

# Micro-Messaging Using Message Queues

By group B-18:

1.Ch V S Anirudh

1602-19-737-057

2. Ch Sujith Reddy

1602-19-737-048

GitHub Link: <https://github.com/CHVSAAnirudh>

# Abstract

A lot of software companies which provide services through internet rely on a well-managed messaging system to communicate with customers as well as other services but still we encounter different problems (messages getting jumbled once received).

So, we propose the use of queue to get the messages from one end to possibly multiple ends. Use of queue will guarantee the order of the messages. Also, since only two operations will be taking place at a time they will be very fast and will lead in reduction of traffic. Even multiple queues can be used to reduce the chance of loss of data to zero.



## Actors wise Use Cases(Features of your project):

### User

- Login.
- Change pin.
- Search a contact.
- Add a contact.
- View previous messages.
- Connect to another user.
- Message the connected user.
- Receive messages from the other users.

### Features

- Secured(pin protected)
- No Internet required.
- low latency.

# Use-Case Descriptions

Use Case ID: UC01

Actors: User

Description: Login into the application.

Pre-conditions: None

Post-conditions: You enter into the world of messaging community.

Main Flow:

User	System
	1.Prompts the user to enter the pin
2.Enters the safety pin.	
	3. Validates the information 3.1 If the pin is wrong it prompts the user to re- enter the pin 3.2 Else it shows the list of contacts.

# Use-Case Descriptions

Use Case ID: UC02

Actors: User

Description: Add a contact.

Pre-conditions: must login successfully.

Post-conditions: New contact is added to the contact list and a file is created on the contact name to store messages.

Main Flow:

User	System
	1.Prompts the user to if he wants to add a contact.
2.Selects the option to add contact.	
	3. Prompts for the New contact info.
4. Enters the New Contact Information.	
	5.Creates the contact txt file.



# Use-Case Descriptions

Use Case ID: UC03

Actors: User

Description: read previous conversation with a contact.

Pre-conditions: must login successfully.

Post-conditions: The whole conversation with the selected contact is presented.

## Main Flow:

User	System
	1.Prompts the user to if he wants to view the list of contacts.
2.Selects the option to view contacts.	
	3. Contacts are presented and asks the user to select one to contact to view previous message.
4. Enters the name of a contact	
	5.Presents the whole conversation by reading the contact txt file.

# Use-Case Descriptions

Use Case ID: UC03

Actors: User

Description: Message with a contact.

Pre-conditions: enter the contact from the contact list.

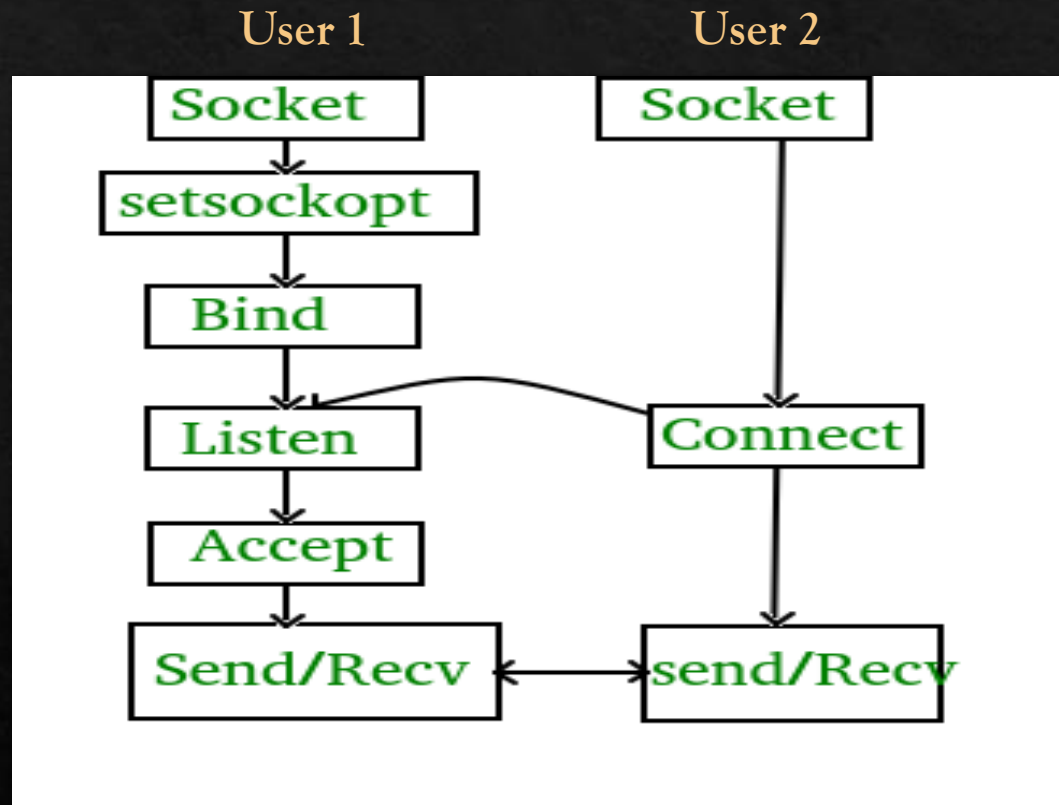
Post-conditions: spread happiness with your messages.

