

*// Phase 3  
Implementation*

# Group 4: Communications

Kai Abillar  
Johnny Hoang

Bryan Madrigal  
John Huynh

Rowwel Ponesto

MEETING MINUTES
GROUP 4 - PHASE 3

DATE	CONTENT	SIGNATURE
10/31/2024	Group Meeting: 5:00pm - 7:30pm, Booked New Meeting for next week November 7	Team
10/31/2024	Visualized and Assigned GUI Components	Team
10/31/2024	Created Client, Server, ClientHandler, GUHandler Classes	Team
10/31/2024	Created individual branches, cloned repository, got network to work	Team
11/1/2024	Updated README, organized repository structure	Kai Ahillar
11/5/2024	Create GUI for main hub with JButtons and JList. Include ActionListener for both.	Johnny Hoang
11/6/2024	Create GUI for User List with JButtons and JList. Include ActionListener for both	Kai Ahillar
11/6/2024	Create GUI for Conversation log and admin frame with JButtons and JList. Include ActionListener for everything	John Hoang
11/7/2024	Create GUI for Chatroom with JList, JButton and JList	Rawel Ponsato
11/4/2024	Create GUI for LoginUI	Bryan Madrigal
11/7/2024	Group Meeting: 5:00pm - 7:35pm , Booked New Meeting for next Week November 14	Team
11/7/2024	Showcased each GUI component that was created, integrated it into GUI Handler, Assigned Week 2 Features	Team
11/7/2024	Made a list of text documents to be used as database : Conversation History & UserAccount	Team
11/7/2024	Created MESSAGE TYPE enum, started Message Class	Kai Ahillar
11/11/2024	Created UserManagement Class & Finished Message Class	Kai Ahillar
11/12/2024	Work on Server and ClientHandler for handling login and sending messages	Johnny Hoang
11/14/2024	Finished UserAccount Class	Bryan Madrigal
11/20/2024	Partial implementation of Chat class (requires discussion)	Rawel Ponsato
11/21/2024	Finished Implementation of RegularUser and Partial Implementation of IUser.	Bryan Madrigal
11/21/2024	Group Meeting: 5:00pm - 7:30pm, Booked New Meeting for December 3. Decided to meet November 25 - 29, 5:00pm onwards via Discord	Team
11/25/2024	First Meeting : Thanksgiving Week, 5:00pm - 7:00pm	Team
11/25/2024	Expand upon connection handling and coded basic passing Message Object between Server and Client	Johnny Hoang
11/25/2024	Presented how Message Passing should work, Renamed ChatRoom to ChatList class (duplication and better representative of code)	Johnny Hoang
11/25/2024	Dived deeper and Delegated tasks for: (1) ChatList, (2) UserAccount, (3) Chat, (4) Conversation History, (5) Passing Messages	Team
11/25/2024	Implemented ChatList Class and created ChatListManager Class (INCOMPLETE) and ChatList database	Kai Ahillar
11/26/2024	Second Meeting : Thanksgiving Week, 5:00pm - 6:34pm	Team
11/26/2024	Presented how ChatList and ChatListManager works	Kai Ahillar
11/26/2024	Solidified format for UserAccount class and ConversationHistory class	Team
11/26/2024	Modified UserAccount and applied better implementation of UserManagement class	Bryan Madrigal
11/27/2024	Third Meeting : Thanksgiving Week, 5:00pm - 6:30pm	Team
11/27/2024	Presented UserAccount Changes	Bryan Madrigal
11/27/2024	Expand more upon Server to initialize UserAccount, and Client handling to receive. Further discussion on how to implement Chat, Convoll History, and Con	Johnny Hoang
11/27/2024	Figure out what to do for Conversation History Class, Chat Class, and Conversation Log	Team
11/29/2024	Fourth Meeting : Thanksgiving Week, 4:30pm - 8:07pm, Booked New Meeting for December 2, 2:00pm - 8:00pm	Team
11/29/2024	Finished Conversation History Class & Conversation Log Class	John Hoang
11/29/2024	Collaborated with John to partially finish Chat	Rawel Ponsato
11/20/2024	Trying to connect the concept of Chat Class to everything, everywhere all at once	Team
12/1/2024	Merging everyone's codes. Finalizing some designs and conclude what still missing.	Johnny Hoang
12/2/2024	Final Meeting: 2:00pm - 9:00pm	Team
12/2/2024	Initialized Powerpoint Presentation	Team
12/2/2024	Polished ChatList, UserManagement	Team
12/2/2024	Implemented Create Chat, & Delete Chat	Team
12/2/2024	Delegated Tasks for (1) updating diagrams, (2) adding logo to gui frame, (3) updating documentation	Team
12/3/2024	PRESENTATION	Team

