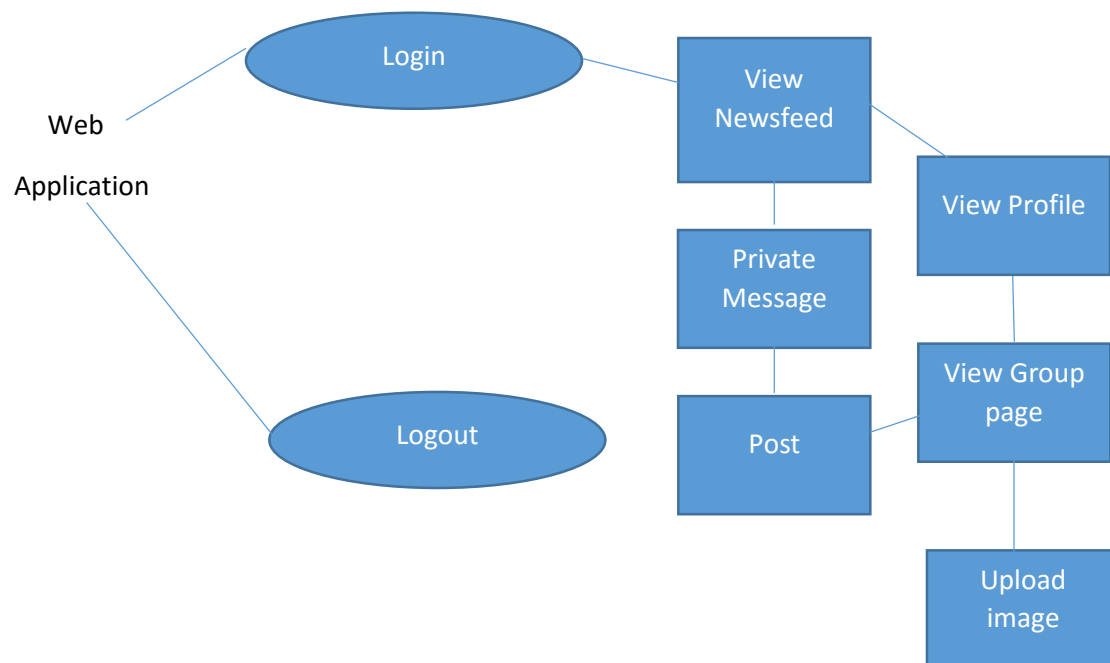


2.4 Assumptions

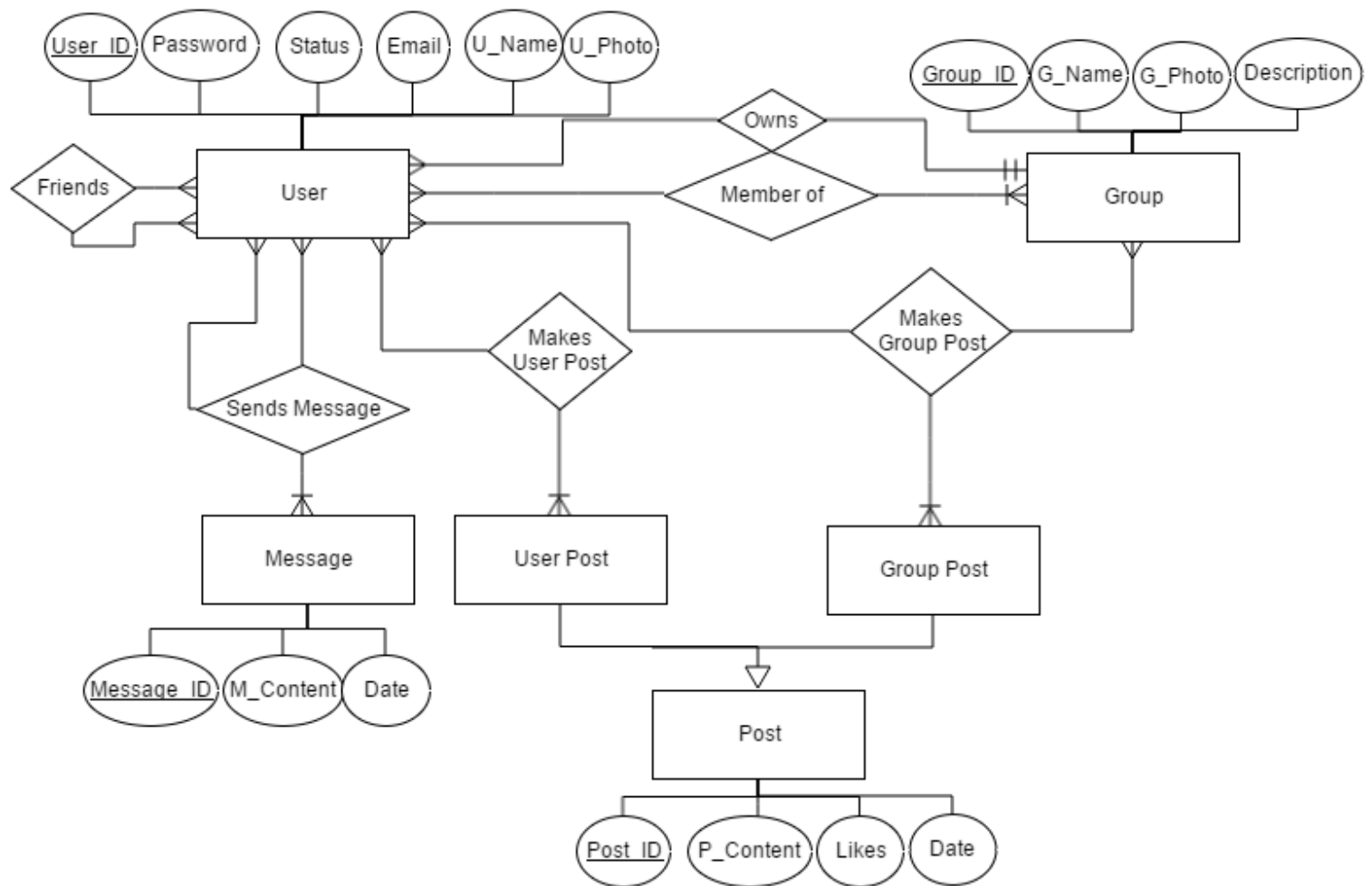
- User has internet access
- User is utilizing the specified versions of Chrome or Internet Explorer
- User is able to read and to input with a keyboard, mouse, or similar device.

3. Architecture Details

3.1 Top Level Architecture



3.2 Database Architecture





3.3 Use Cases

US-1 Registration

ID and Name: Registration			
Created by:	Project group 2	Date Created:	3/4/2016
Primary Actor	User, Server		
Description	User creates an account by filling up a sign up form. The User enters his/her first name, last name, date of birth, email-id, password and profile picture.		
Trigger	User clicks on the signup button on the welcome screen.		
Pre-condition	User has accessed the application.		
Post-condition	A record is made for the User in the server. The user should login again using email-id (user name) and password		
Steps	<ol style="list-style-type: none"> 1) User clicks on the signup button on the welcome screen. 2) The application sends a sign-up form window. 3) User fills up the form (first name, last name, date of birth, email-id, password and profile picture) and submits it. 4) The details provided by the User are stored in the server. 		
Exceptions	<ul style="list-style-type: none"> • User account already exists, username is already in use, or password is too weak. • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	The Server creates an entry/record for the new User.		



US-2 Delete account

ID and Name: Delete account			
Created by:	Project group 2	Date Created:	3/4/2016
Primary Actor	User		
Description	User deletes account.		
Trigger	User clicks on the SETTINGS button in the Homepage. Once there, the User clicks on the DELETE MY ACCOUNT button.		
Pre-condition	User has been logged into the application by the server and is in Homepage.		
Post-condition	User has deleted his/her account.		
Steps	<ol style="list-style-type: none"> 1) User goes to the Homepage. 2) User clicks on the SETTINGS button. 3) User clicks on the DELETE MY ACCOUNT. 		
Exceptions	<ul style="list-style-type: none"> • Server is down and not reachable. • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	The user has successfully deleted his/her account.		



US-3 Login

ID and Name: Login			
Created by:	Project group 2	Date Created:	3/4/2016
Primary Actor	User, Server		
Description	User logs into the application with username and password.		
Trigger	User clicks the login button in the welcome screen.		
Pre-condition	User opens the w and clicks the login button.		
Post-condition	Server logs in the User.		
Steps	<ol style="list-style-type: none"> 1) User sends LOGIN request to the server. 2) Server responds back requesting for login credentials. 3) User sends login credentials. 4) IF login credentials are verified <ul style="list-style-type: none"> • THEN Server logs in the User • ELSE Server throws an error “INVALID login credentials” 		
Exceptions	<ul style="list-style-type: none"> • INVALID login credentials were provided by the User. • Server is down and not reachable. • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	The User has a record in the server and hence he/she logs into the application		



US-4 Forgot password

ID and Name: Forgot password			
Created by:	Project group 2	Date Created:	3/4/2016
Primary Actor	User, Server		
Description	User has forgotten his/her password and wants to reset it.		
Trigger	User clicks on forgot password in the login page.		
Pre-condition	User has forgotten his/her password.		
Post-condition	User successfully changed the password and logged into his/her account		
Steps	<ol style="list-style-type: none"> 1) User clicks on forgot password in the login page. 2) The application asks for security question. 3) When user enters correct answer he can enter a new password. 4) The new password will be updated in the database. 5) The user now logs in using his/her new password 		
Exceptions	<ul style="list-style-type: none"> • Password is too weak. • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	The user successfully changed his/her password.		



US-5 Logout

ID and Name: Logout			
Created by:	Project group 2	Date Created:	3/4/2016
Primary Actor	User, Server		
Description	User will exit the application.		
Trigger	User clicks the logout button.		
Pre-condition	The User has been logged into the application by the server.		
Post-condition	The User is asked for confirmation. The user will be logged out of the application by the server if user confirms.		
Steps	1) User clicks the logout button 2) Application closes		
Exceptions	<ul style="list-style-type: none"> • Server is down and not reachable. • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	The User has been logged out of the application by the server.		



US-6 Search for friends and send Requests

ID and Name: Search for friends and send Requests			
Created by:	Project group 2	Date Created:	3/4/2016
Primary Actor	User A, User B		
Description	User A searched for User B and sent friend request		
Trigger	User A clicks on the SEARCH ICON from the MENU option and then types the first and last name of User B.		
Pre-condition	User has been logged into the application by the server and is in the application homepage.		
Post-condition	User A has searched and either found and sent a friend request to User B or not found User B.		
Steps	<ol style="list-style-type: none"> 1) User A clicks on the SEARCH ICON 2) User A then types the name of user B 3) User A gets list of all person's with name B. 4) User A can view the details of all the users (profile picture, email-id and status) 5) After identifying correct person, User A sends a friend request to user B. 		
Exceptions	<ul style="list-style-type: none"> • User B has blocked User A. • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	User A was successfully able to find and send friend request to User B.		



US-7 Accept/Delete friend requests

ID and Name: Accept friend requests			
Created by:	Project group 2	Date Created:	3/4/2016
Primary Actor	User A, User B		
Description	User A adds User B to his/her friend list.		
Trigger	User B sends friend request to user A.		
Pre-condition	<ul style="list-style-type: none"> User A has been logged into the application and is in the home page. The friends icon has lit up on user A's homepage 		
Post-condition	User A has added User B to his/her friend list and is now able to chat, check his/her status updates/profile picture, etc.		
Steps	<ol style="list-style-type: none"> 1) User A clicks on Friends icon 2) User A can view the details of all the users (profile picture, email-id and status) 3) User A clicks on the accept button or reject button. 4) If accepted user B is now listed in User A's friend list. 		
Exceptions	<ul style="list-style-type: none"> User B has blocked User A. User B has deleted his/her account. Application closes due to lack of memory or Web Browser closes. 		
Expected Output	User B appears in User A's friend list if accepted.		



US-8 Friends list

ID and Name: Friends list			
Created by:	Project group 2	Date Created:	3/4/2016
Primary Actor	User		
Description	User views/delete friends from his/her friend's list.		
Trigger	User clicks on Friends button.		
Pre-condition	User has been logged into the application and is in the home page.		
Post-condition	<ul style="list-style-type: none"> User has viewed/deleted friends from his/her friend's list. If the user clicks on the view button, he will be redirected to the user's information page. If the user clicks on the delete button, the respective user is removed from his/her friends list. 		
Steps	<ol style="list-style-type: none"> 1) User A clicks on Friends icon. 2) User A can view the details of all the users (profile picture, email-id and status) 3) User A can click on the view/delete button. 		
Exceptions	<ul style="list-style-type: none"> Application closes due to lack of memory or Web Browser closes. 		
Expected Output	User A has viewed/deleted User B.		



