

CHAT BUDDY

UCS503 Software Engineering Project Report

End-Semester Evaluation

Submitted by:

(102003177) Sanchita Bora

(102003183) Anshika

(102003188) Medhansh Singh Verma

BE Third Year, CoE8

Group No: 1

Submitted to:

Dr. Sumit Kumar



Computer Science and Engineering Department

TIET, Patiala

December 2022

TABLE OF CONTENTS

S.No.	Assignment
1.	Project Selection Phase
1.1	Software Bid
1.2	Project Overview
2.	Analysis Phase
2.1	Use Cases
2.1.1	Use-Case Diagram
2.1.2	Use Case Templates
2.2	Swimlane Diagram
2.3	Data Flow Diagrams (DFDs)
2.3.1	DFD Level 0
2.3.2	DFD Level 1
2.3.3	DFD Level 2
2.4	Software Requirement Specification in IEEE Format
2.5	User Story Cards
3.	Design Phase
3.1	Class Diagram and Object Diagram
3.2	Sequence Diagram
3.3	Collaboration Diagram
3.4	Database Design : ER Diagram
3.5	State Chart Diagram

4. Implementation

4.1 Component Diagrams

4.2 Deployment Diagrams

4.3 Screenshots of Working Project

5. Testing

5.1 Cyclomatic Complexity (All modules)

5.2 Test Cases

5.3 Test Reports

Software Bid

UCS 503- Software Engineering Lab

Group : 3COE8

Dated: 2/8/2022

Team Name: Mind Benders

Team ID (will be assigned by Instructor):

Please enter the names of your Preferred Team Members. :

- You are required to form a three to four person teams'
- Choose your team members wisely. You will not be allowed to change teams.

Name	Roll No	Project Experience	Programming Language used	Signature
Sanchita Bora	102003177	Plasma Life: An app to find plasma donors	Java	
Anshika	102003183	Handwritten Digit Recognition	Python	
Medhansh Singh Verma	102003188	Handwritten Digit Recognition	Python	
Devaang Goswami	102003197	AI chat bot	Python	

Programming Language / Environment Experience

List the languages you are most comfortable developing in, as a team, in your order of preference. Many of the projects involve Java or C/C++ programming.

1. C/C++
2. Python
3. Java
4. PL/SQL

Choices of Projects:

Please select 4 projects your team would like to work on, by order of preference: *[Write at-least one paragraph for each choice (motivation, reason for choice, feasibility analysis, etc.)]*

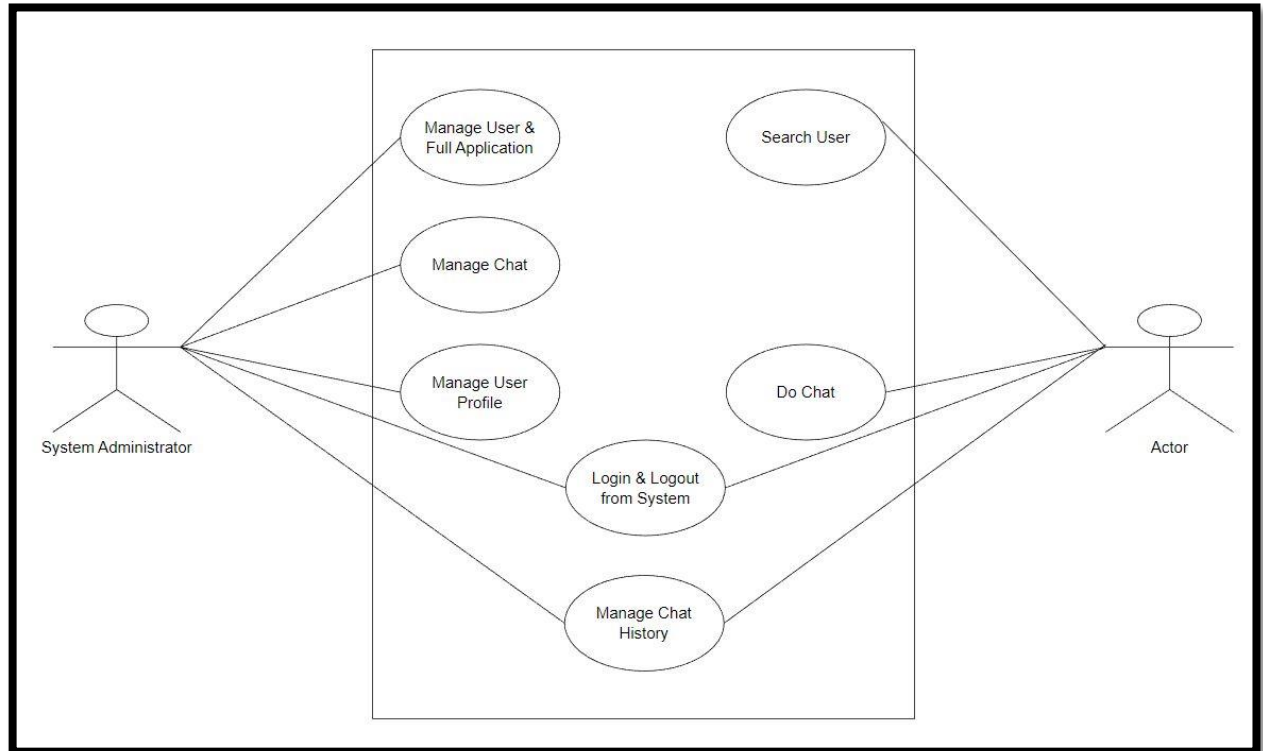
First Choice: Messenger with Encryption	A messenger application which has different security levels selectable by user hosted locally on a private VPN server (peer to peer).
Second Choice: Fingerprint based Voting Application	It can work by feeding each voter's fingerprint and identity proof. The voters will be allowed to login using their IP and password and vote using their fingerprint.
Third Choice: Local Train Booking Application	The passengers will have to fill out a booking form stating their destination and suitable timings.
Fourth Choice: Car Rental System	Login to get access, select the car model you want to rent and choose the duration. You can also check cars available. It also provides the specification of the car chosen.

Project Overview

The objective of the project is to develop an instant encrypted messaging application that will help people to have secure and comfortable messaging experience. The project will fulfil below mentioned properties:

- **Communication:** To establish seamless communication between two people across the globe.
- **User Friendly:** The project will be beginner friendly.
- **Secure:** It will establish secure communication between people.