all in all,  
we met 10 times

1,422 total commits



# com.com

[com]pany  
[com]munications



demo

# BEHIND THE SCENES

# NETWORK ARCHITECTURE OVERVIEW



The diagram illustrates the network architecture overview with four components arranged horizontally. Each component is represented by a circular icon with a purple-to-orange gradient. Below each icon is a text block describing its function. The components are: SERVER, CLIENT, CLIENT HANDLER, and GUI HANDLER.

## SERVER

manages all  
connections and  
maintains  
databases

## CLIENT

user application  
that connects to  
server and  
provides chat  
interface

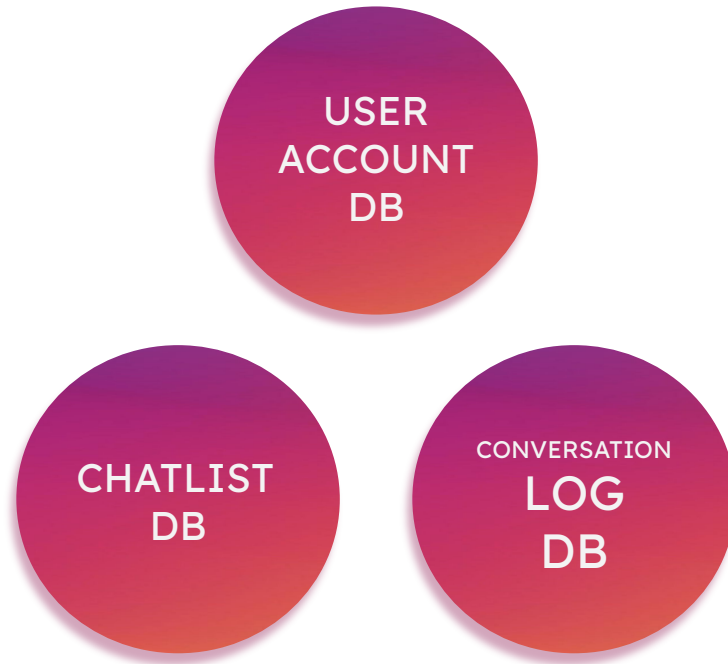
## CLIENT HANDLER

server-side thread  
that manages  
individual client  
connections

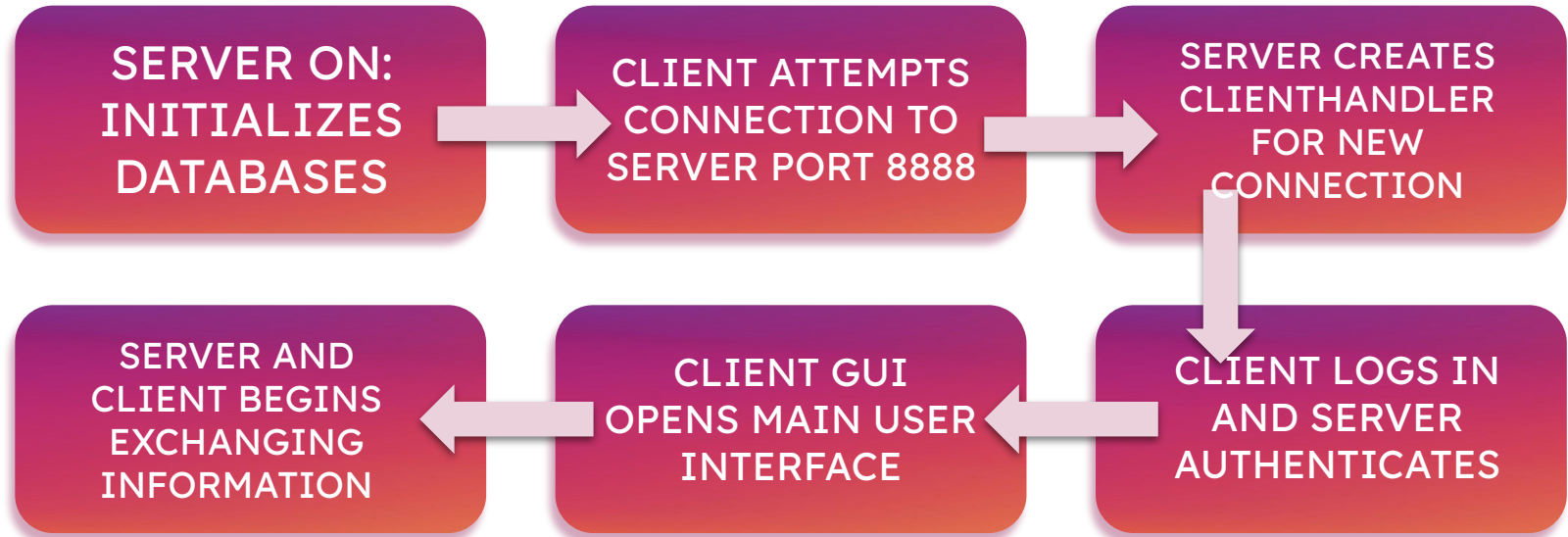
## GUI HANDLER

manages all user  