US-9 View Newsfeed

ID and Name: View News-Feed			
Created by:	Project group 2	Date Created:	3/4/2016
Primary Actor	User		
Description	User views notifications from the newsfeed.		
Trigger	User scrolls on the newsfeed to view the latest notifications.		
Pre-condition	User has been logged into the application by the server and is in the application homepage.		
Post-condition	User views the notifications of the recent activities performed by his/her friends.		
Steps	<ol style="list-style-type: none"> 1) User logs into his/her account. 2) User views notifications of latest activities. 		
Exceptions	<ul style="list-style-type: none"> • Application closes due to lack of memory or Web Browser closes. • The number of notifications has exceeded the maximum allowed limit (50). 		
Expected Output	The User has successfully viewed the notifications.		



US-10 Like posts in Newsfeed

ID and Name: Like posts in Newsfeed			
Created by:	Project group 2	Date Created:	3/9/2016
Primary Actor	User		
Description	<p>User views posts on newsfeed and decides to like particular post/posts.</p> <p>Note: When user X clicks the like button, a comment stating that user X likes the post will be added. "User X" will be hyperlinked to user X's information page.</p>		
Trigger	User scrolls on the newsfeed and clicks on the like button of particular post.		
Pre-condition	User has been logged into the application by the server and is viewing the posts		
Post-condition	User has liked the posts.		
Steps	<ol style="list-style-type: none"> 1) User hits the like button. 2) Comment stating that user X likes the post will be added. 3) When user X (hyperlink) is clicked, it will redirect to information page (containing profile picture, status and Email-id) of user X 		
Exceptions	<ul style="list-style-type: none"> • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	The User has successfully liked the post.		



US-11 View Inbox

ID and Name: View Inbox			
Created by:	Project group 2	Date Created:	3/9/2016
Primary Actor	User		
Description	Inbox contains the messages sent and received by the user.		
Trigger	User clicks on the Inbox button to view the messages.		
Pre-condition	User has been logged into the application by the server and is in the application homepage.		
Post-condition	User has viewed the messages from the Inbox.		
Steps	<ol style="list-style-type: none"> 1) User clicks on the Inbox button on the Homepage. 2) User clicks on the message to view. 		
Exceptions	<ul style="list-style-type: none"> • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	The User has successfully viewed the message from his/her Inbox.		



US-12 Send message from Inbox

ID and Name: Send message from inbox			
Created by:	Project group 2	Date Created:	3/9/2016
Primary Actor	User A, User B, Server		
Description	User A sends a message to User B via the Inbox.		
Trigger	User A goes to the Inbox tab. Once there, the User clicks on User B's name and enters the email window. From there on, User A types and sends messages to User B.		
Pre-condition	User A has been logged into the application by the server and is in Inbox.		
Post-condition	User A has sent message(s) via Inbox to User B.		
Steps	<ol style="list-style-type: none"> 1) User A goes to the Inbox tab. 2) User A clicks on User B's name and enters the email window. 3) User A types the message in the text box and clicks on the arrow button. 4) Server receives the message and sends the message to User B. 		
Exceptions	<ul style="list-style-type: none"> • User B has blocked User A. • User B has deleted his/her account. • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	User A has sent messages to User B through the Inbox.		



US-13 Create Group page and Add profile picture/Add users to group page

ID and Name: Create Group Page and Add profile picture/Add users to group page			
Created by:	Project group 2	Date Created:	3/9/2016
Primary Actor	User		
Description	<p>User creates a group page. Adds a group picture and other Users from his/her friend list.</p> <p>Visibility: This group page can be viewed by only the existing members of the group.</p>		
Trigger	The user clicks on the CREATE GROUP PAGE button and adds group picture, users to the group.		
Pre-condition	User (the one who wants to create the group page) has been logged into the application by the server and is in the home page.		
Post-condition	User creates a new group page and adds group picture, members to it.		
Steps	<ol style="list-style-type: none"> 1) User goes to the home page 2) User clicks on the NEW GROUP PAGE button 3) User types in the name of the group and clicks on the NEXT button. 4) User clicks on the CREATE button. 5) User adds a group picture. 6) User adds users to the group. 		
Exceptions	<ul style="list-style-type: none"> • The number of group members has exceeded the maximum allowed limit (50). • Application closes due to lack of memory or Web Browser closes. 		
Expected Output	The user creates a new group page and adds group members from his/her contact list.		



US-14 Leave a group page

ID and Name: Leave a group page			
Created by:	Project group 2	Date Created:	3/9/2016
Primary Actor	User		
Description	User exits a group page.		
Trigger	User clicks and holds on the group page that he wants to leave and then clicks on the EXIT GROUP PAGE.		
Pre-condition	User has been logged into the application by the server and is in the group page.		
Post-condition	User has left the selected group page and the remaining members of the group will be notified of this activity.		
Steps	<ol style="list-style-type: none"> 1) User clicks the group page that he wants to leave. 2) User enters the group 3) User clicks on the EXIT GROUP PAGE button. 4) Remaining group members are sent notifications about this activity. 		
Exceptions	Application closes due to lack of memory or Web Browser closes.		
Expected Output	The User has successfully left the group page.		



US-15 Change Profile Settings

ID and Name: Change Profile Settings			
Created by:	Project group 2	Date Created:	3/9/2016
Primary Actor	User		
Description	The user can change his settings: profile picture and status.		
Trigger	User clicks on the Settings button to change his/her settings.		
Pre-condition	User has been logged into the application by the server and is in the settings page.		
Post-condition	User has changed his/her settings as desired.		
Steps	1) User clicks on the Settings button on the Homepage. 2) User changes his profile picture/status as desired.		
Exceptions	<ul style="list-style-type: none"> Application closes due to lack of memory or Web Browser closes. 		
Expected Output	The User has successfully changed his profile settings.		



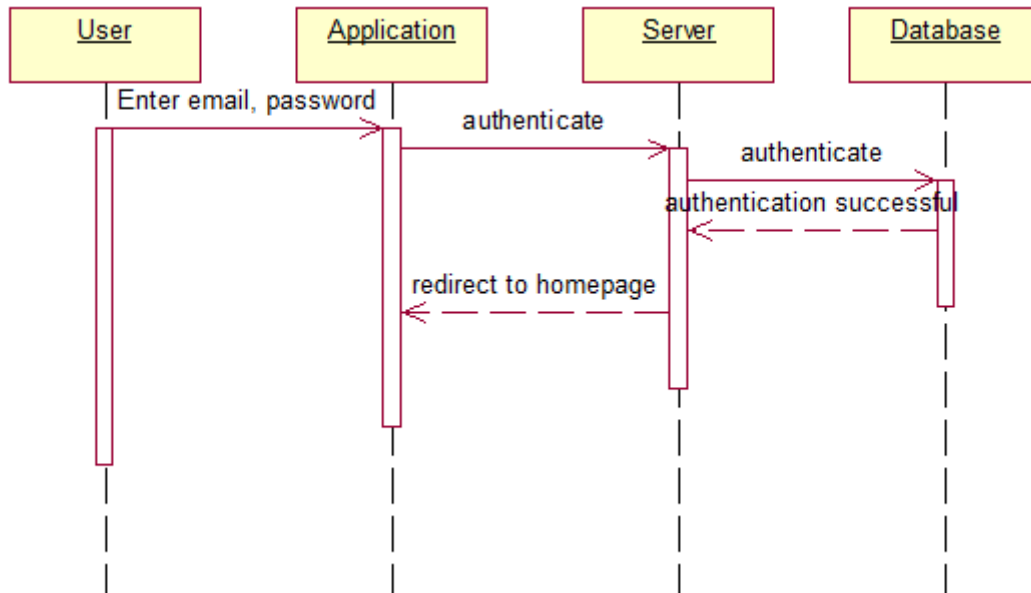
US-16 Information Page

ID and Name: Information Page			
Created by:	Project group 2	Date Created:	3/9/2016
Primary Actor	User A, User B		
Description	The information page contains basic information about the user which will include a display picture, status and email-id.		
Trigger	User B clicks on User A (hyperlinks present in Search/Newsfeed/Comments) to view his/her information page.		
Pre-condition	User B has been logged into the application.		
Post-condition	User B has landed on User A's information page.		
Steps	1) User B clicks on the hyperlink button. 2) User B views User A's information page.		
Exceptions	<ul style="list-style-type: none"> Application closes due to lack of memory or Web Browser closes. 		
Expected Output	User B has successfully viewed User A's information.		