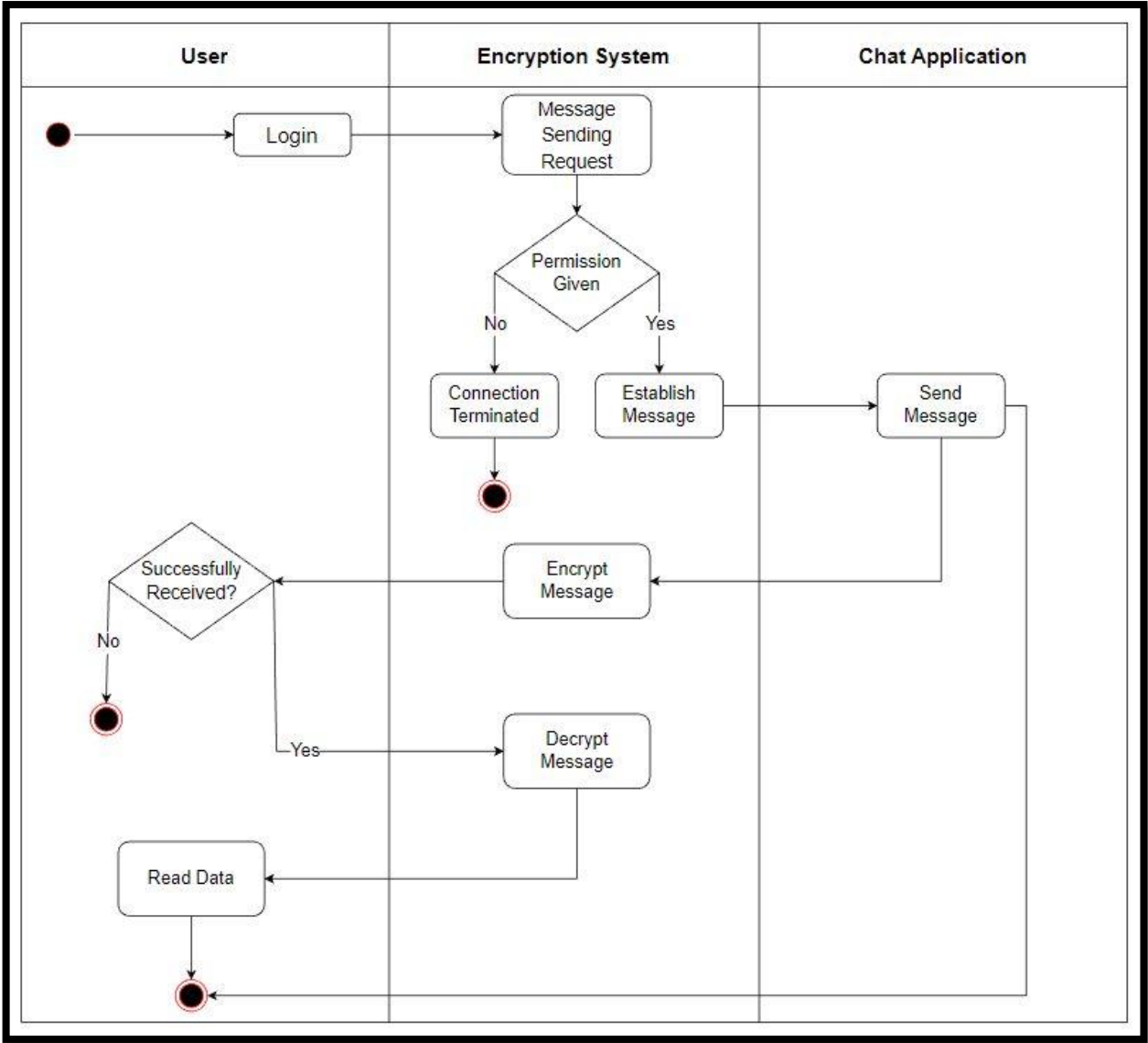
Use Case Diagram



Use Case Template

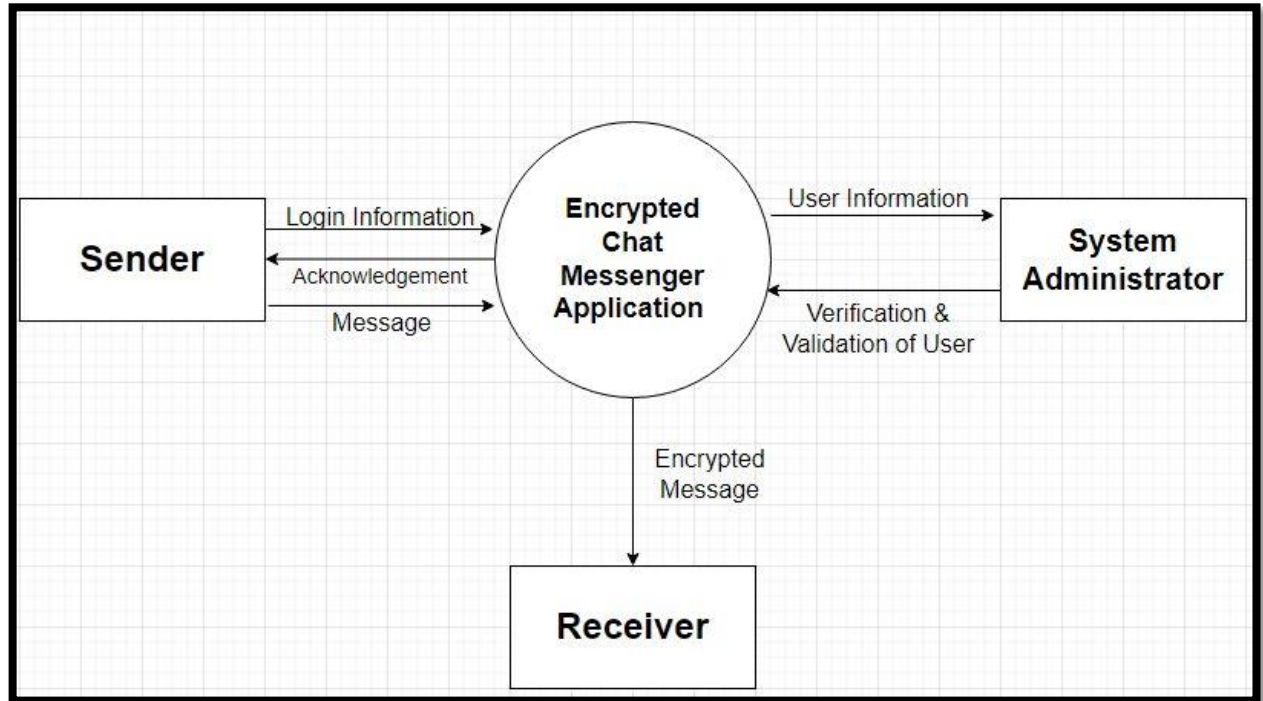
Use Case Title	Select a user to start chat
Abbreviated Title	Select
Use Case ID	3
Actors	User
Description: With this facility user can search for another user from list of users to start chat.	
Pre-Conditions: User must be logged in	
Task Sequence: <ol style="list-style-type: none">1. Search screen will be show by system2. Select search criteria and enter the name of required user3. On clicking the searching button, system will show search results	
Post Conditions: <ol style="list-style-type: none">1. User can view his desired results2. User can go for another search	
Modification History: Oct 2022	
Author: Group 1	