interface  
elements and user  
interactions

# SERVER DATABASE STRUCTURE

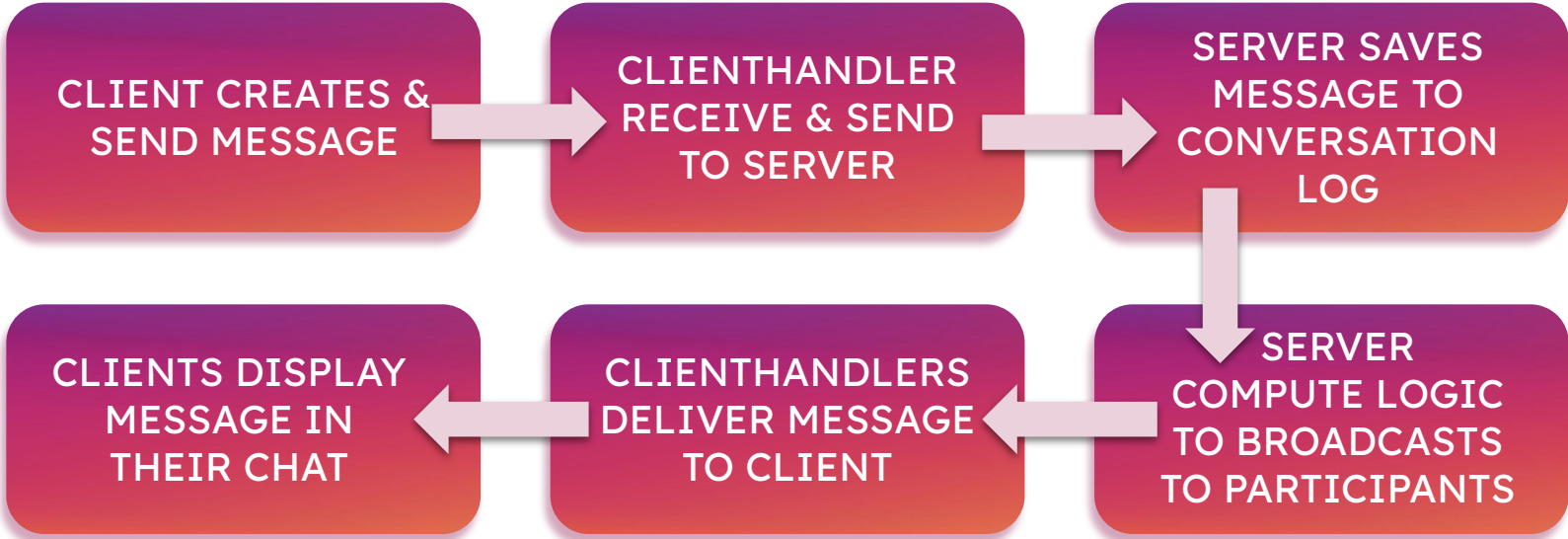


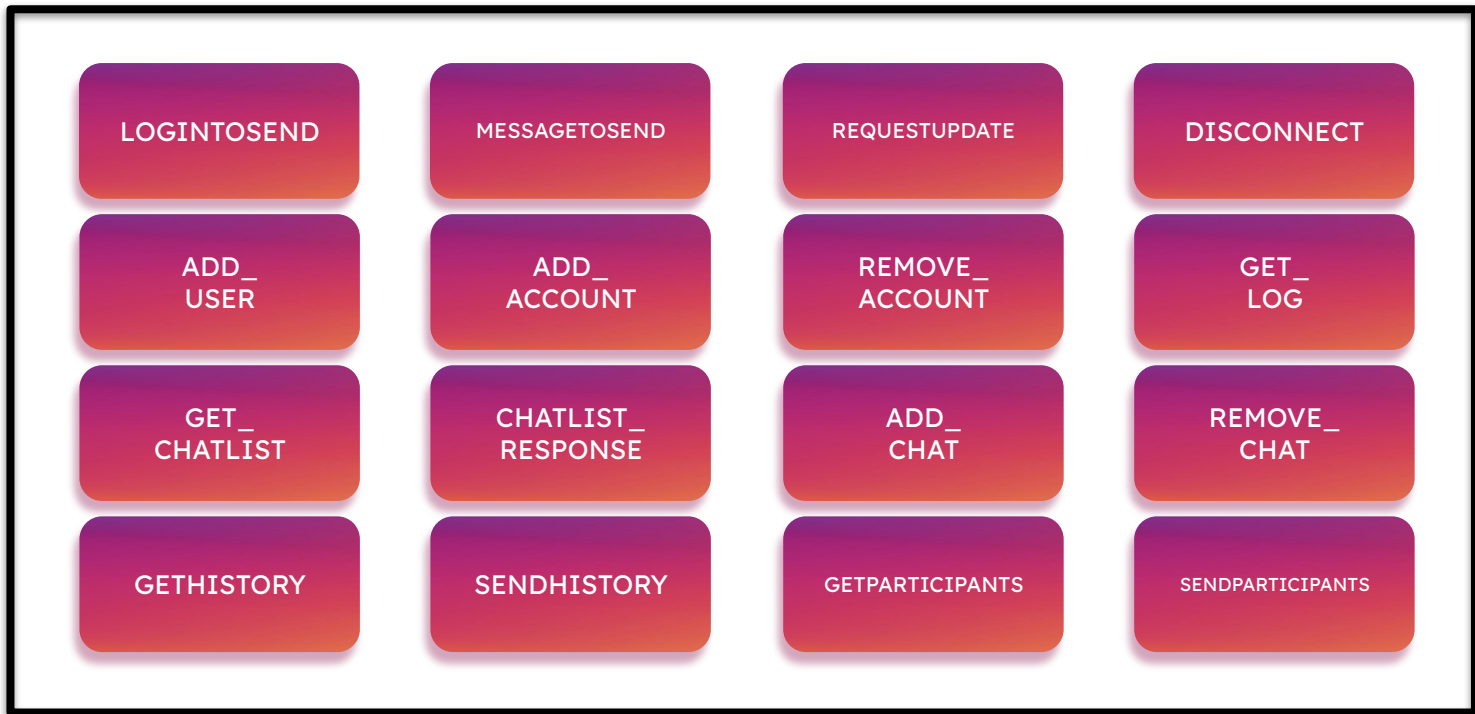
# CLIENT-SERVER CONNECTION





# MESSAGE BROADCASTING FLOW



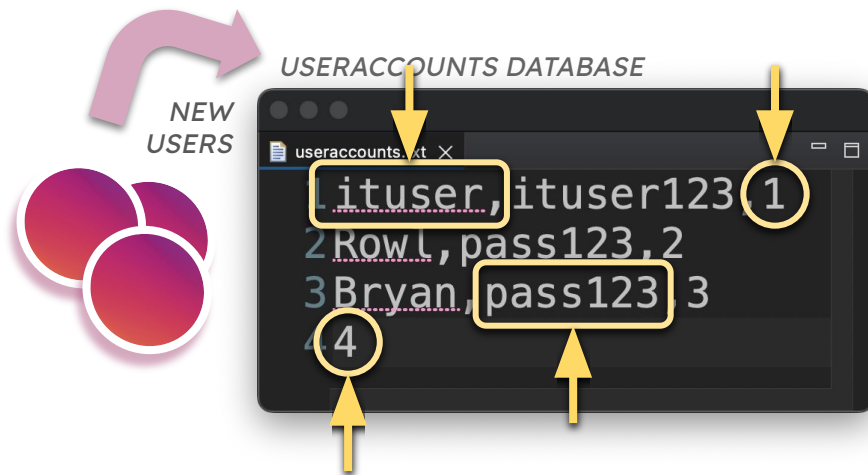


# 16 MESSAGE TYPES

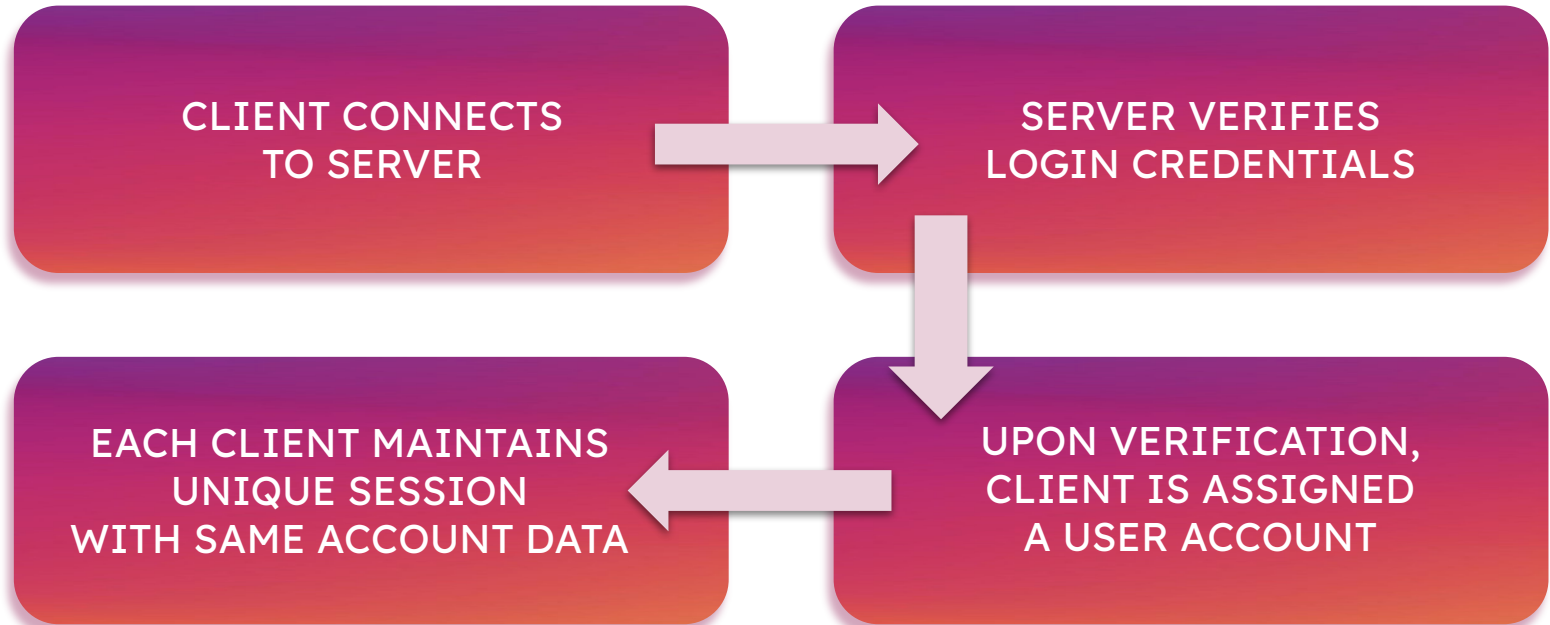
- MANAGES CHAT OPERATIONS
- HANDLES USER AUTHENTICATION
- CONTROLS USER MANAGEMENT
- MAINTAINS CONVERSATION HISTORY

# ACCOUNT CREATION & STORAGE

- NEW **UserAccounts** ARE AUTOMATICALLY STORED IN SERVER DATABASE
- EACH ACCOUNT HAS UNIQUE IDENTIFIERS AND PROPERTIES



# LOGIN PROCESS



# I.T. USER SPECIFICATIONS

- ❖ I.T. USERS ALWAYS HAS ID#1
- ❖ SYSTEM ALLOWS ONLY ONE(1) I.T. USER
- ❖ I.T. USER HAS EXCLUSIVE BUTTON IN THE G.U.I.; ALLOWS I.T. TO CREATE/REMOVE **UserAccounts** AND VIEW **ConversationLog**