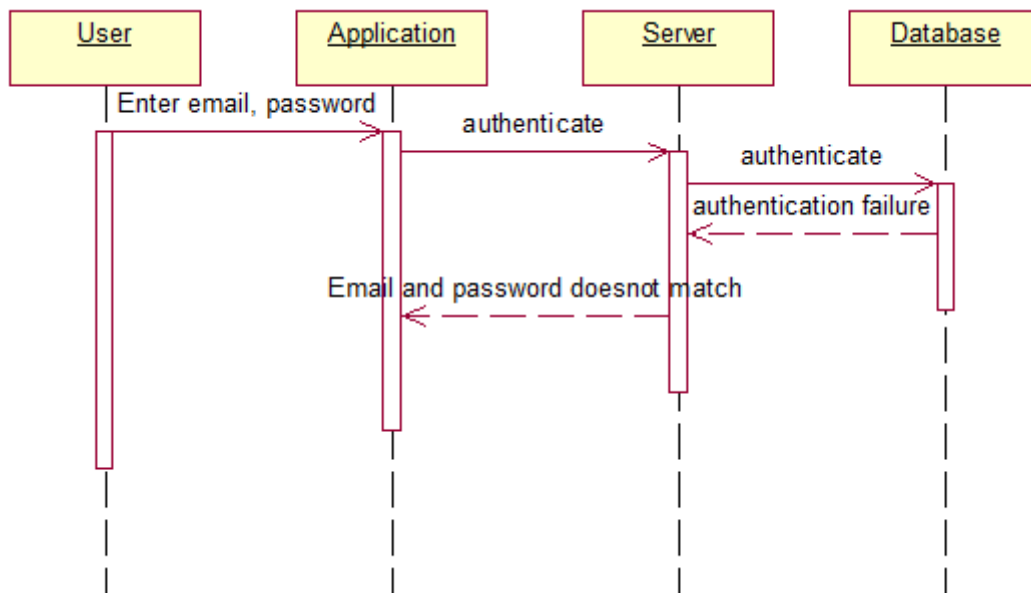


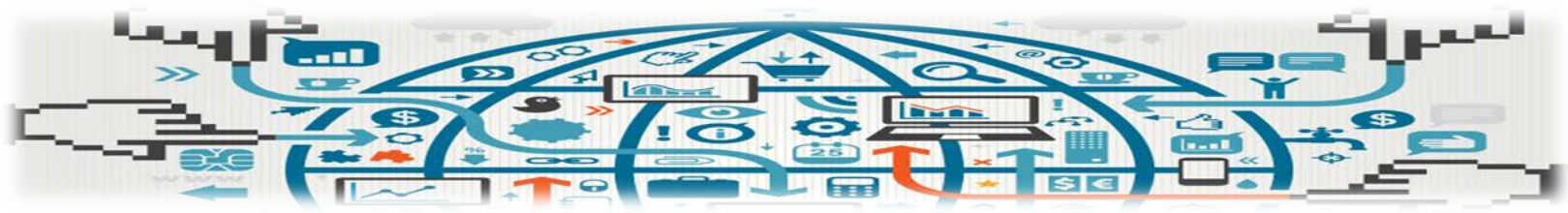
3.5 Sequence Diagrams

SD - 1 Login

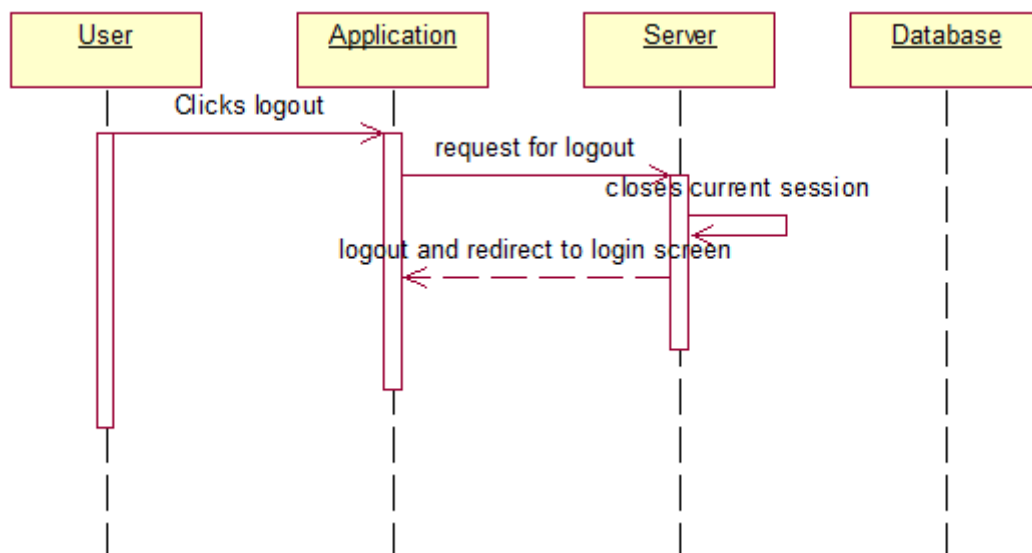


SD - 2 Login alternative

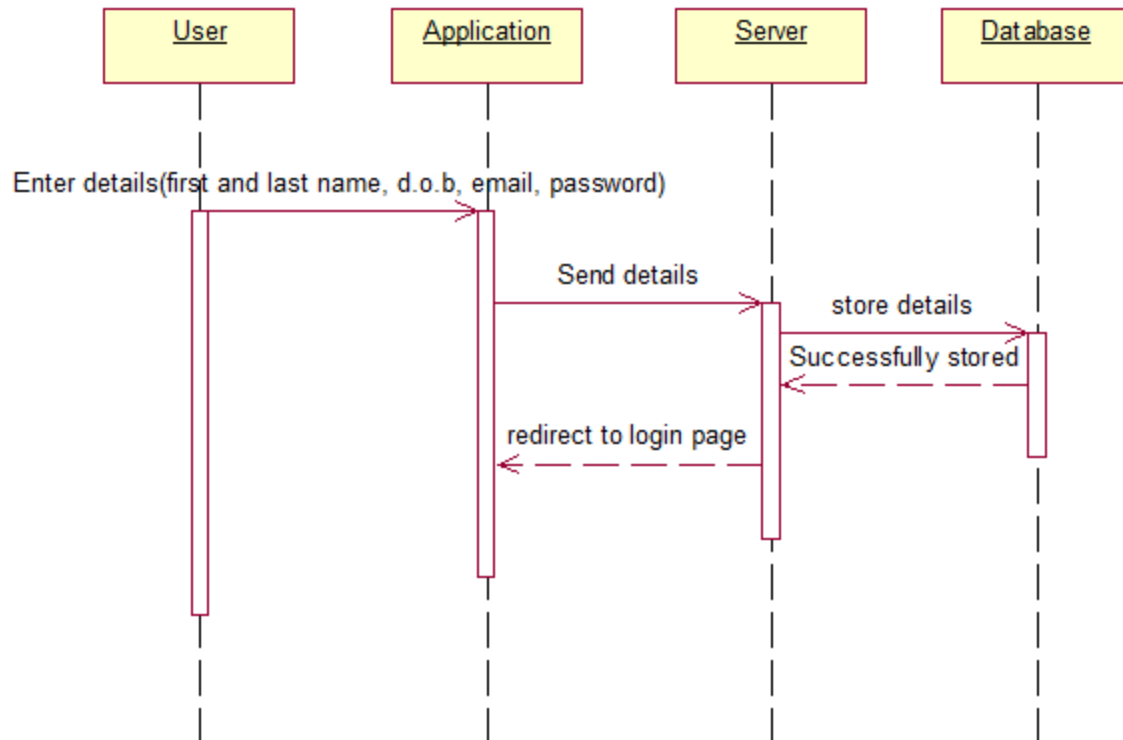


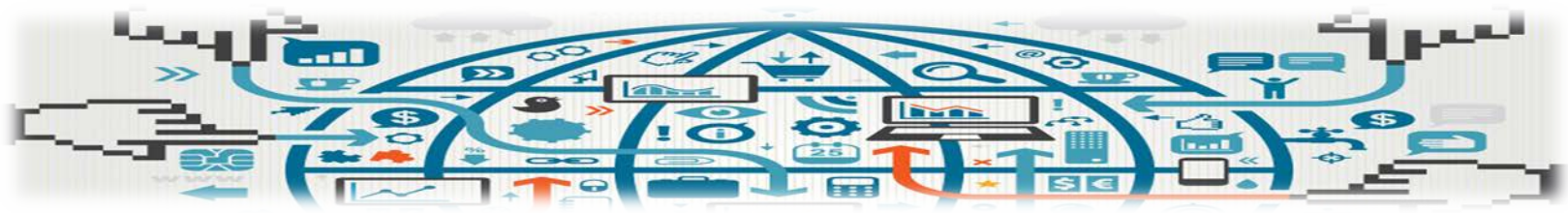


SD - 3 Logout

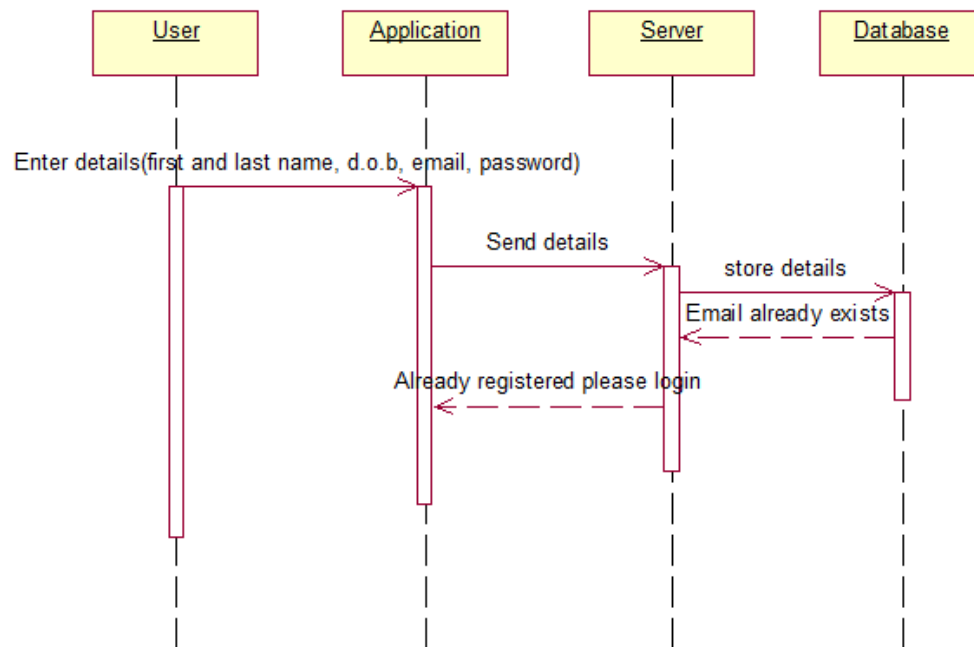


SD - 4 Registration

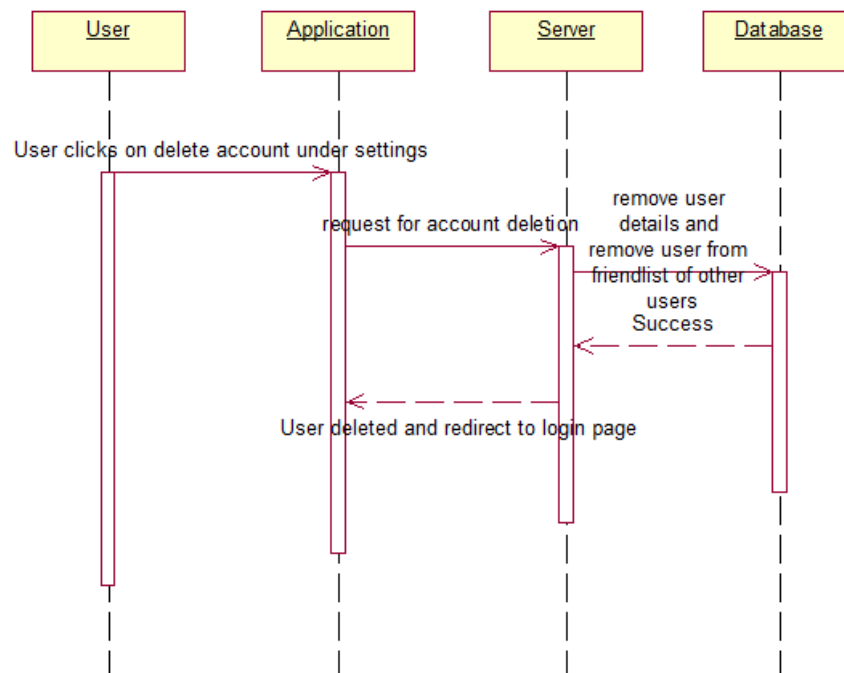




SD - 5 Registration alternative

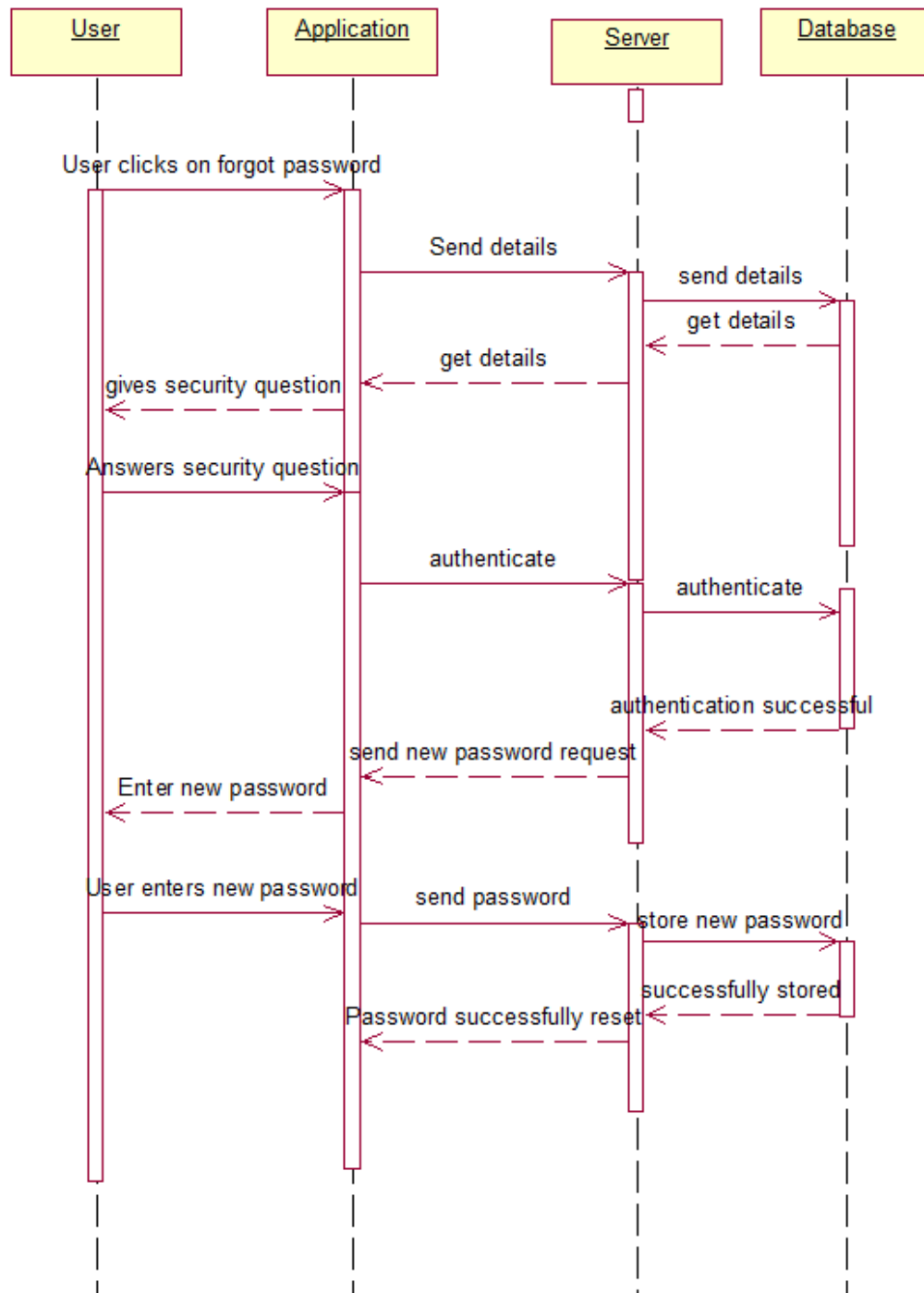


SD - 6 Remove user