Swimlane Diagram

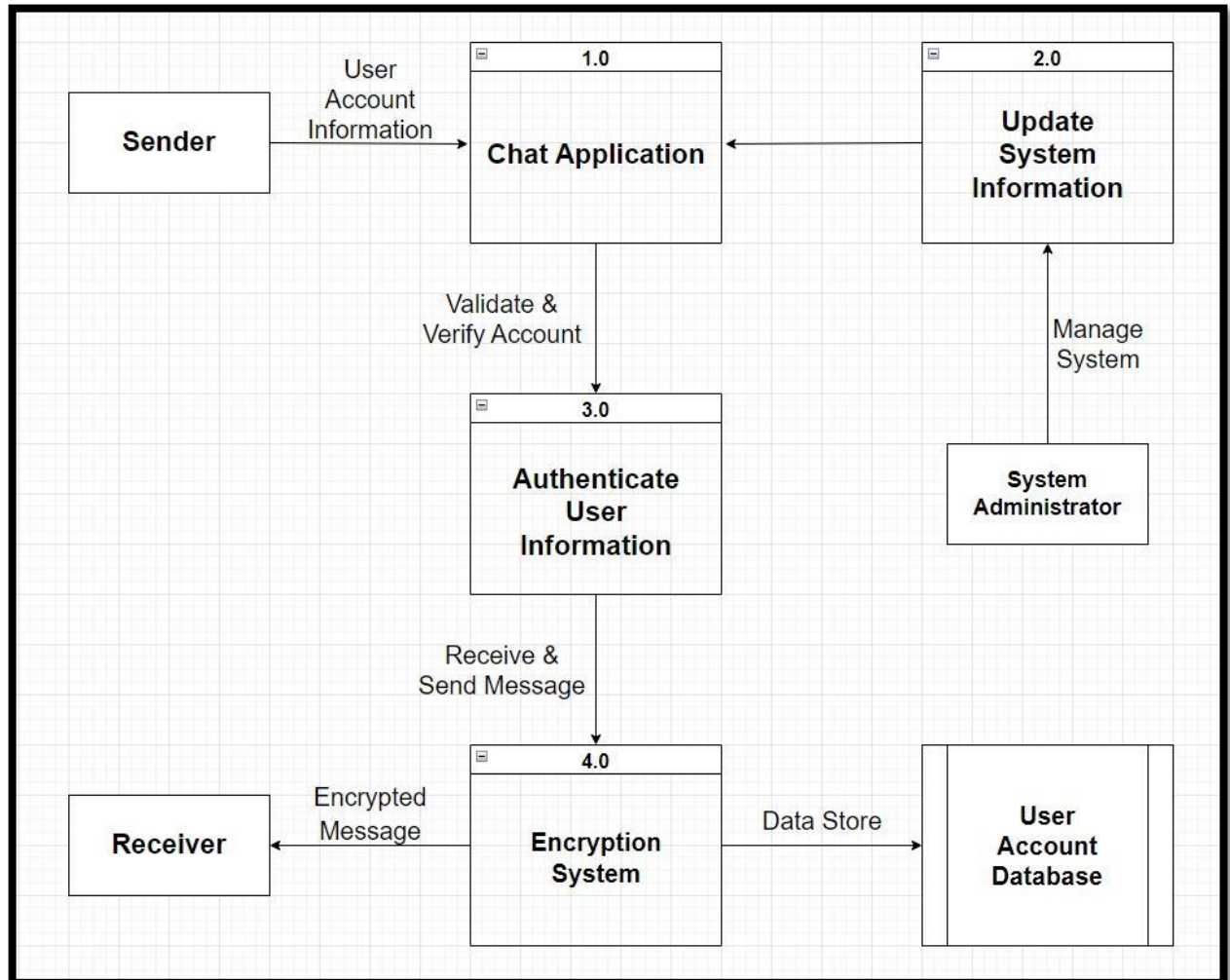


Data Flow Diagrams

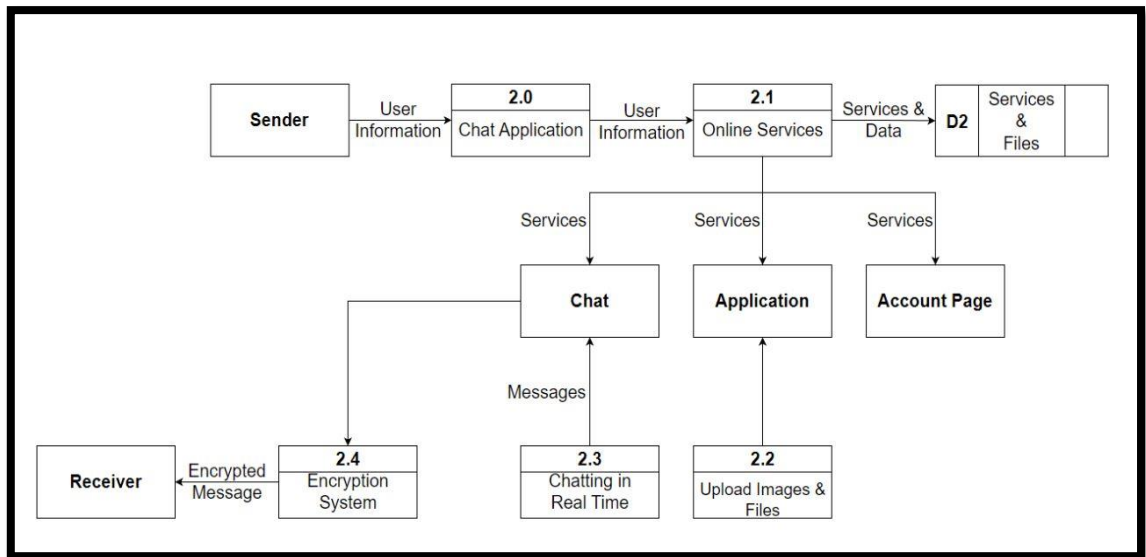
DFD Level 0



DFD Level 1



DFD Level 2



SOFTWARE REQUIREMENTS SPECIFICATION

for

Chat Buddy

Version 1.0

Prepared by: 1. Sanchita Bora (102003177)

2. Anshika (102003183)

3. Medhansh Singh Verma (102003188)

Submitted to: Dr. Sumit Kumar

Contents

1 Introduction

- 1.1 Purpose
- 1.2 Scope of the Development Project
- 1.3 References
- 1.4 Overview

2 Overall Description

- 2.1 Product Perspective
- 2.2 Product Functions
- 2.3 User Characteristics
- 2.4 General Constraints, Assumptions and Dependencies
- 2.5 Operating Environment
- 2.6 Design and Implementation Constraints

3 External Interface Requirements

- 3.1 User Interfaces
- 3.2 Hardware Interfaces
- 3.3 Software Interfaces
- 3.4 Communications Interfaces

4 Nonfunctional Requirements

- 4.1 Availability
- 4.2 Scalability
- 4.3 User Friendly Interface
- 4.4 Extensibility

5 Feasibility Report

- 5.1 Schedule Feasibility
- 5.2 Technical Feasibility
- 5.3 Economical Feasibility
- 5.4 Operational Feasibility
- 5.5 Legal Feasibility
- 5.6 Cultural/Behavioural Feasibility

6 Work Breakdown Structure

7 Gantt Chart

1 Introduction

1.1 Purpose

The purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the product and its functionality.

1.2 Scope of the Development Project

- The Encrypted Messaging Application will be a text communication application that will establish communication between two computers using point to point connection.
- The application will not support communication. We are working on developing better technologies to overcome the limitations of the application.
- Companies can communicate instantly within their organization.
- The encryption provided to the messages makes it very secure from outside attacks.

1.3 References

- <https://www.codingnepalweb.com/chat-web-application-using-php/>
- <https://stackoverflow.com/questions/3422759/php-aes-encrypt-decrypt>
- <https://www.slideshare.net/atulrockx/srs-of-3>

1.4 Overview

The remaining sections of this document provide a general description, including characteristics of the users of this project, the product's hardware, and the functional and data requirements of the product. General description of the project is discussed in section 2 of this document. Section 2 gives the functional requirements, data requirements and constraints and assumptions made while designing the multi-utility system. It also gives the user viewpoint of product use. Section 3 gives the specific requirements of the product. Section 3.0 also discusses the external interface requirements and gives detailed description of functional requirements.

2 Overall Description

2.1 Product Perspective

- This software is a user-friendly real time chat application for sharing encrypted messages from one user to another.
- There is a two way communication between different users.
- It allows users to find other logged in users.

2.2 Product Functions

The product should be able to perform the following operations:

- **User Authentication:** The application will keep into account the authentication and reliability of the user using the application.
- **End to End Encrypted communication:** The application will use a reliable encryption method to secure the message from foreign attacks.

2.3 User Characteristics

The goal is to design a user friendly application for all kinds of users. Thus while designing the software one can assume that each user is computer literate.

2.4 General Constraints, Assumptions and Dependencies

Constraints:-

- The application does not by any means open the web browser. If user wishes to open the web browser he must open it externally.
- The system need to be permanent connected with internet.

Assumptions and Dependencies:-

- There should be LAN or internet connection.
- Users should know each other.
- There can be multiple clients.

2.5 Operating Environment

The website will be operate in any Operating Environment - Mac, Windows, Linux, iOS 9+, Android 4.4+ etc.

Using browsers such as - Chrome 45+, Firefox 38+, Opera 30+, Internet Explorer 10+, Edge 12+, Safari 9+ etc.

2.6 Design and Implementation Constraints

The front end design shall be made using HTML. For implementation of the software, we shall use Javascript for front-end while using PHP for developing the back-end and SQL as our database.

3 External Interface Requirements

3.1 User Interfaces

This application interacts with the user through G.U.I. The interface is simple , easy to handle and self-explanatory.

Once opened, user will easily come into the flow with the application and easily uses all interfaces properly.

The User interface shall comprise of the following: (Refer Figure 3.1, 3.2)

- Fonts Used: Poppins, Sans-sarif
- Colors Used:
 - Background : #f7f7f7, #fff
 - Headings, Subheadings and Paragraph : #000, #333

3.2 Hardware Interfaces

The following list presents the Hardware interface requirements:

- The product requires very limited graphics usage.
- The interaction requires just a simple keyboard for taking the user input.
- A simple mouse/keyboard can be used to navigate the website.
- The product does not require usage of speaker/microphone, web-cam or animation and graphics.

3.3 Software Interfaces

The Software Interfaces used are:

- MySQL Workbench for Database.
- PHP for Backend.
- HTML for Frontend.

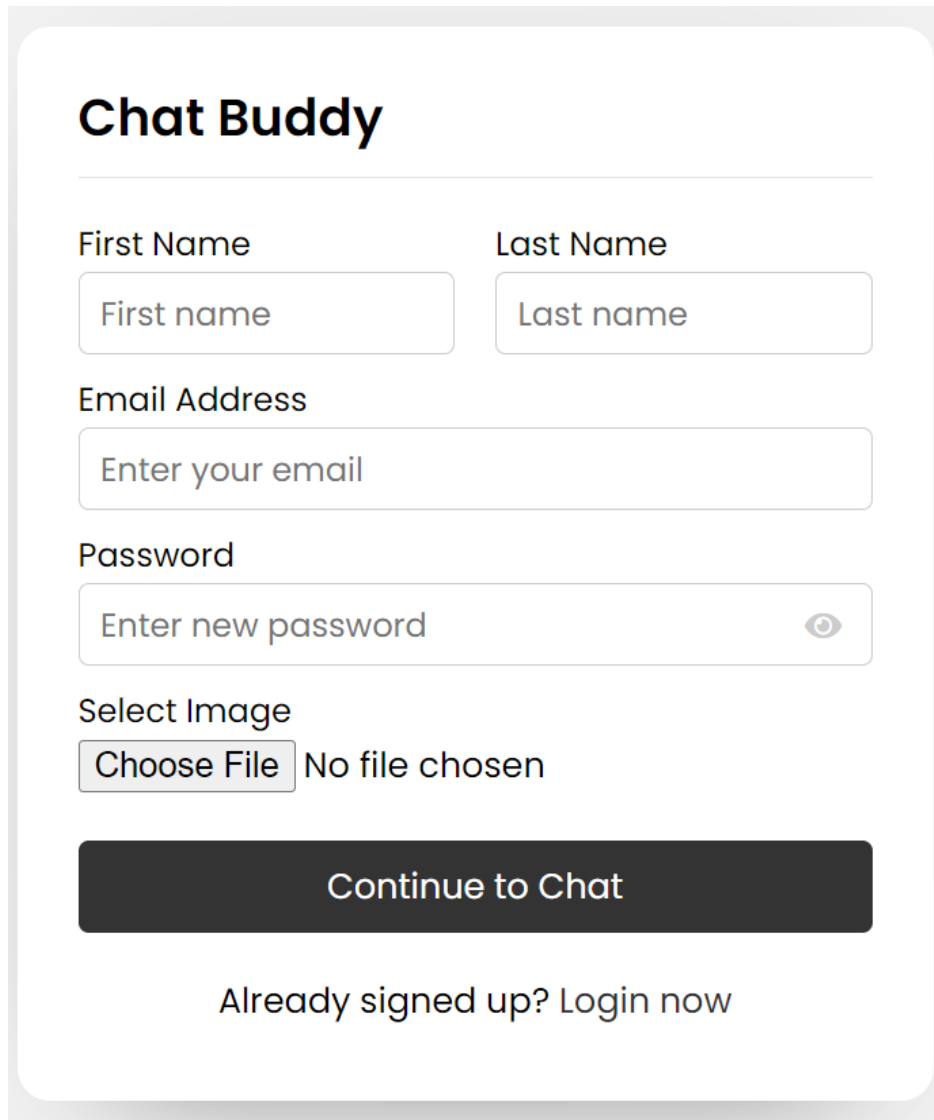
The requirements for usage of the application are:

- Any Operating Environment - Mac, Windows, Linux, iOS 9+, Android 4.4+ etc.
- Supporting browsers such as - Chrome 45+, Firefox 38+, Opera 30+, Internet Explorer 10+, Edge 12+, Safari 9+ etc.