## Main Flow:

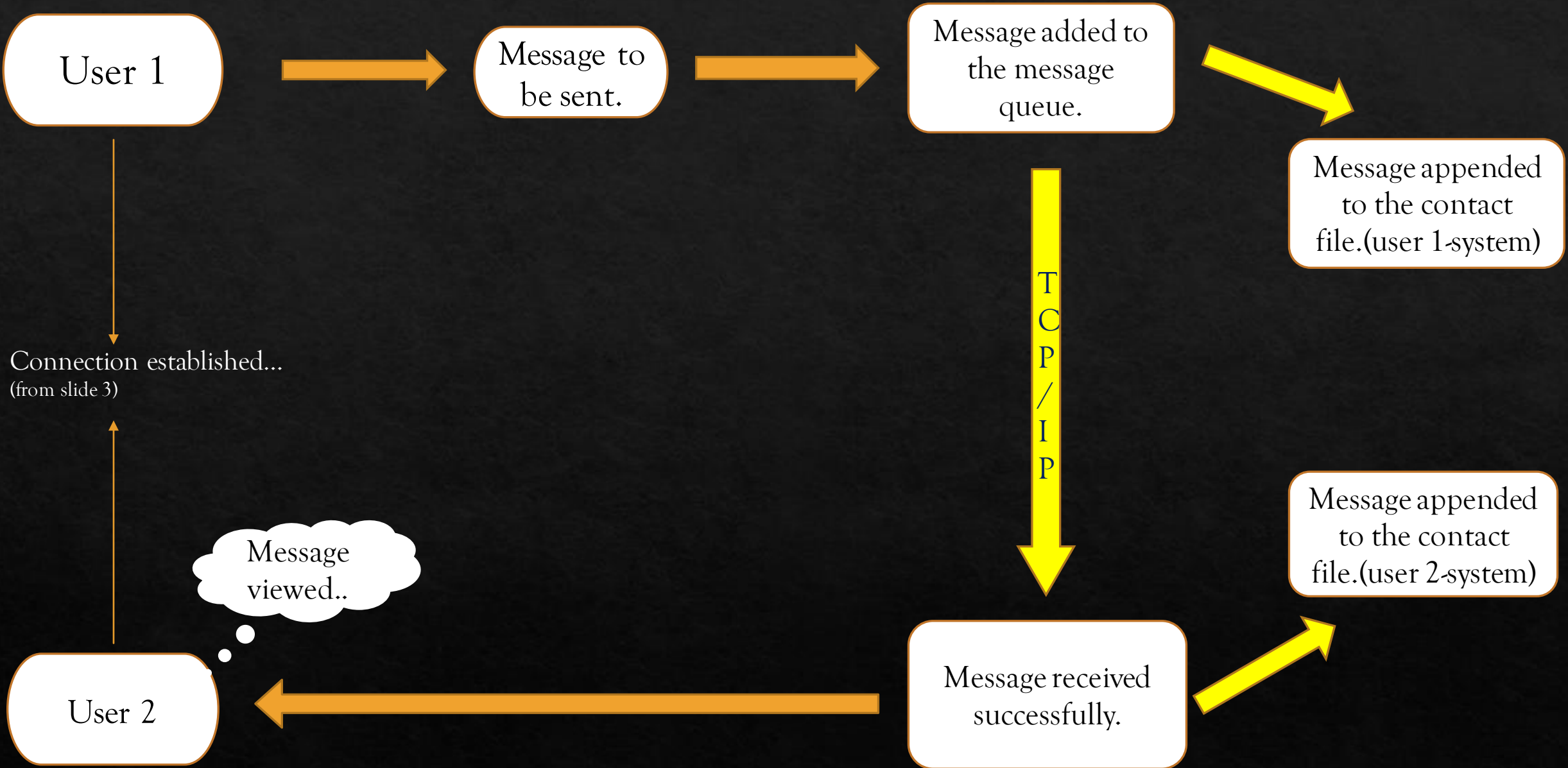
User	System
	1.Prompts the user if he wants to send message.
2.Selects the option to message the contact.	
	3. Promotes the user to enter and send message.
4. Enters the message.	
	5.Presents the the message to the 2 <sup>nd</sup> user.

# Design Of The Project

The Networking section...









Enter the IP of the receiver and get connected.

Write the message to be sent.

Wait for the reply.

Wait for the connection to be established.

Receive the message.

Reply ☺



## Main Flow:

User 1	User 2
1.Enter the IP of the receiver and get connected.	1.Wait for the connection to be established.
2. Write the message to be sent to user User2, once the connection is established.	2.Receiver gets the message sent by the User 1.
3.Wait for the reply from the User2.	3. Reply 😊 by entering the text to User1.

Thank You!!