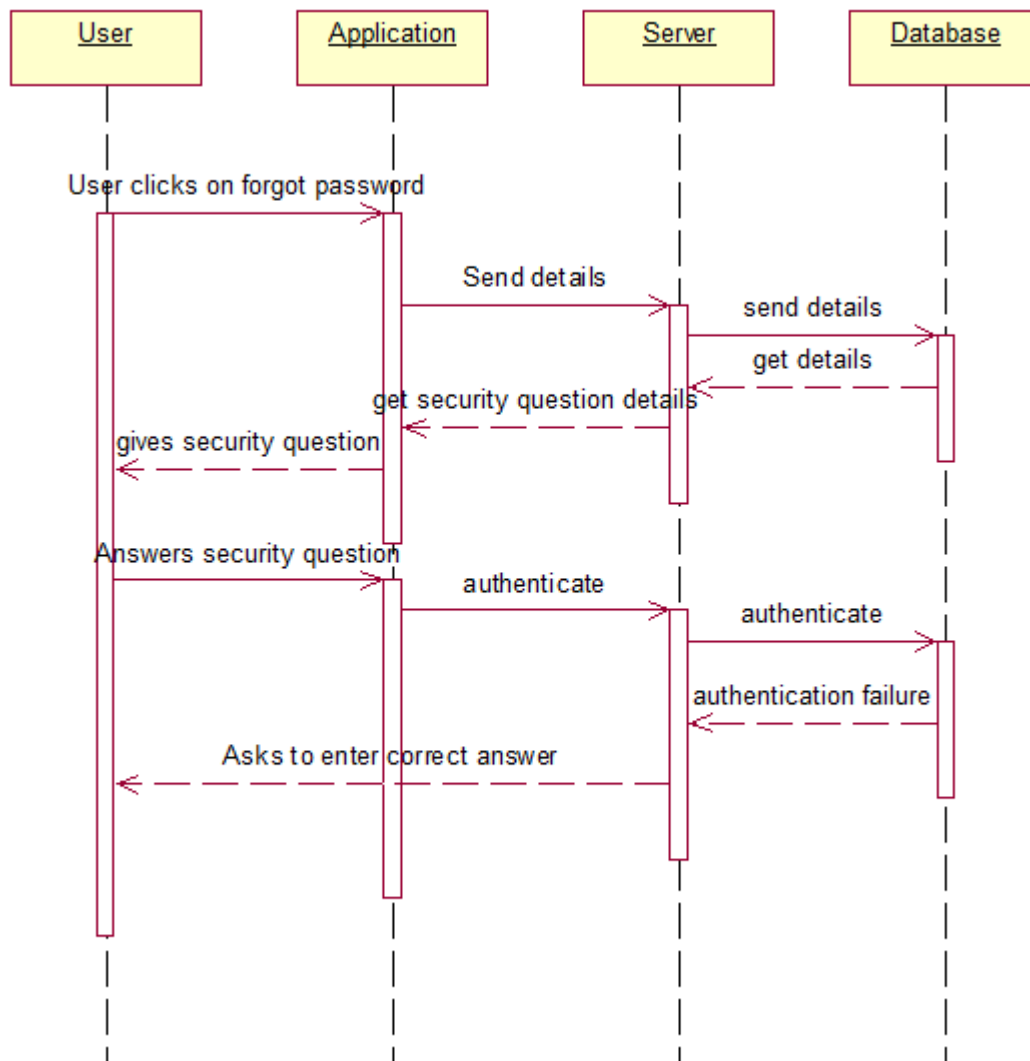


SD - 7 Forgot password



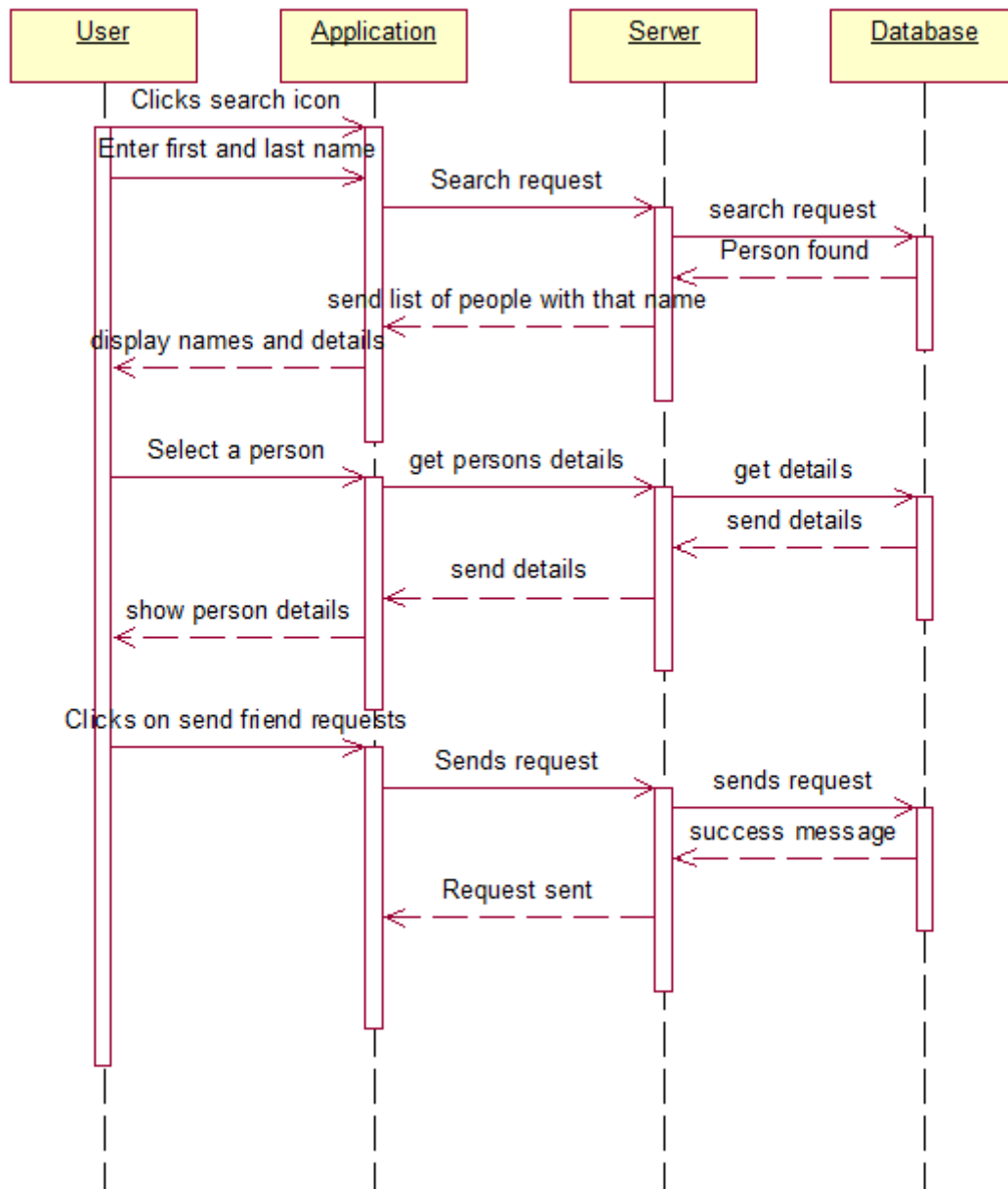


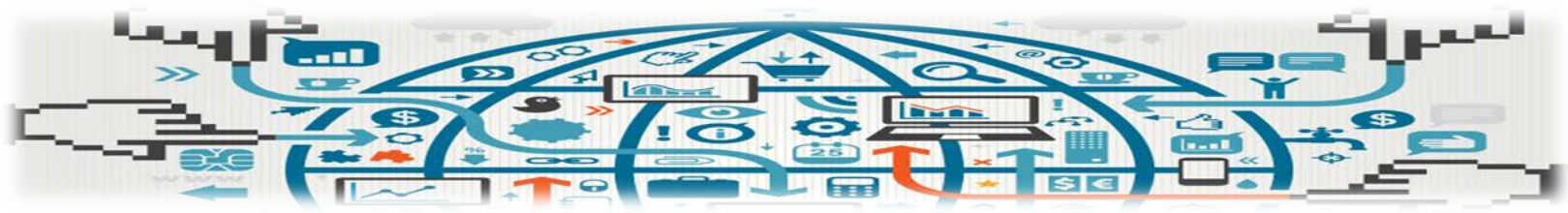
SD - 8 Forgot password alternative



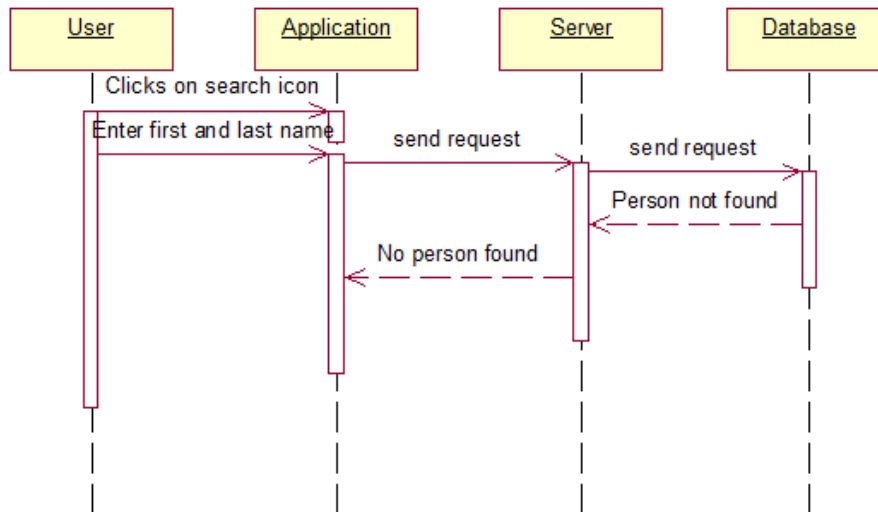


SD - 9 Search and send requests

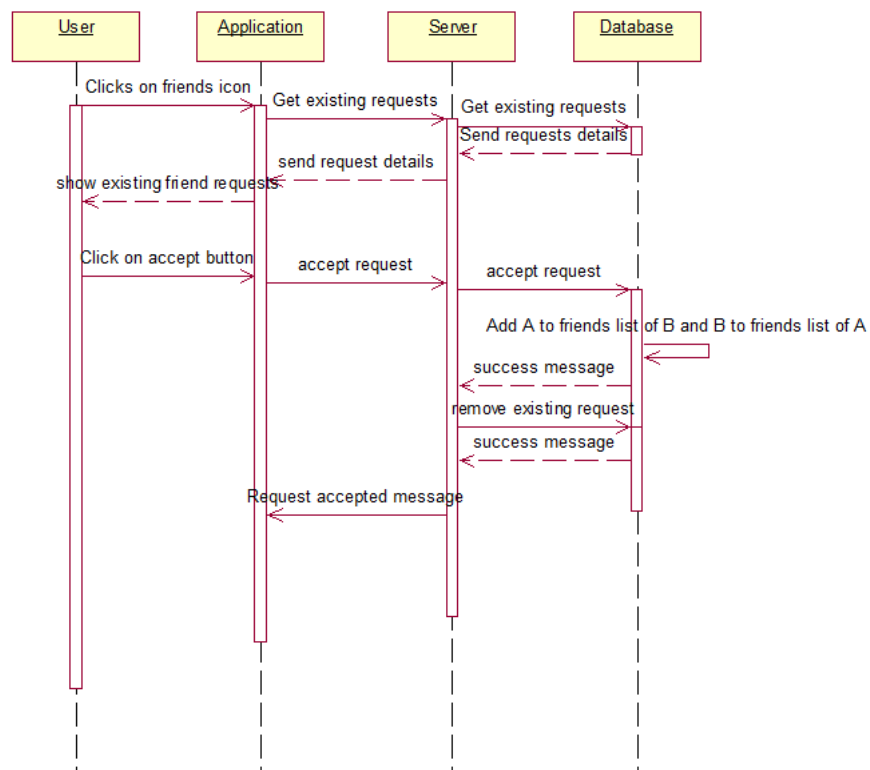


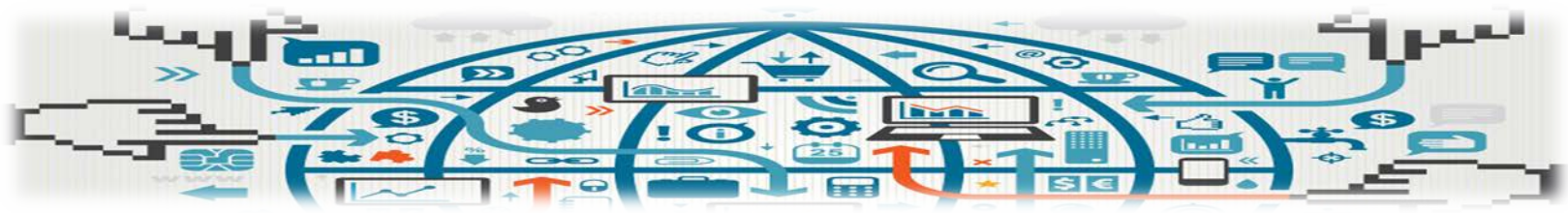


SD - 10 Search and send requests alternative

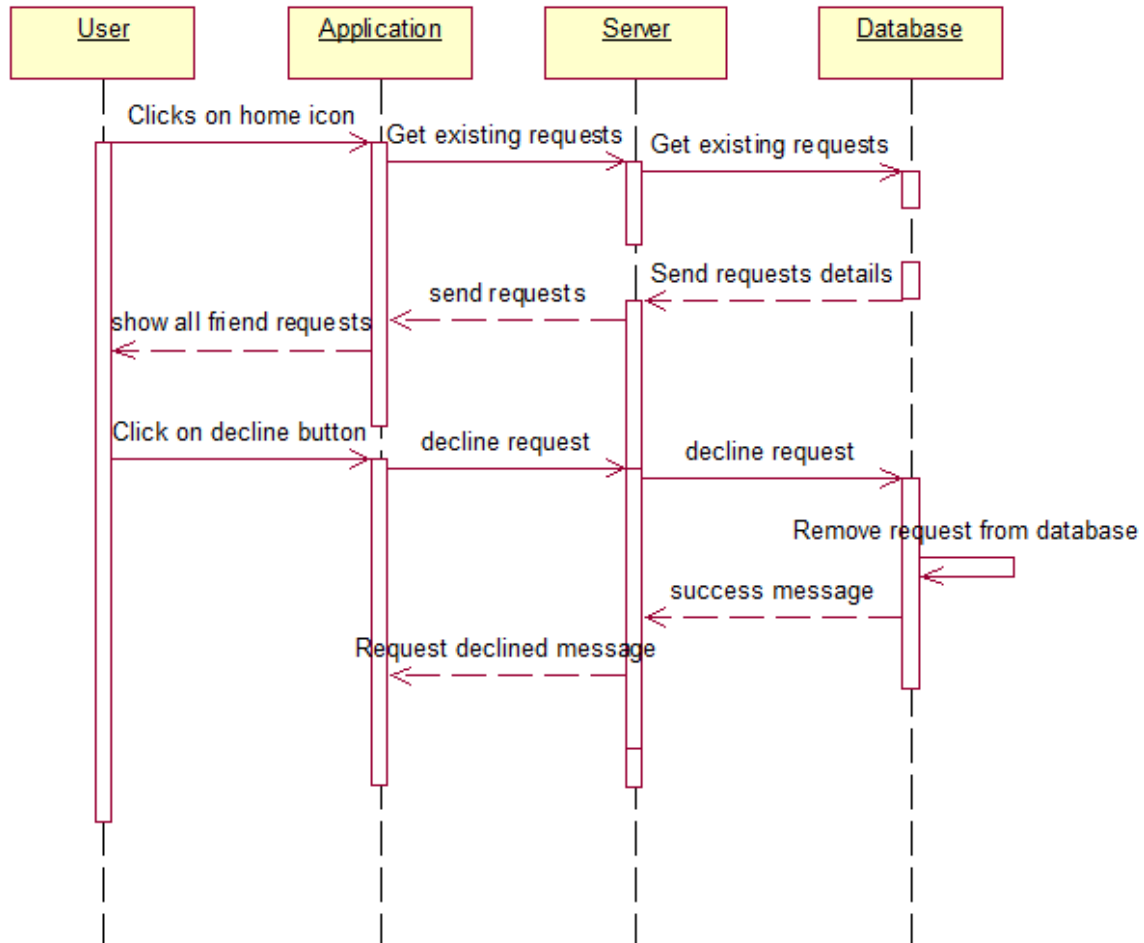


SD - 11 Accept friend request



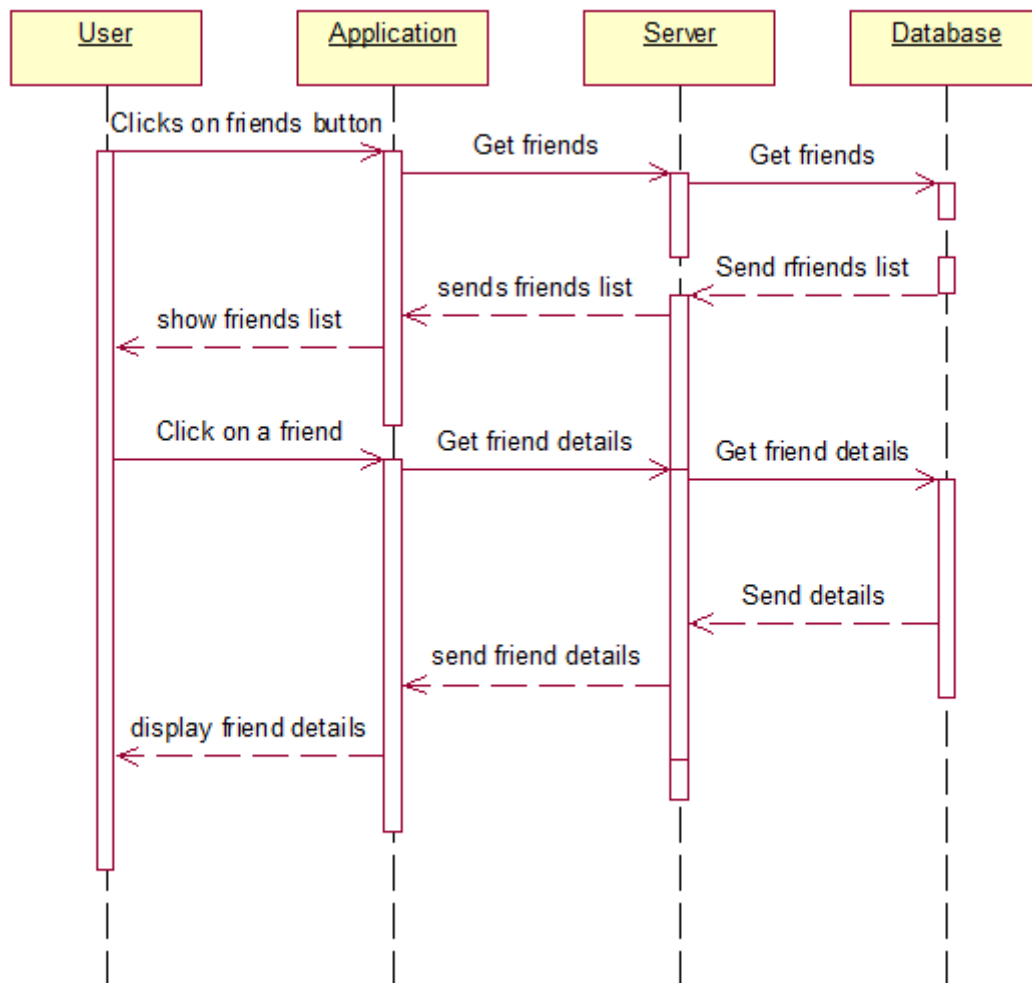