3.4 Communications Interfaces

The requirements associated with communications functions required by this product are:

- Network server communications protocol used shall be HTTP/HTTPS using port 8080/4433 on the server.

A registration form titled "Chat Buddy" with a light gray background and rounded corners. The form contains several input fields: "First Name" and "Last Name" (each with a placeholder "First name" and "Last name" respectively), "Email Address" (with placeholder "Enter your email"), and "Password" (with placeholder "Enter new password" and a toggle icon). Below the password field is a "Select Image" section with a "Choose File" button and the text "No file chosen". At the bottom is a large dark gray button labeled "Continue to Chat" and a link "Already signed up? Login now".

Chat Buddy


First Name Last Name

First name Last name

Email Address

Enter your email

Password

Enter new password 

Select Image

Choose File No file chosen

Continue to Chat

Already signed up? [Login now](#)

Figure 3.1

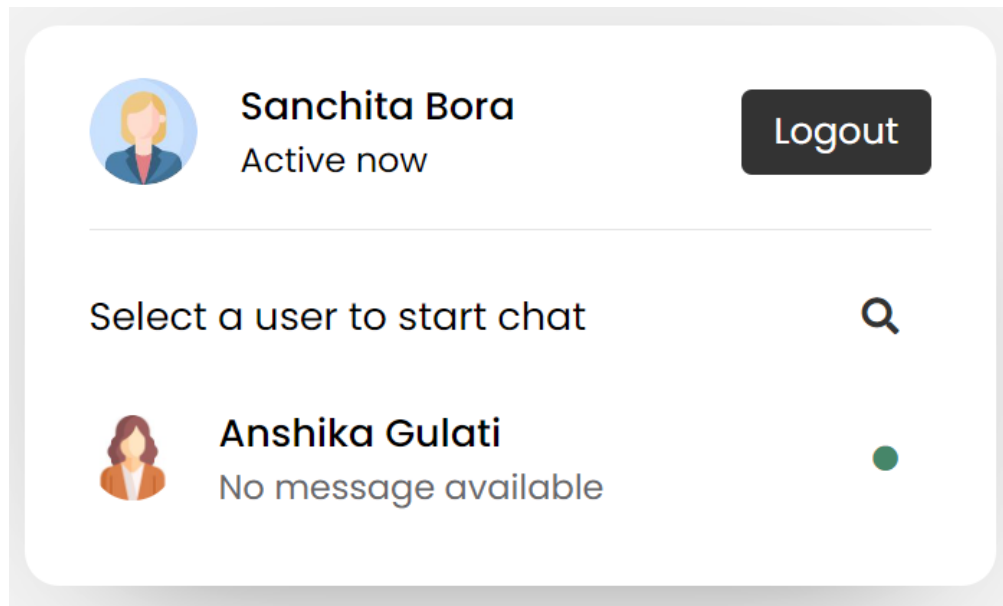


Figure 3.2

4 Nonfunctional Requirements

4.1 Availability

The application will provide 24/7 service.

4.2 Scalability

The application will be compatible in various browsers and system. The expansion should not affect performance of the system.

4.3 User Friendly Interface

The interface will be easy to use and beginner friendly.

4.4 Extensibility

Any further expansion of the application will be depended on the improvement in performance by development of new technology.

5 Feasibility Report

5.1 Schedule Feasibility

Estimated period of building the project: 4 Months

The development of the project contains following phases:

- Planning: 2 Weeks
- Designing: 3 Weeks
- Implementation: 1 Month
- Testing: 2 Weeks
- Maintenance: 3 Weeks

5.2 Technical Feasibility

- The frontend will be built using HTML and implemented using Javascript.
- The backend of the application will be developed using PHP and SQL as the database.

5.3 Economical Feasibility

The making of the project is financially feasible and its maintenance cost is also very low. All the softwares that we require during the implementation of the project are open-source and the hosting and database services will be availed at little to no prices.

5.4 Operational Feasibility

The project will be fully operational and it can be operated across various platforms. The application will be developed with a view to provide the user a secure chatting experience.

5.5 Legal Feasibility

We are ensuring that we are not using any pirated stuff or any copyrighted stuff. If any resource is used while making the project, it will be cited in the documentation of the project in the references section.

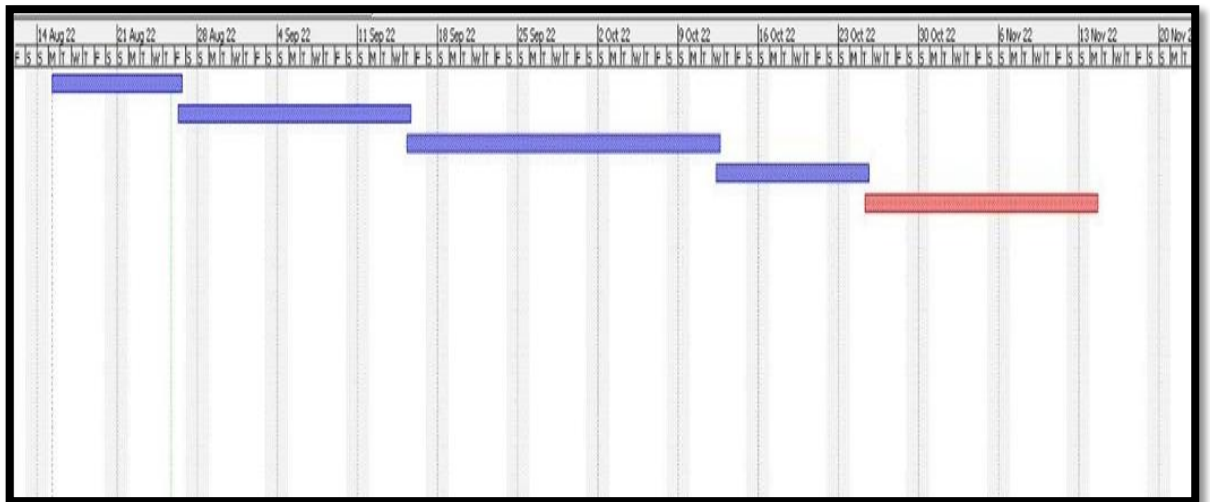
5.6 Cultural/Behavioural Feasibility

The application will ensure connectivity between the people across an organization with utmost priority given to the security and privacy of messages being sent. The project is unbiased and will be equally accessible to all citizens within the provided territory within which the application will be accessible.

6 Work Breakdown Structure

Sr. No.	Task	Duration	Start Date	End Date
1	Planning	11 Days	15-08-2022	26-08-2022
1.1	Requirement	5 Days		
1.2	Resource Plan	3 Days		
1.3	Risk Plan	3 Days		
2	Designing	20 Days	26-08-2022	15-09-2022
2.1	System Design	5 Days		
2.2	Database Design	5 Days		
2.3	Program Design	10 Days		
3	Implementation	27 Days	15-09-2022	12-10-2022
3.1	Perform Coding	22 Days		
3.2	Build Deliverable	3 Days		
3.3	Time Management	2 Days		
4	Testing	13 Days	12-10-2022	25-10-2022
4.1	Develop Test Cases	3 Days		
4.2	Implement Test Cases	10 Days		
5	Maintenance	20 Days	25-10-2022	14-11-2022
5.1	Project Closure	10 Days		
5.2	Review	10 Days		

7 Gantt Chart



User Story Cards

#0001	User Login
As a [registered user], I want to [login], so I can access the [content of the application]	