SD - 12 Decline friend request



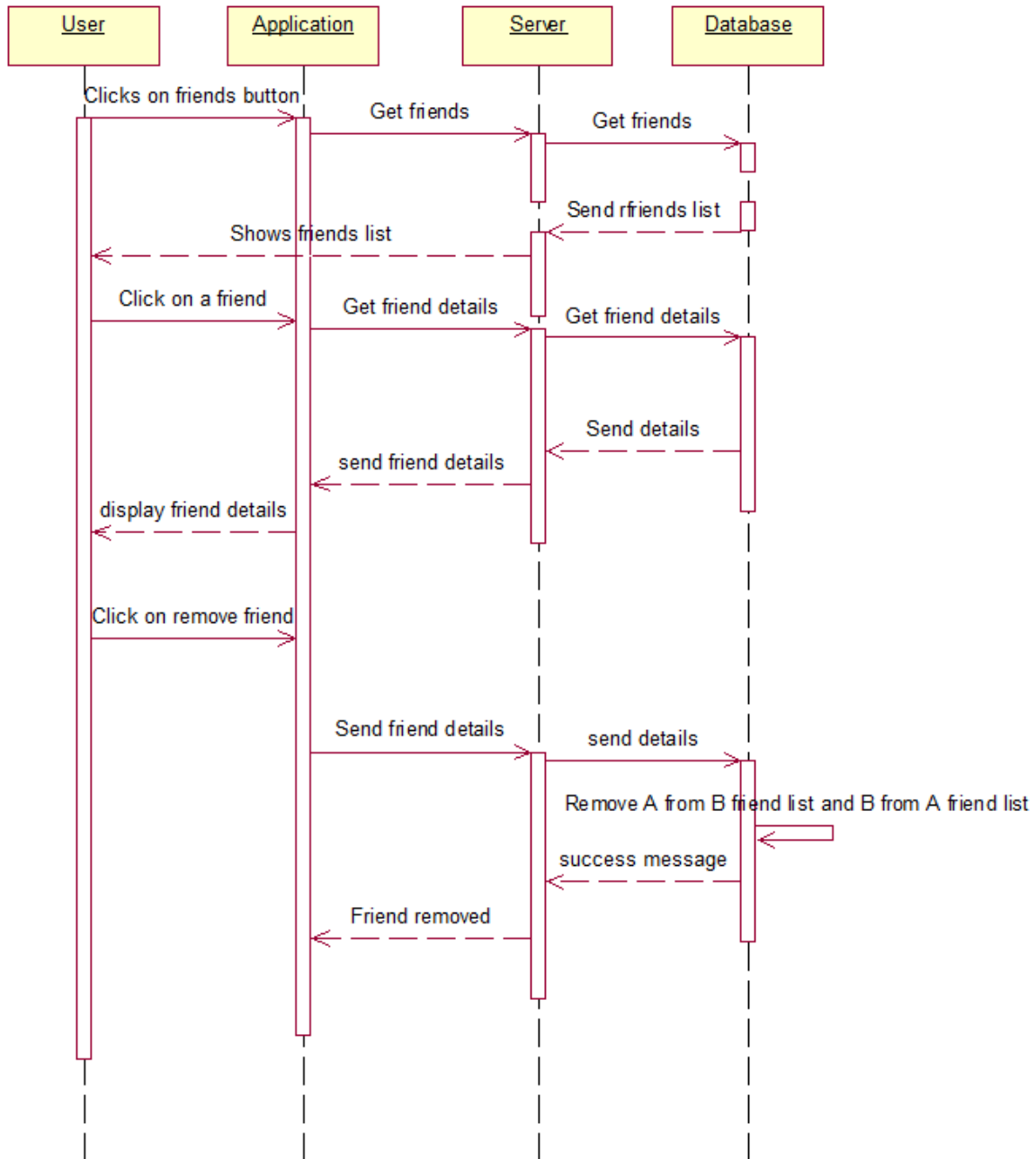


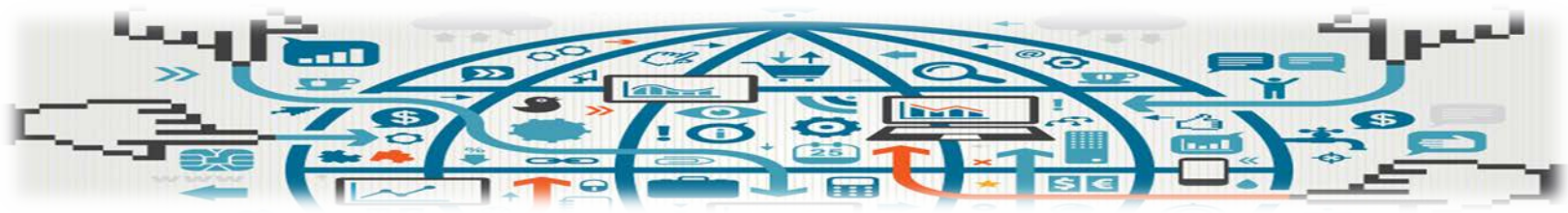
SD - 13 View friends



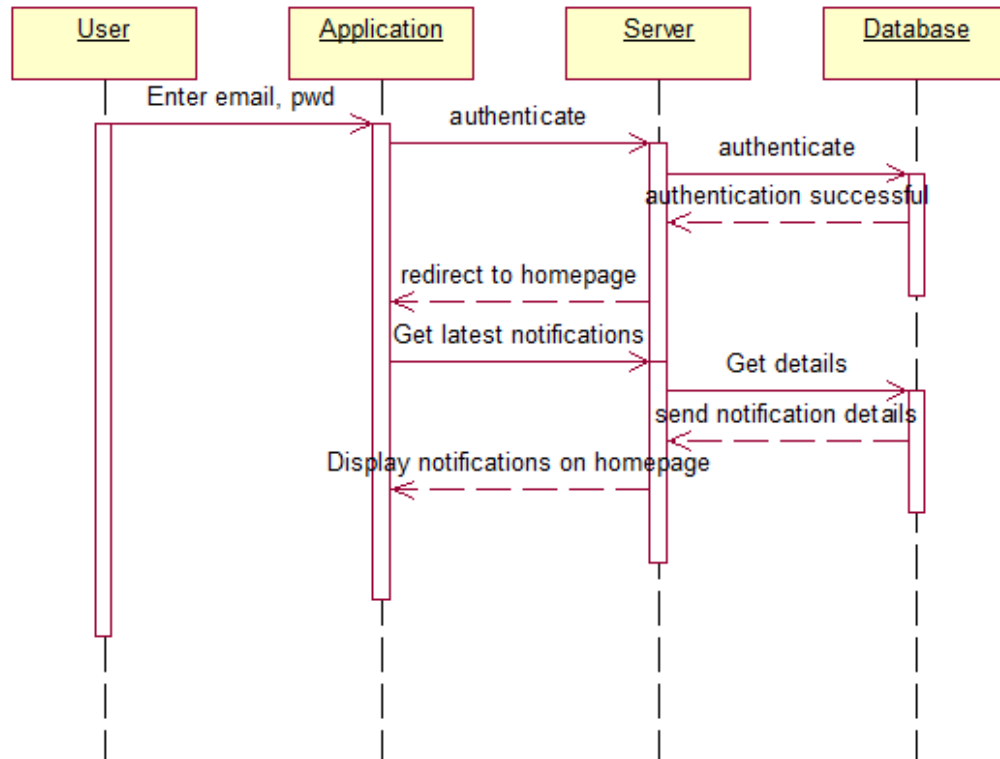


SD - 14 Delete friends

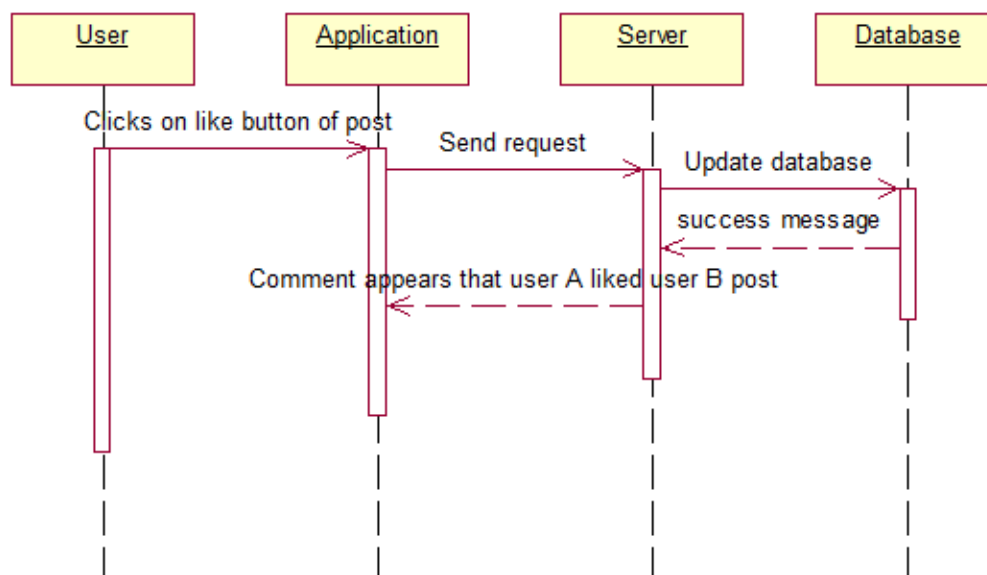


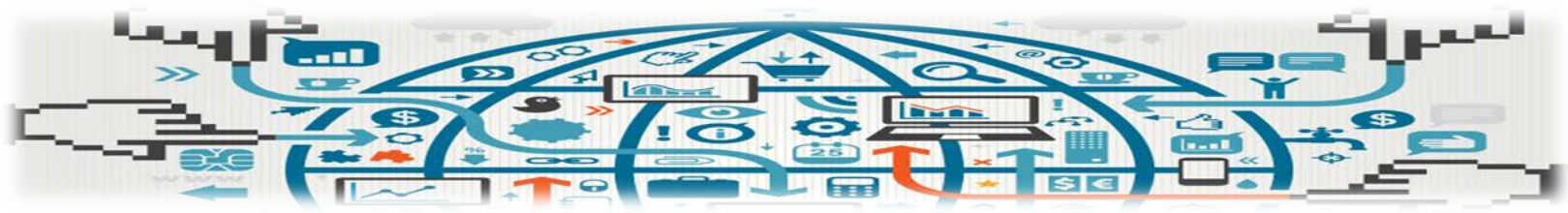


SD - 15 View newsfeed

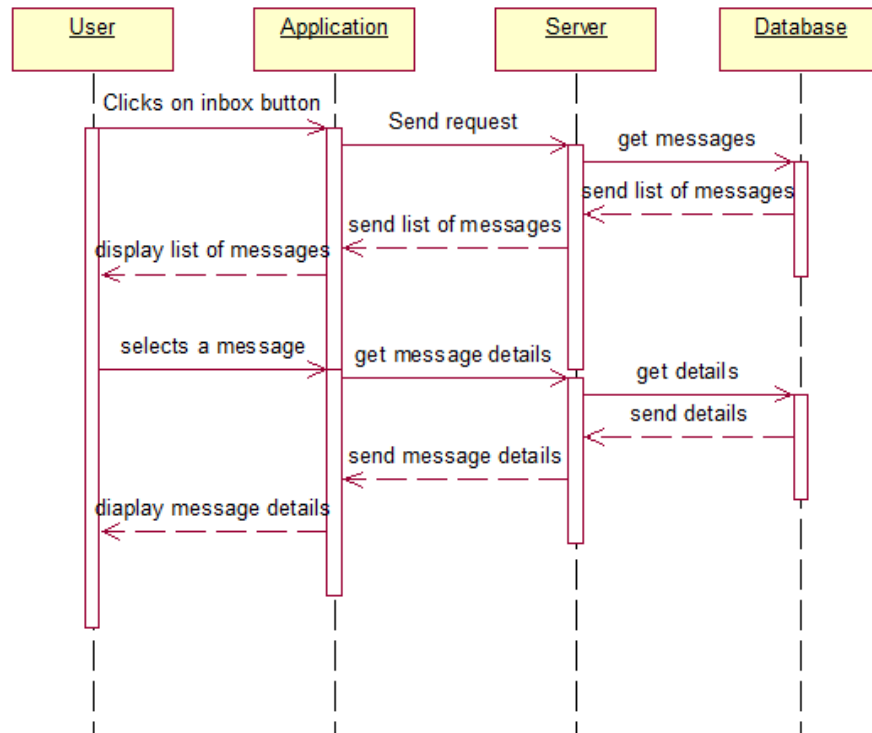


SD - 16 Like posts

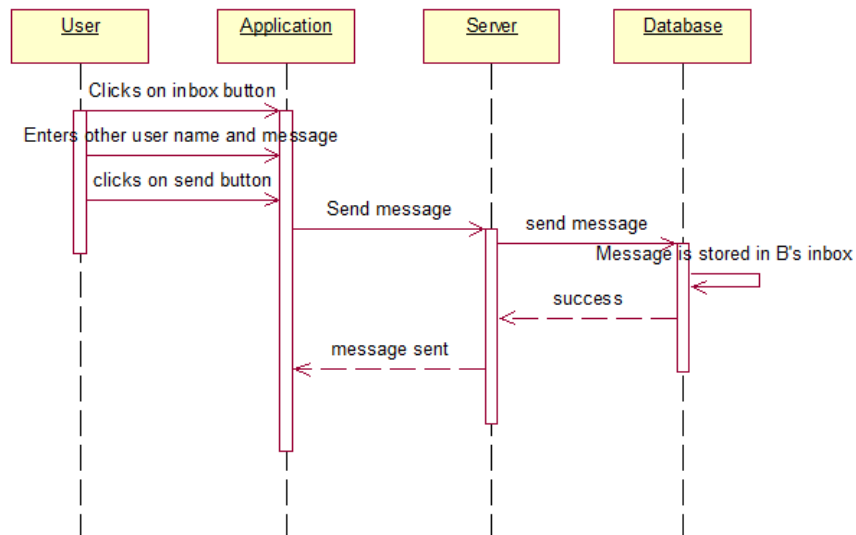


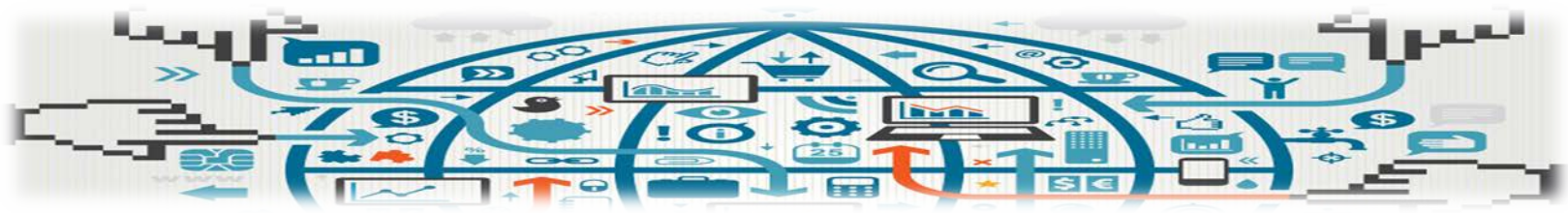


SD - 17 View inbox

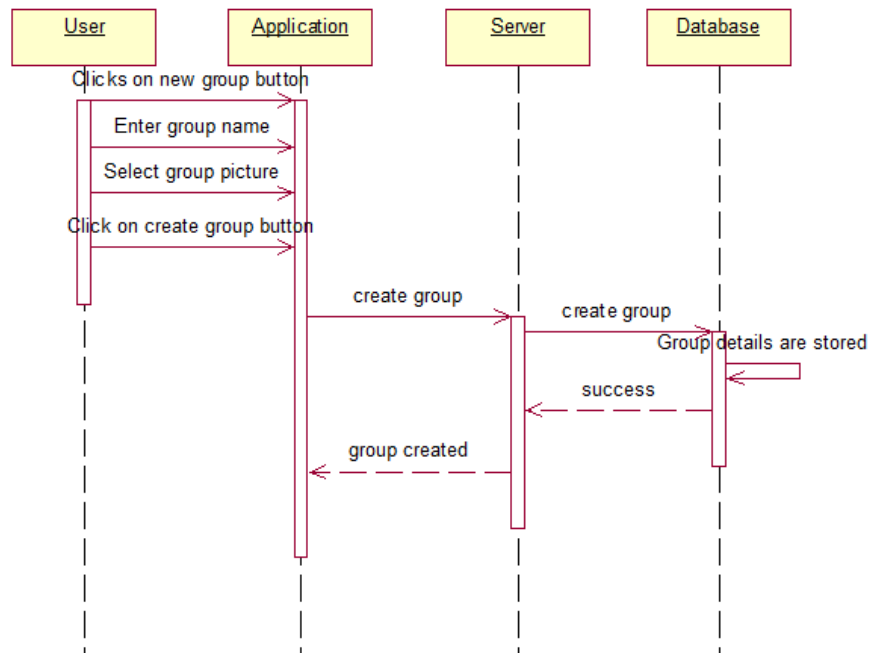


SD - 18 Send messages from inbox

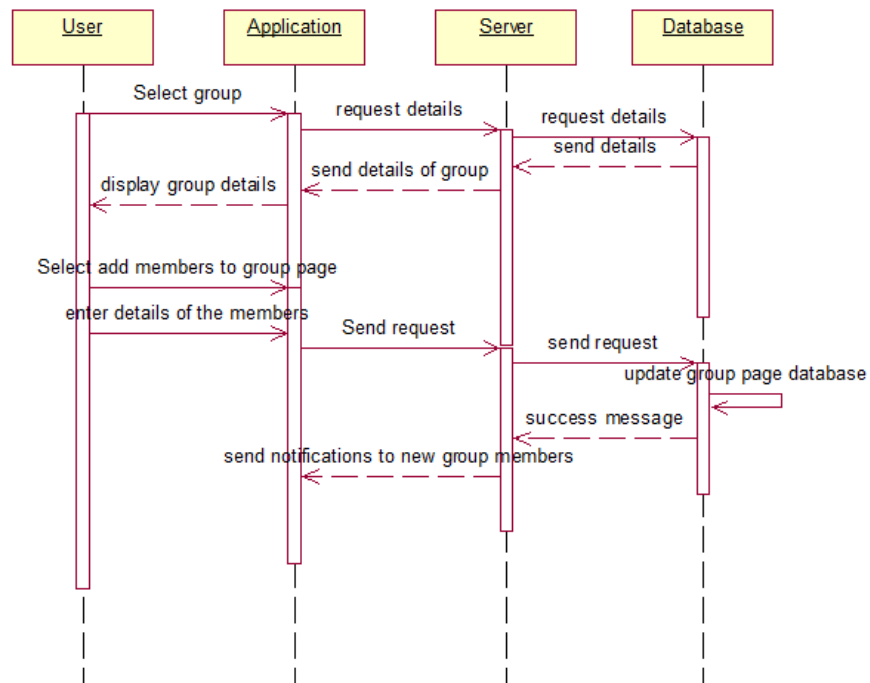




SD - 19 Create group page

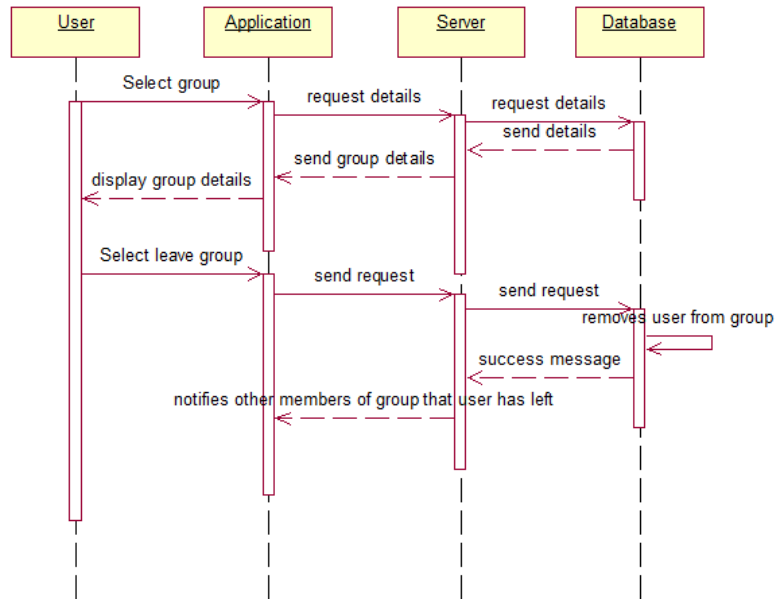


SD - 20 Add friends to group page

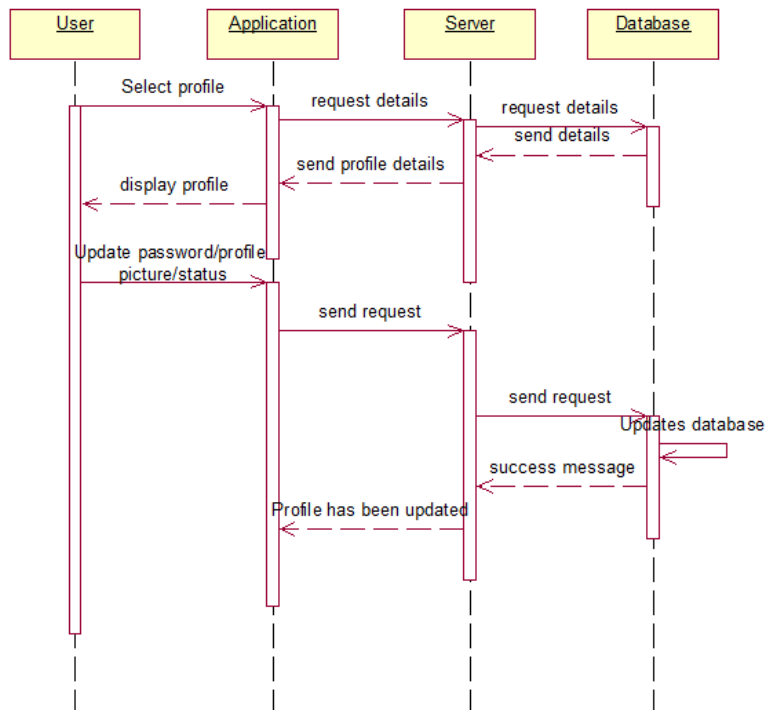




SD - 21 Leave group page



SD - 22 Change profile settings





3.6 User Interface

- This is the login page. The user has to register for the first time. He/she can login using the username/password from the second time.