Show messages here if not successful

Realtime Chat App

Email Address

Enter your email

Password

Enter your password

Continue to Chat

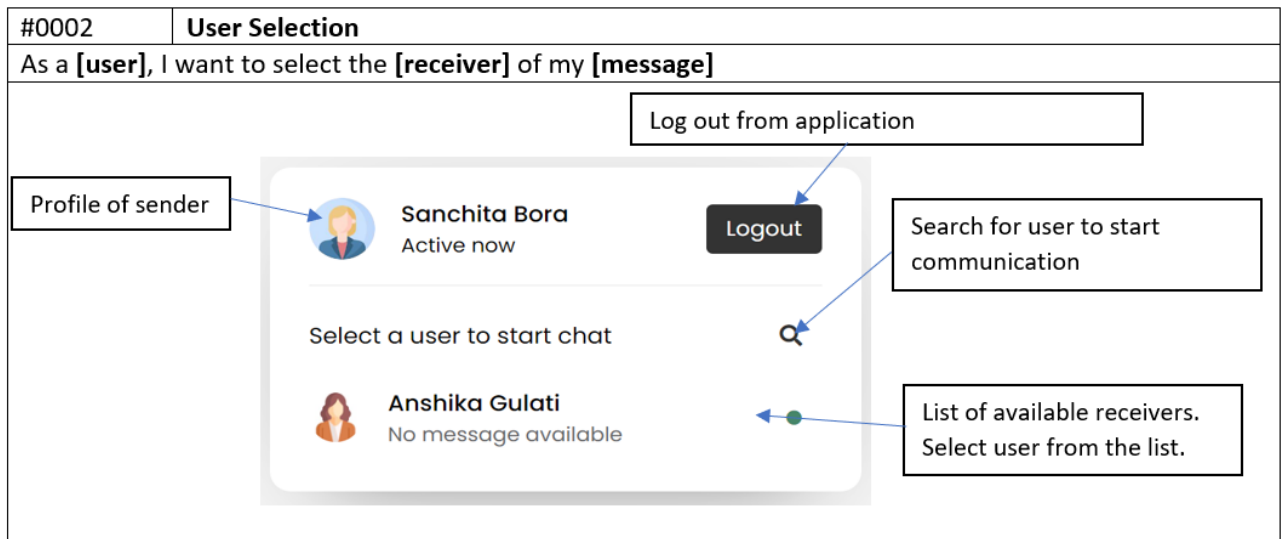
Not yet signed up? Signup now

User's email address. Validate format.

Authenticate login

Confirmation:

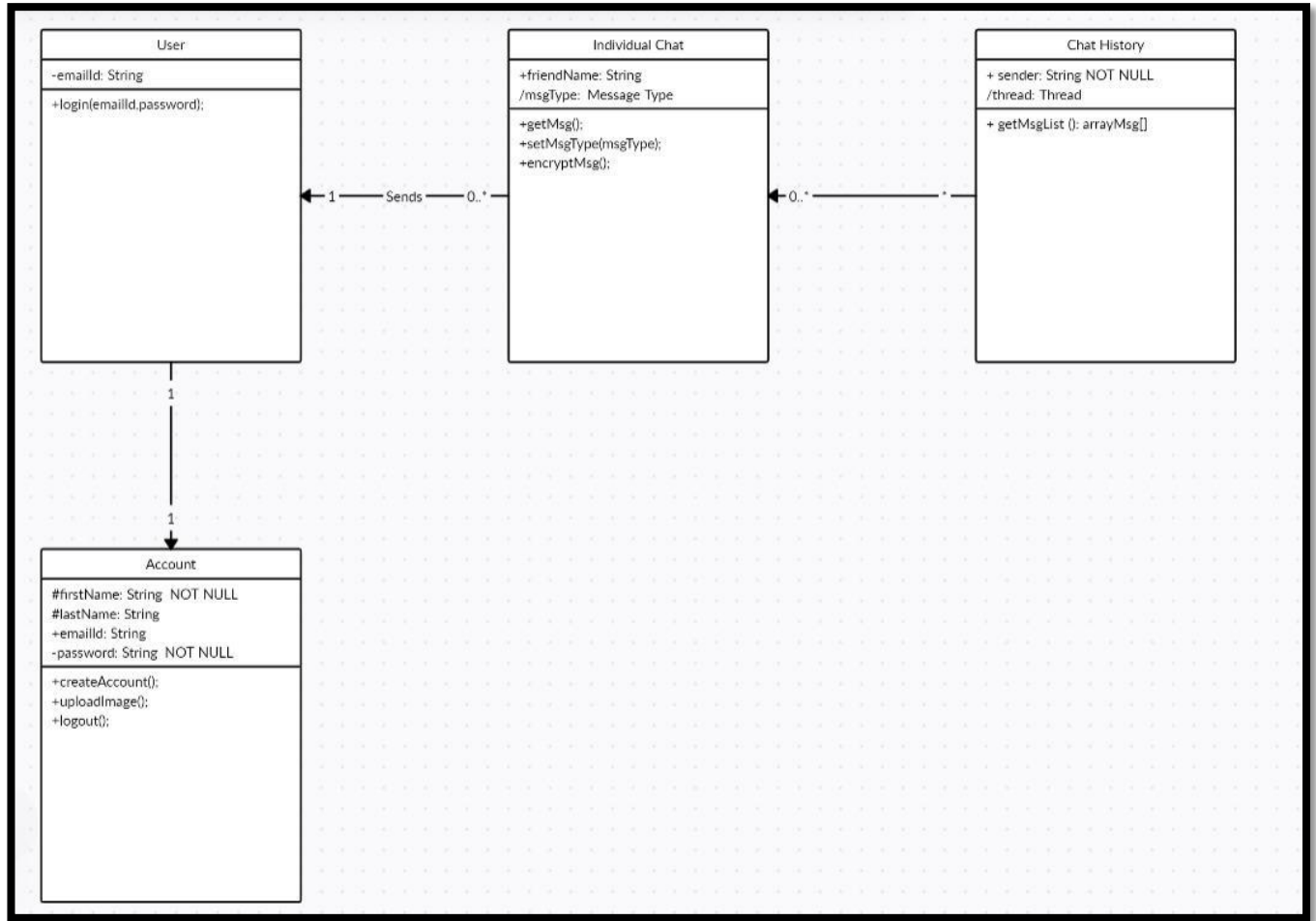
1. Success: Valid user logged in and referred to chatting window
2. Failure: Display message
 - "Not a valid email!"
 - "This email does not Exist!"
 - "Email or Password is Incorrect"
 - "All input fields are required!"
 - "Something went wrong. Please try again!"



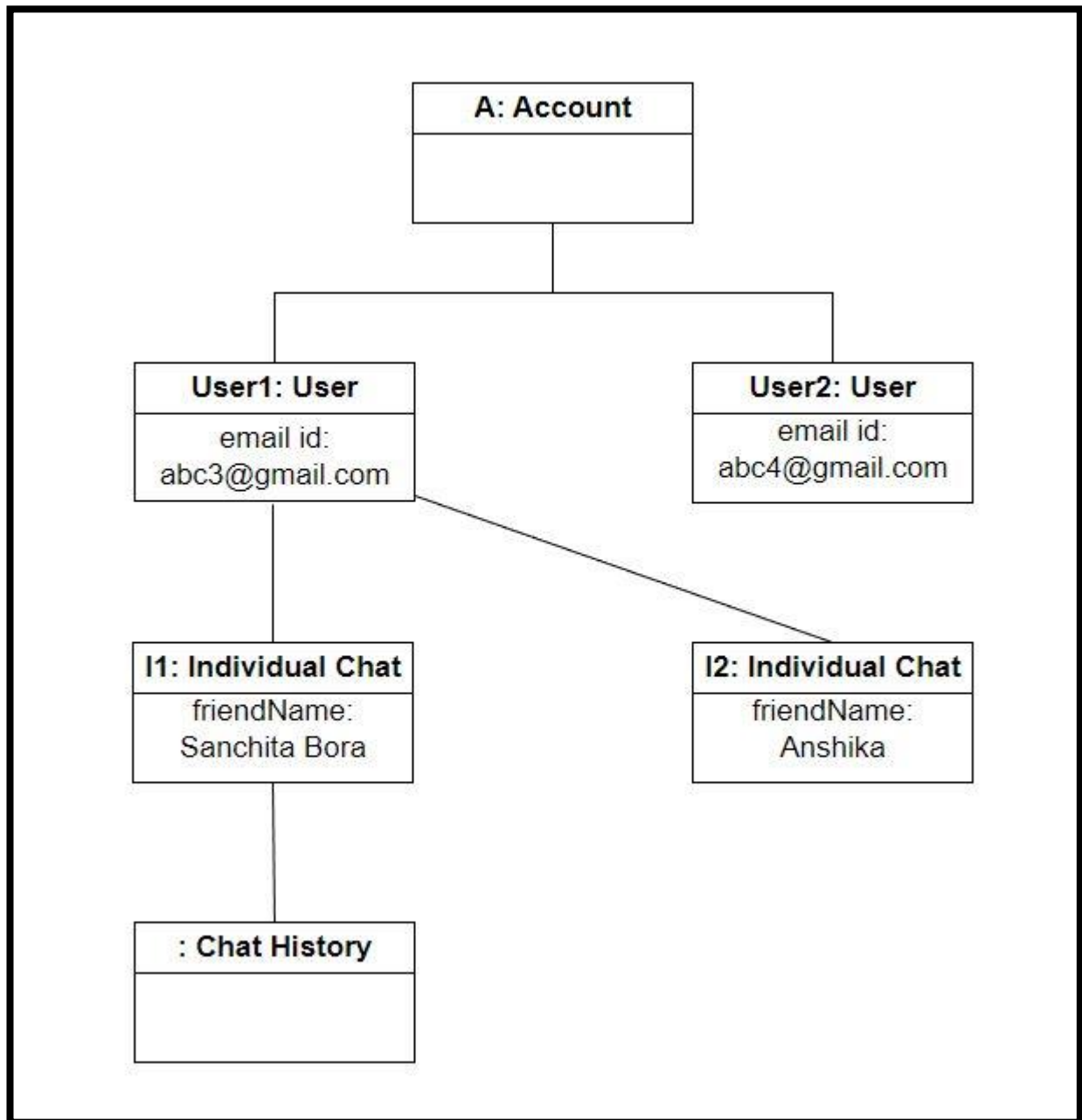
Confirmation:

1. Success: User is able to chat successfully with the intended recipient.
2. Failure: Display message
 - "No user found related to your search term"

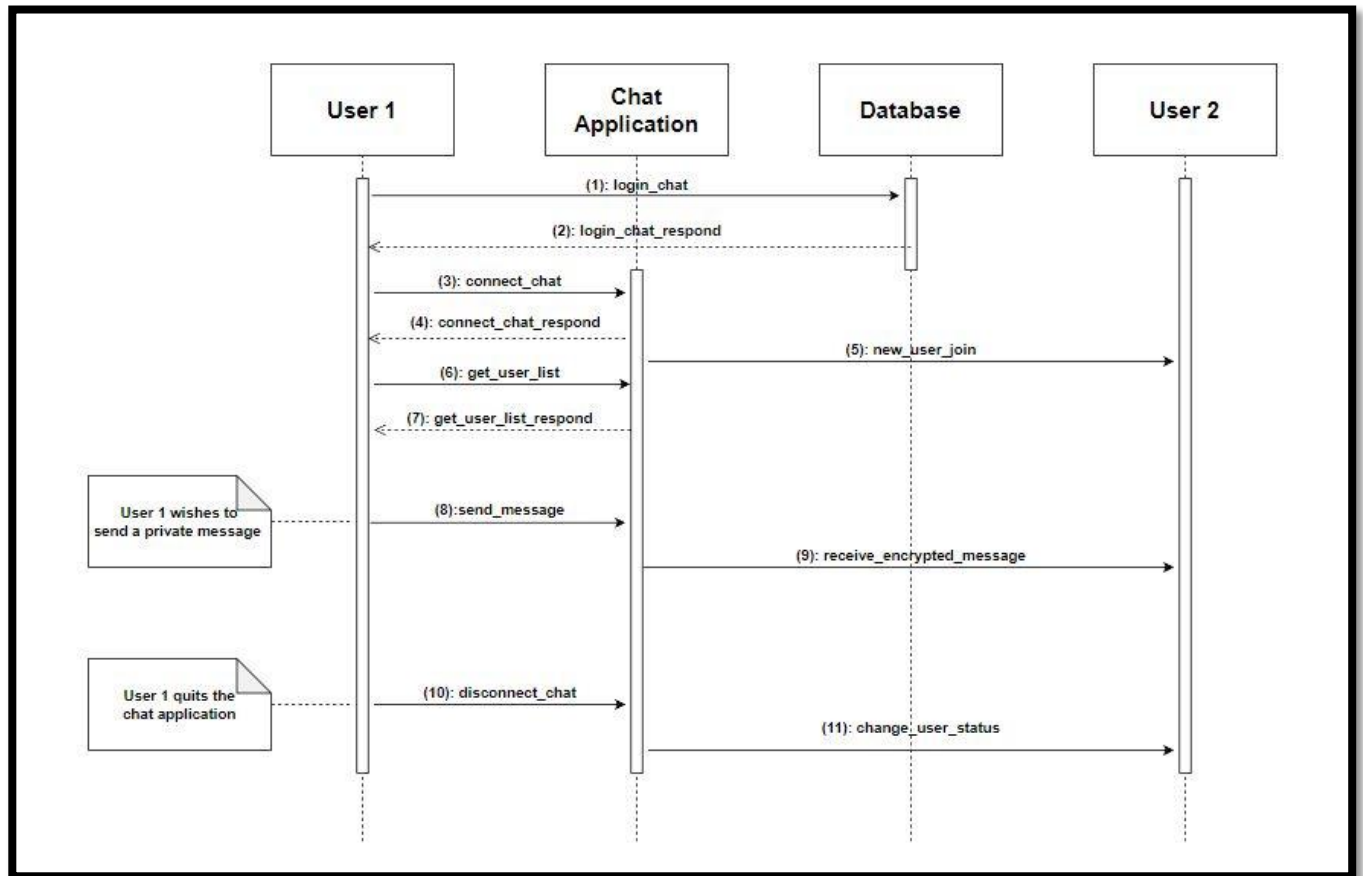
Class Diagram



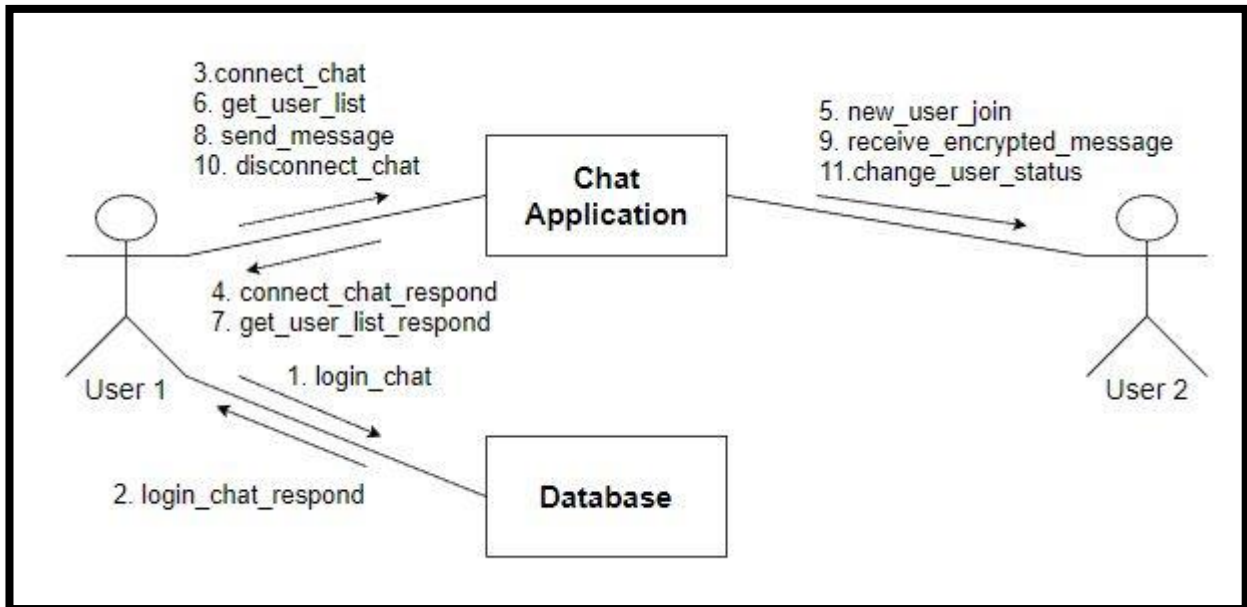
Object Diagram



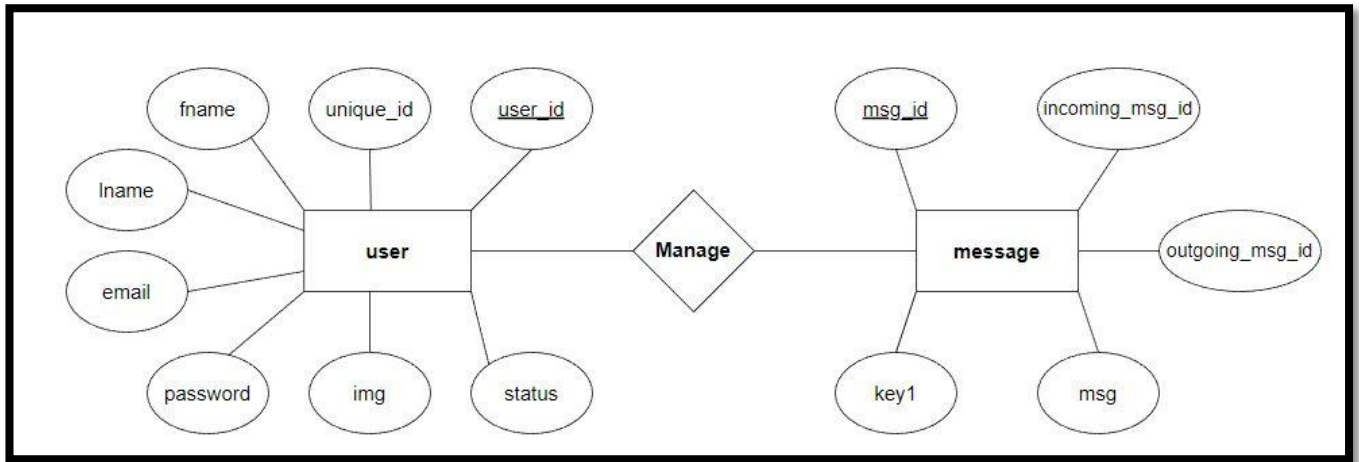
Sequence Diagram