The screenshot shows a web browser window with the address bar displaying "https://xyz.com". The page content is divided into two main sections: "REGISTER" and "LOGIN".

REGISTER Form:

- First Name:
- Last Name:
- Email Id:
- Password:
- Date of Birth:
- Add Profile Picture:
- Register:

LOGIN Form:

- Email Id:
- Password:
- Submit:
- Forgot Password:

- The user can also retrieve his/her forgotten password by answering the security question, i.e. by providing his/her e-mail id.

The screenshot shows a web browser window with the address bar displaying "https://xyz.com". The page content includes a "HOME PAGE" button and a "FORGOT PASSWORD" section.

FORGOT PASSWORD Form:

- Security Question:
- Enter your Email ID:
- RESET:
- SUBMIT:



https://xyz.com

HOMEPAGE

FORGOT PASSWORD

Enter New Password:

Confirm Password:

RESET SUBMIT

- This is the UI of our application.

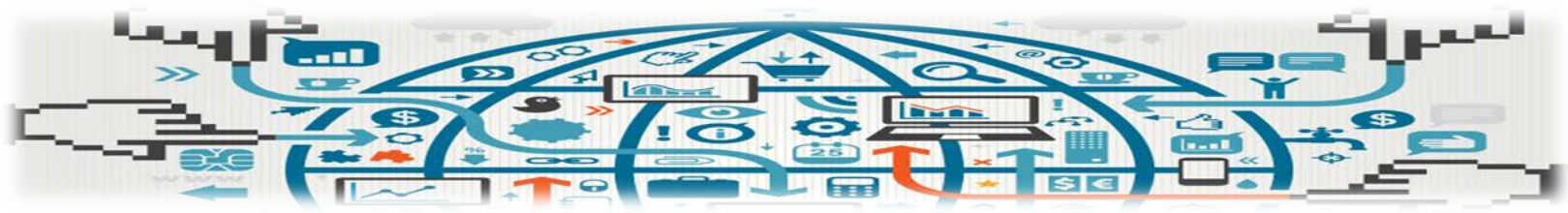
https://xyz.com

Profile Pic

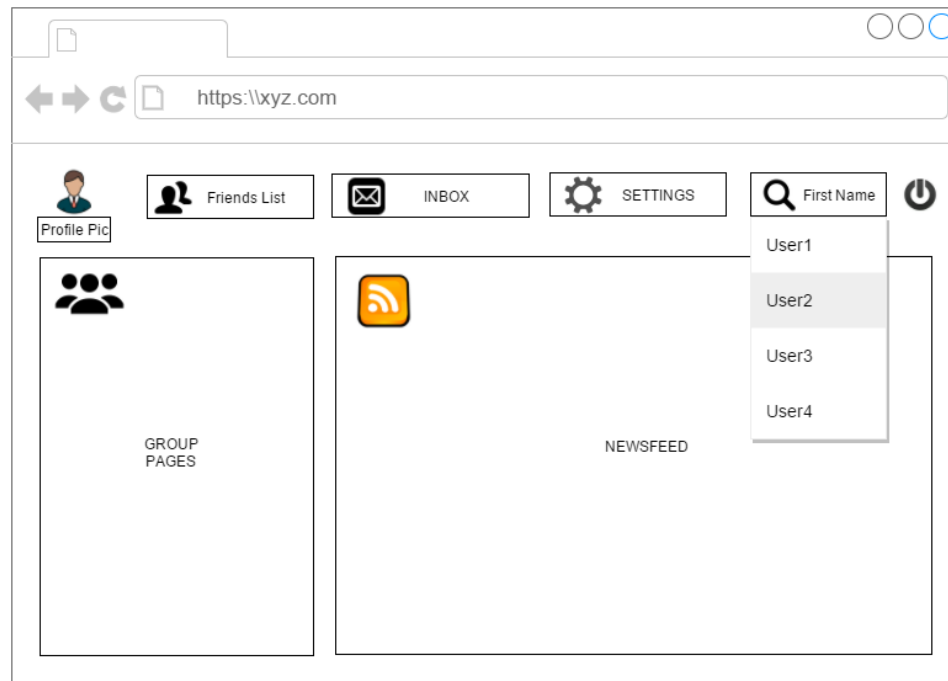
Friends List INBOX SETTINGS SEARCH

GROUP PAGES

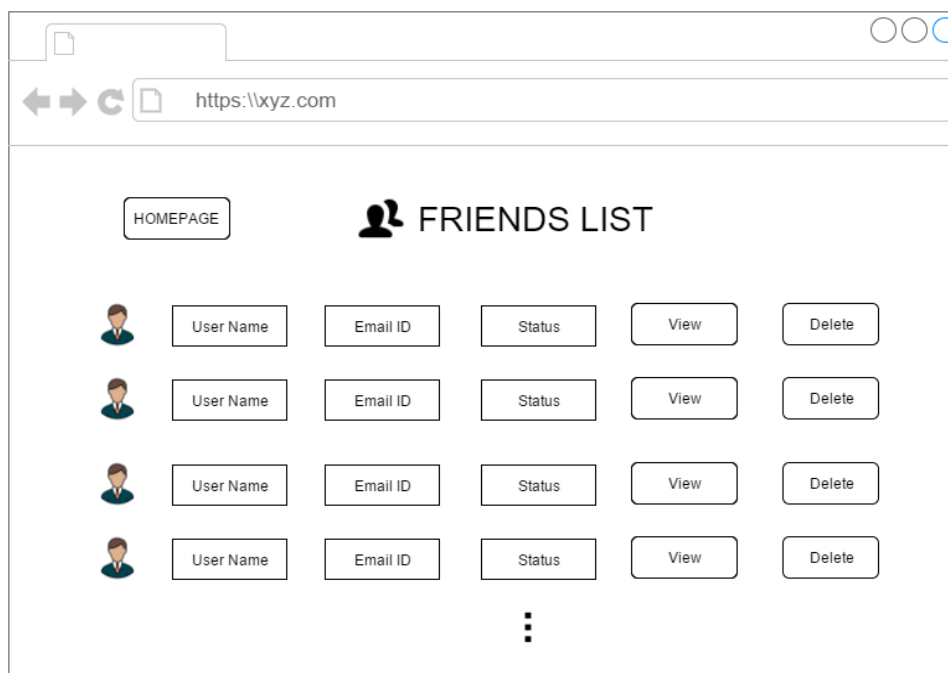
NEWSFEED

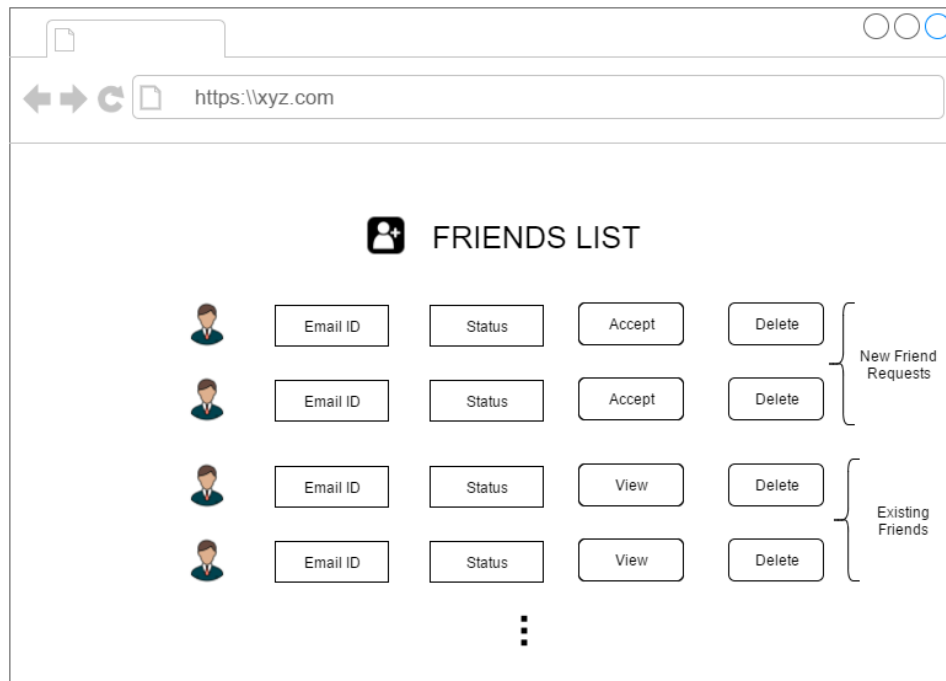
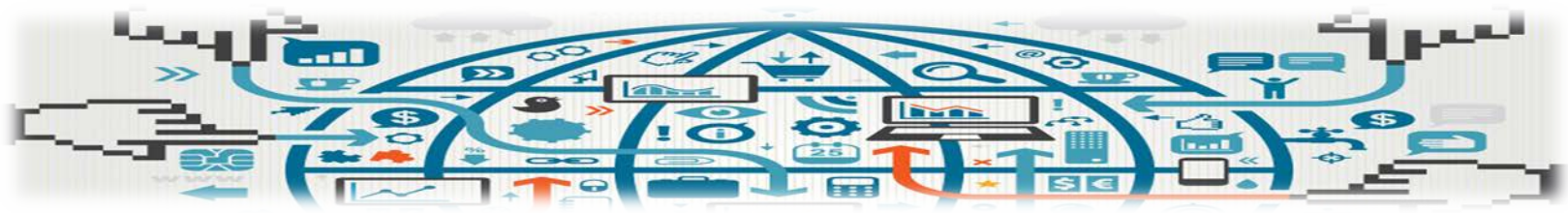


- The user can search for his friends using their first/last names.

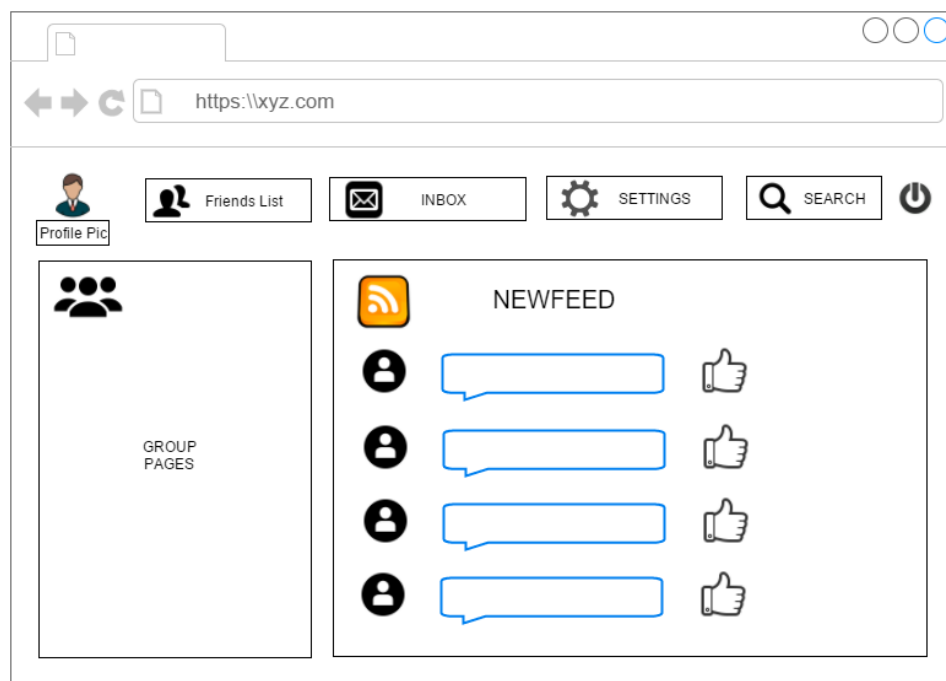


- The user can view his friends under the “friends list”. He/she can also accept/delete requests, view/delete friends.