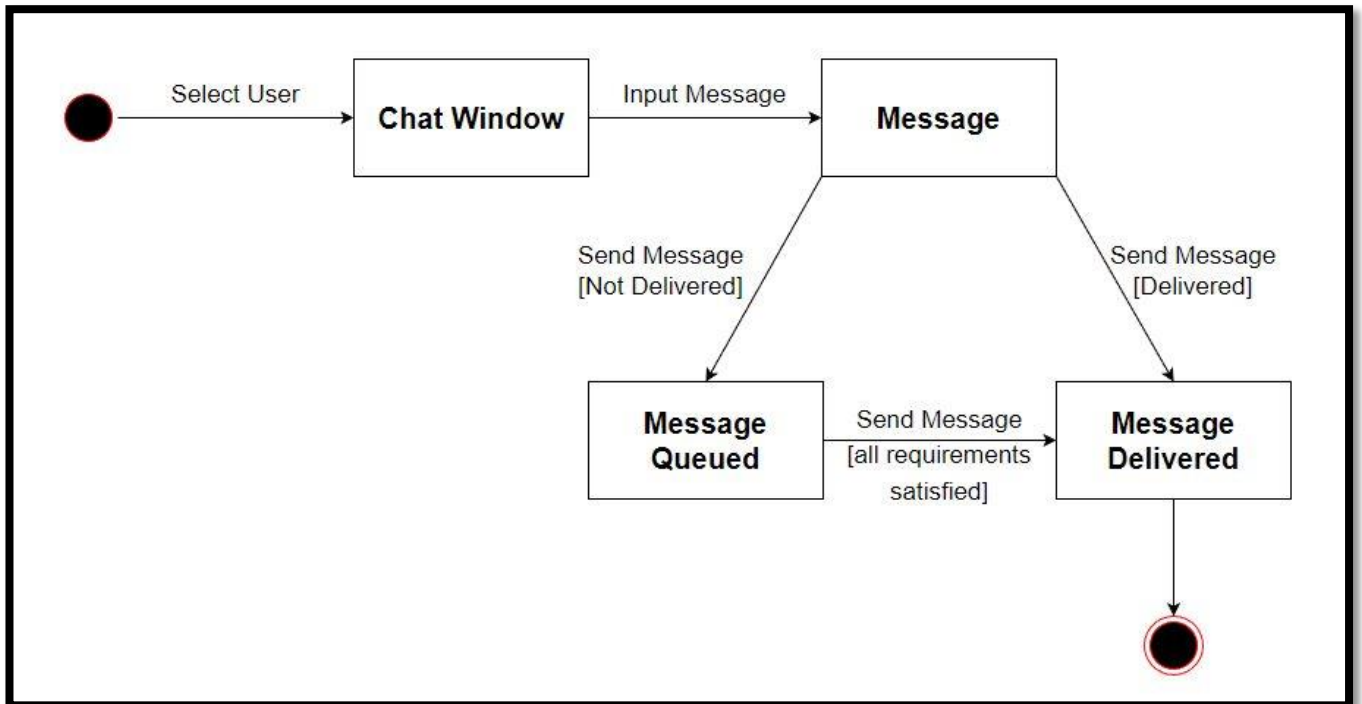
Collaboration Diagram



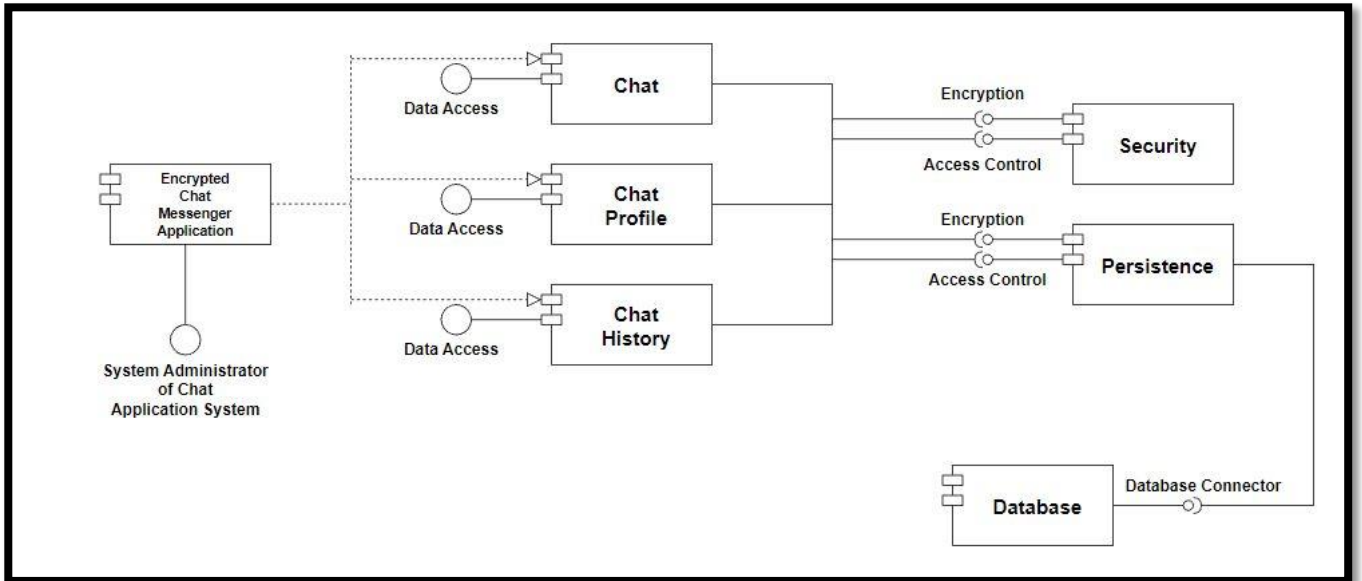
ER Diagram



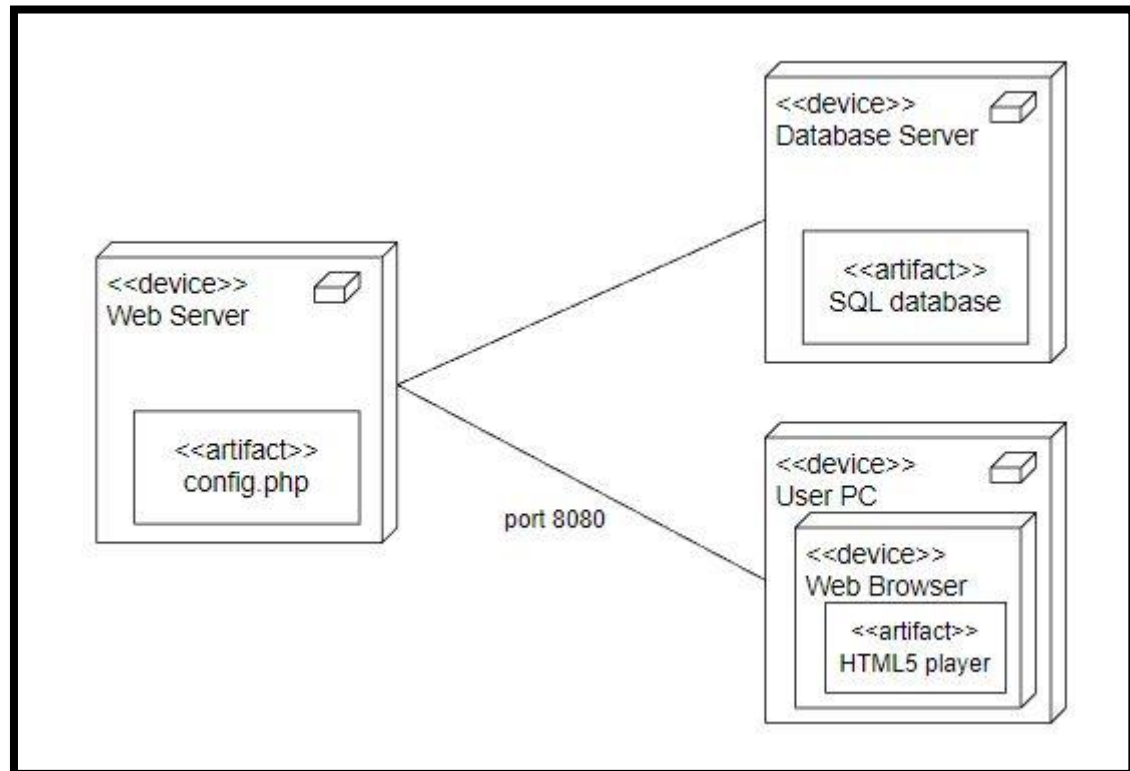
State Chart Diagram



Component Diagram



Deployment Diagram



Screen Shots of Working Project

Chat Buddy

First Name

Sanchita

Last Name

Bora

Email Address

sbora_be20@thapar.edu

Password

...



Select Image

Choose File Profile.jpeg

Continue to Chat

Already signed up? Login now



Sanchita Bora

Active now

Logout

Select a user to start chat



Medhansh Singh Verma

No message available



Anshika Gulati

Hello





Anshika Gulati

Active now

Hi



Hello

Type a message here...



Cyclomatic Complexity

$V(G) = P + 1$, where P is the number of predicate nodes in the flow graph G

- get_chat.php

$V(G) = 4$ (Highlighted) + 1

$V(G) = 5$

```
php > get-chat.php
1  <?php
2      session_start();
3      function decrypt($encryptedText, $password) {
4          $encryptedText = base64_decode($encryptedText);
5          $method = "AES-256-CBC";
6          $iv = substr($encryptedText, 0, 16);
7          $hash = substr($encryptedText, 16, 32);
8          $ciphertext = substr($encryptedText, 48);
9          $key = hash('sha256', $password, true);
10         if (!hash_equals(hash_hmac('sha256', $ciphertext . $iv, $key, true), $hash)) return null;
11         return openssl_decrypt($ciphertext, $method, $key, OPENSSSL_RAW_DATA, $iv);
12     }
13     if(isset($_SESSION['unique_id'])){
14         include_once "config.php";
15         $outgoing_id = $_SESSION['unique_id'];
16         $incoming_id = mysqli_real_escape_string($conn, $_POST['incoming_id']);
17         $output = "";
18         $sql = "SELECT * FROM messages LEFT JOIN users ON users.unique_id = messages.outgoing_msg_id
19                 WHERE (outgoing_msg_id = {$outgoing_id} AND incoming_msg_id = {$incoming_id})
20                 OR (outgoing_msg_id = {$incoming_id} AND incoming_msg_id = {$outgoing_id}) ORDER BY msg_id";
21         $query = mysqli_query($conn, $sql);
22         if(mysqli_num_rows($query) > 0){
23             while($row = mysqli_fetch_assoc($query)){
24                 $decrypt_msg=decrypt($row['msg'],$row['key1']);
25                 if($row['outgoing_msg_id'] === $outgoing_id){
26                     $output .= '<div class="chat outgoing">
27                             <div class="details">
28                                 <p>'. $decrypt_msg .'</p>
29                             </div>
30                         </div>';
31                 }else{
32                     $output .= '<div class="chat incoming">
33                             
34                             <div class="details">
35                                 <p>'. $decrypt_msg .'</p>
36                             </div>
37                         </div>';
38                 }
39             }
40         }else{
41             $output .= '<div class="text">No messages are available. Once you send message they will appear he
42         }
43         echo $output;
44     }else{
45         header("location: ../login.php");
46     }
47 }
48 ?>
```

- insert_chat.php

$V(G) = 2 \text{ (Highlighted)} + 1$

$V(G) = 3$

php > insert-chat.php

```
1  <?php
2      session_start();
3      function encrypt($plaintext, $password) {
4          $method = "AES-256-CBC";
5          $key = hash('sha256', $password, true);
6          $iv = openssl_random_pseudo_bytes(16);
7          $ciphertext = openssl_encrypt($plaintext, $method, $key, OPENSSL_RAW_DATA, $iv);
8          $hash = hash_hmac('sha256', $ciphertext . $iv, $key, true);
9          return base64_encode($iv . $hash . $ciphertext);
10     }
11     if(isset($_SESSION['unique_id'])){
12         include_once "config.php";
13         $outgoing_id = $_SESSION['unique_id'];
14         $incoming_id = mysqli_real_escape_string($conn, $_POST['incoming_id']);
15         $message = mysqli_real_escape_string($conn, $_POST['message']);
16         $key=rand();
17         $encrypt_msg=encrypt($message,$key);
18         if(!empty($message)){
19             $sql = mysqli_query($conn, "INSERT INTO messages (incoming_msg_id, outgoing_msg_id, msg, key1)
20                                     VALUES ({$_incoming_id}, {$_outgoing_id}, '$_encrypt_msg','$_key')") o
21         }
22     }else{
23         header("location: ../login.php");
24     }
25 }
```

- data.php

V(G) = 2 (Highlighted) + 1

V(G) = 3

```
php > data.php
1  <?php
2      function decrypt($encryptedText, $password) {
3          $encryptedText = base64_decode($encryptedText);
4          $method = "AES-256-CBC";
5          $iv = substr($encryptedText, 0, 16);
6          $hash = substr($encryptedText, 16, 32);
7          $ciphertext = substr($encryptedText, 48);
8          $key = hash('sha256', $password, true);
9          if (!hash_equals(hash_hmac('sha256', $ciphertext . $iv, $key, true), $hash)) return null;
10         return openssl_decrypt($ciphertext, $method, $key, OPENSSL_RAW_DATA, $iv);
11     }
12     while($row = mysqli_fetch_assoc($query)){
13         $sql2 = "SELECT * FROM messages WHERE (incoming_msg_id = {$row['unique_id']}
14             OR outgoing_msg_id = {$row['unique_id']}) AND (outgoing_msg_id = {$outgoing_id}
15             OR incoming_msg_id = {$outgoing_id}) ORDER BY msg_id DESC LIMIT 1";
16         $query2 = mysqli_query($conn, $sql2);
17         $row2 = mysqli_fetch_assoc($query2);
18         (mysqli_num_rows($query2) > 0) ? $result = decrypt($row2['msg'], $row2['key1']) : $result = "No message
19         (strlen($result) > 28) ? $msg = substr($result, 0, 28) . '...' : $msg = $result;
20         if(isset($row2['outgoing_msg_id'])){
21             ($outgoing_id == $row2['outgoing_msg_id']) ? $you = "You: " : $you = "";
22         }else{
23             $you = "";
24         }
25         ($row['status'] == "Offline now") ? $offline = "offline" : $offline = "";
26         ($outgoing_id == $row['unique_id']) ? $hid_me = "hide" : $hid_me = "";
27
28         $output .= '<a href="chat.php?user_id='. $row['unique_id'] .'>
29             <div class="content">
30                 
31                 <div class="details">
32                     <span>'. $row['fname']. " " . $row['lname'] .</span>
33                     <p>'. $you . $msg .</p>
34                 </div>
35             </div>
36             <div class="status-dot ' . $offline .' "><i class="fas fa-circle"></i></div>
37         </a>';
38     }
39     ?>
```

Test Cases and Test Reports

Test Case: 1.0		Test Name: Invalid Email		
System: Chat Buddy		Subsystem: Invalid Email		
Designed By: Group 1		Design Date: 20-11-2022		
Executed By: Group 1		Execution Date: 28-11-2022		
Short Description: User signing up with email that does not exist				
Pre-Condition: User with some email id X				
Step	Action	Expected Response	Pass/Fail	Comment
1	Sign Up	System gives a form to input details and signup	Pass	
2	Enter details	Takes Input	Pass	
3	Click on “Continue to chat”	System displays error that email id does not exist and is not valid	Fail	
4	Check post- condition 1		Fail	
Post-Condition: No changes should be reflected in the database				

Test Case: 1.1		Test Name: Sign Up		
System: Chat Buddy		Subsystem: Sign Up		
Designed By: Group 1		Design Date: 20-11-2022		
Executed By: Group 1		Execution Date: 28-11-2022		
Short Description: Sign up a new user using existing email id				
Pre-Condition: User with some email id X				
Step	Action	Expected Response	Pass/Fail	Comment
1	Sign Up	System gives a form to input details and signup	Pass	
2	Enter the details	Take input	Pass	
3	Click on “Continue to chat”	System displays error that email id already exists.	Pass	
4	Check post- condition		Pass	
Post-Condition: No changes should be reflected in the database				

Test Case: 1.2		Test Name: Login		
System: Chat Buddy		Subsystem: Login		
Designed By: Group 1		Design Date: 20-11-2022		
Executed By: Group 1		Execution Date: 28-11-2022		
Short Description: User logging in with incorrect email id or password				
Pre-Condition: User with some email id X and password Y				
Step	Action	Expected Response	Pass/Fail	Comment
1	Login	System gives a form to input details and login	Pass	
2	Enter the details	Take input	Pass	
3	Click on “Continue to chat”	System displays error that email id or password is incorrect	Pass	
4	Check post- condition		Pass	
Post-Condition: No changes should be reflected in the database				

Test Case: 1.3		Test Name: Offline Chat		
System: Chat Buddy		Subsystem: Offline Chat		
Designed By: Group 1		Design Date: 20-11-2022		
Executed By: Group 1		Execution Date: 28-11-2022		
Short Description: Sending message when receiver is offline				
Pre-Condition: 1. The intended receiver is registered and have an account. 2. Receiver should be offline.				
Step	Action	Expected Response	Pass/Fail	Comment
1	Click on “Continue to chat”	Gives the list of registered users	Pass	
2	Select user	Opens chat window	Pass	
3	Send message	The message is successfully delivered	Pass	
4	Check post- condition		Pass	
Post-Condition: The message is being accessed by the receiver when he comes online.				