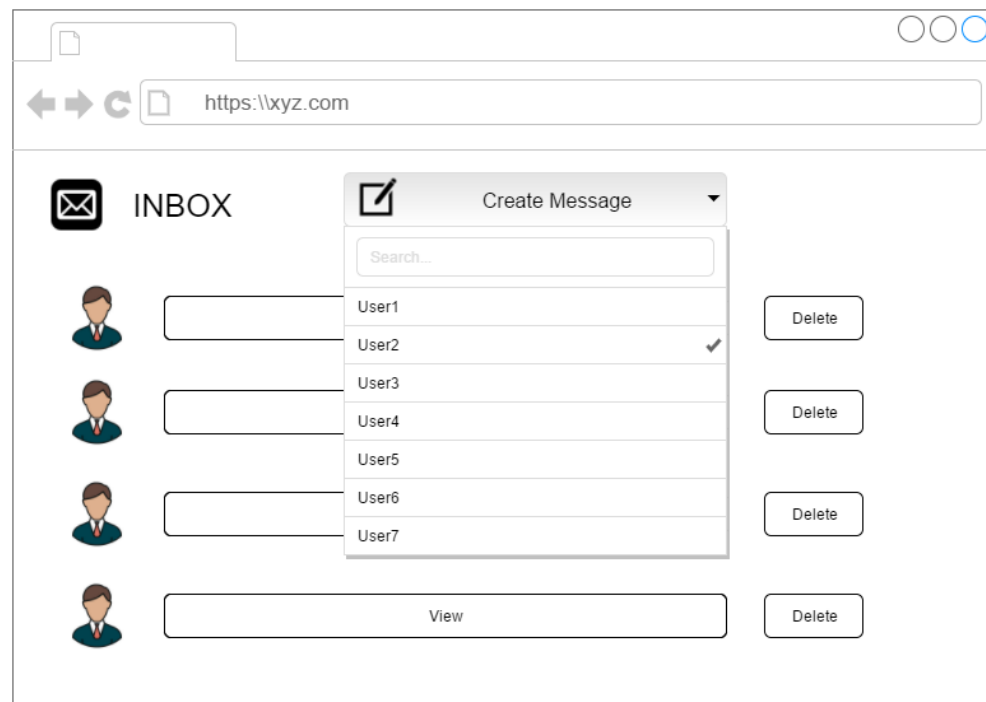
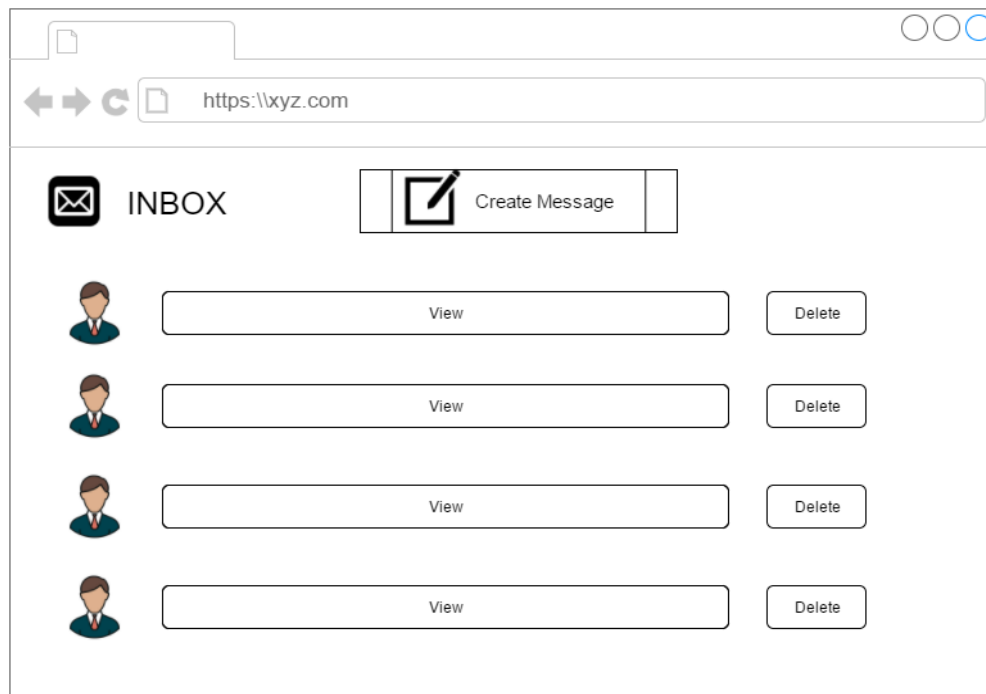


- The user can post in the newsfeed. It will be viewable by all his friends. The user can also like any post.





- The user can use the Inbox feature to view messages received. He/she can also compose messages using the create message feature.





https://xyz.com

INBOX Create Message: USER2

Type the message here...

SEND

- The user can create a group, add friends to the group as well as delete a group (if admin). He/she can also post in the group which can be viewed by only the group members.

https://xyz.com

Profile Pic Friends List INBOX SETTINGS Search

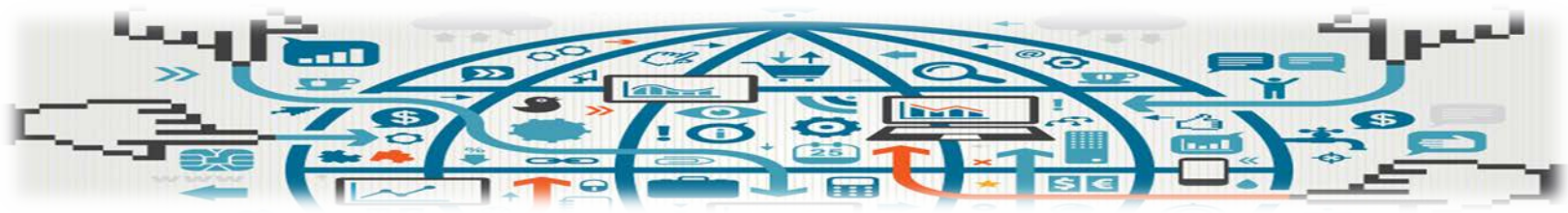
Create New Group

GROUP 1

GROUP 2

GROUP 3

NEWSFEED



Browser window showing the homepage of a social networking application. The address bar displays `https://xyz.com`.

Navigation buttons: **HOMEPAGE** and **Create New Group**.

Form fields for creating a group:

- Group Name** (text input)
- Add users** (button with a person icon and a plus sign)
- Add group picture** (button with a group of people icon)

Browser window showing the group page of the social networking application. The address bar displays `https://xyz.com`.

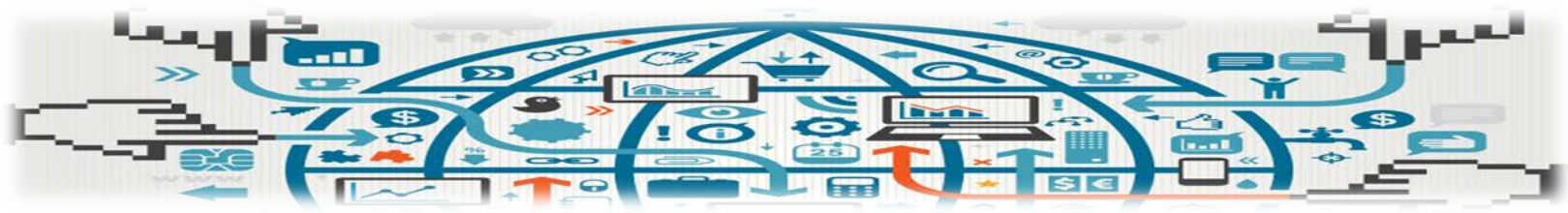
Navigation buttons: **HOMEPAGE** and **Exit group page XYZ**.

Group header information:

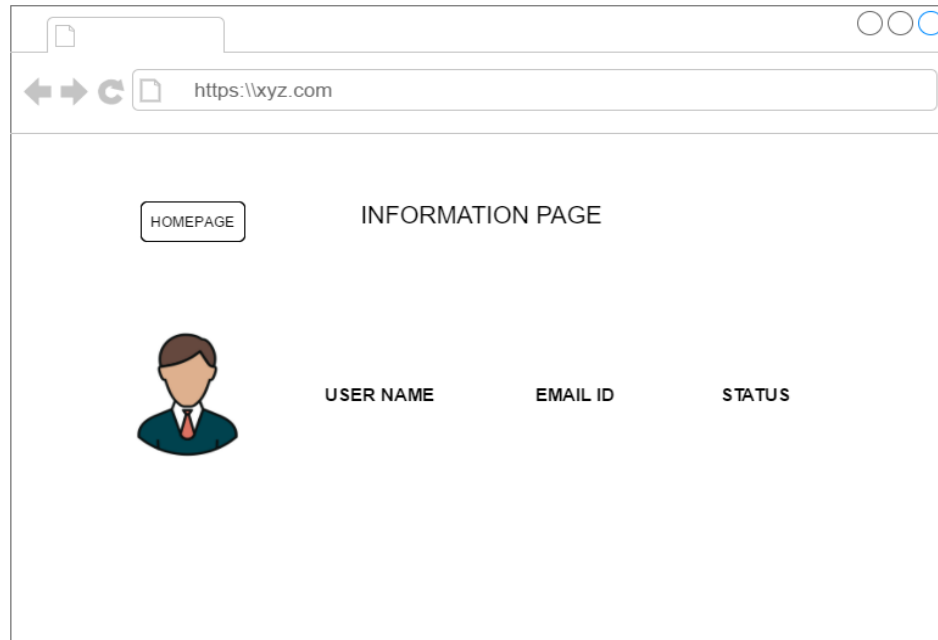
- Group icon (group of people)
- GROUP XYZ** (group name)
- Add users** (button with a person icon and a plus sign)

Group members list (three entries shown):

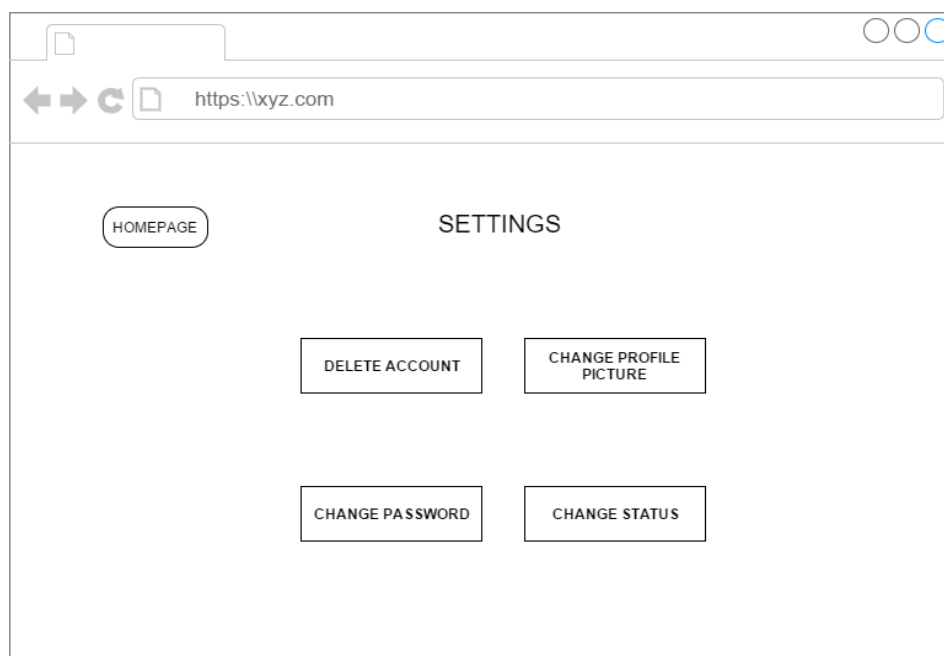
- Member 1: [Name input field]
- Member 2: [Name input field]
- Member 3: [Name input field]



- The information page is used to display the profile picture, username, email-id and status of the user.



- The user can change his/her profile picture, password/status or delete account using the settings feature.





3.7 Error Handling

- When the user accidentally clicks on the Back/Home screen button while typing a post or a message, a prompt should be displayed asking the user if he/she wants to leave the current page.
- A log enabled application will keep track of errors leading to application failure. These logs will help retrieving the data lost due to the application failure.
- During Registration there should be validation checks on the username and password. Passwords (string) should containing at least 1 uppercase, 1 lowercase and 1 number with a predefined minimum (9) and maximum value. There should also be a check for a null string.
- There should be a predefined time limit for fetching data from the database. If it takes longer than the predefined time, then a timeout message should be displayed or there can be an attempt to contact the database again (maximum or 3 re-attempts).
- When the user enters invalid credentials, the application should display a message “Email and password does not match”.
- The number of likes for a post cannot be a negative value.

3.8 REFERENCES:

- <https://www.draw.io/>