# SESSION MANAGEMENT

- ❑ ACTIVE **UserAccount** CAN NOT BE USED BY MULTIPLE **Clients**; THIS PREVENTS DUPLICATE LOG INS
- ❑ EACH **UserAccount** MAINTAINS CONSISTENT DATA ACROSS SESSIONS

# CHAT SYSTEM ARCHITECTURE



## CHAT STRUCTURE

- ❖ UNIQUE ID
- ❖ HAS DEDICATED **ConversationHistory** OBJECT
- ❖ LIST OF PARTICIPANTS

- ❖ CONTAINS ONE **Chat** INSTANCE
- ❖ DISPLAYS CHAT'S **ConversationHistory** IN JList
  - ❖ PROVIDES MESSAGE SENDING INTERFACE



## CHATROOM STRUCTURE

# CHAT LIST ARCHITECTURE



```
1ChatListID:1
20
3ChatListID:2
41,2,3,4,5
5ChatListID:3
61
```

## CHAT LIST DB

- ❖ A **UserAccount's** LIST OF Chat OBJECTS
- ❖ LINES 2, 4, & 6 ARE CHATLISTS
- ❖ LINES 1, 3, & 5 ARE **ChatList** ID'S THAT MATCH A **UserAccount's** ID

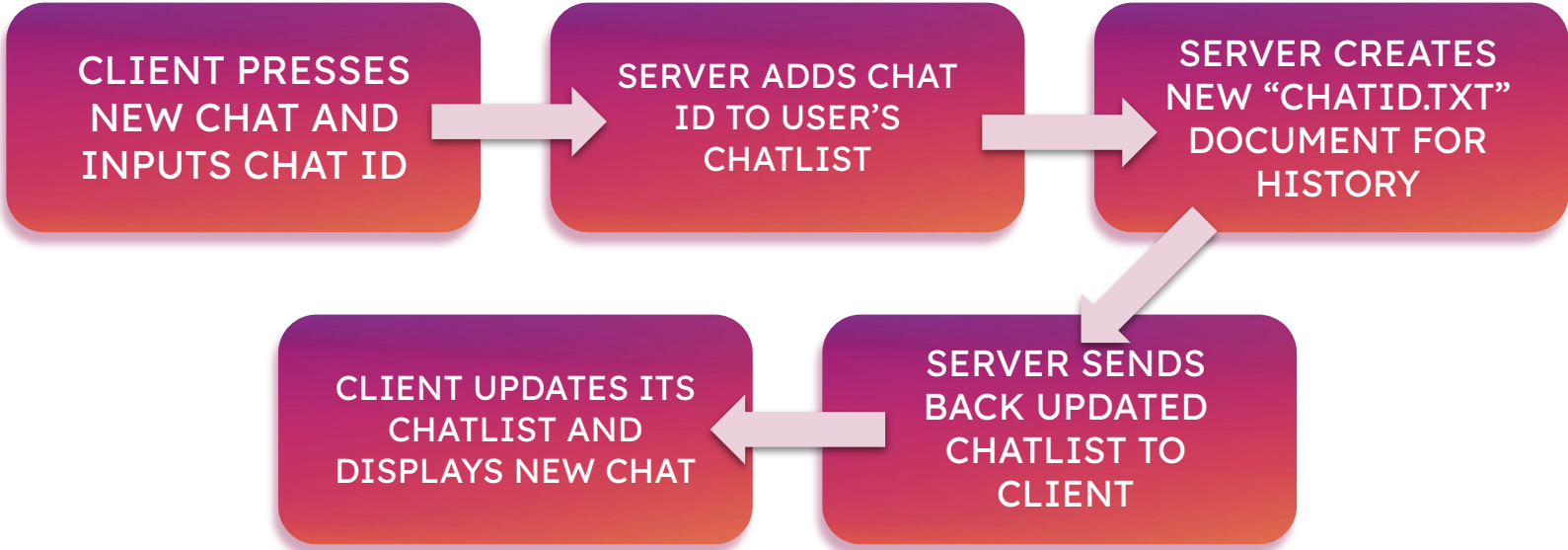


# CHATLIST CREATION



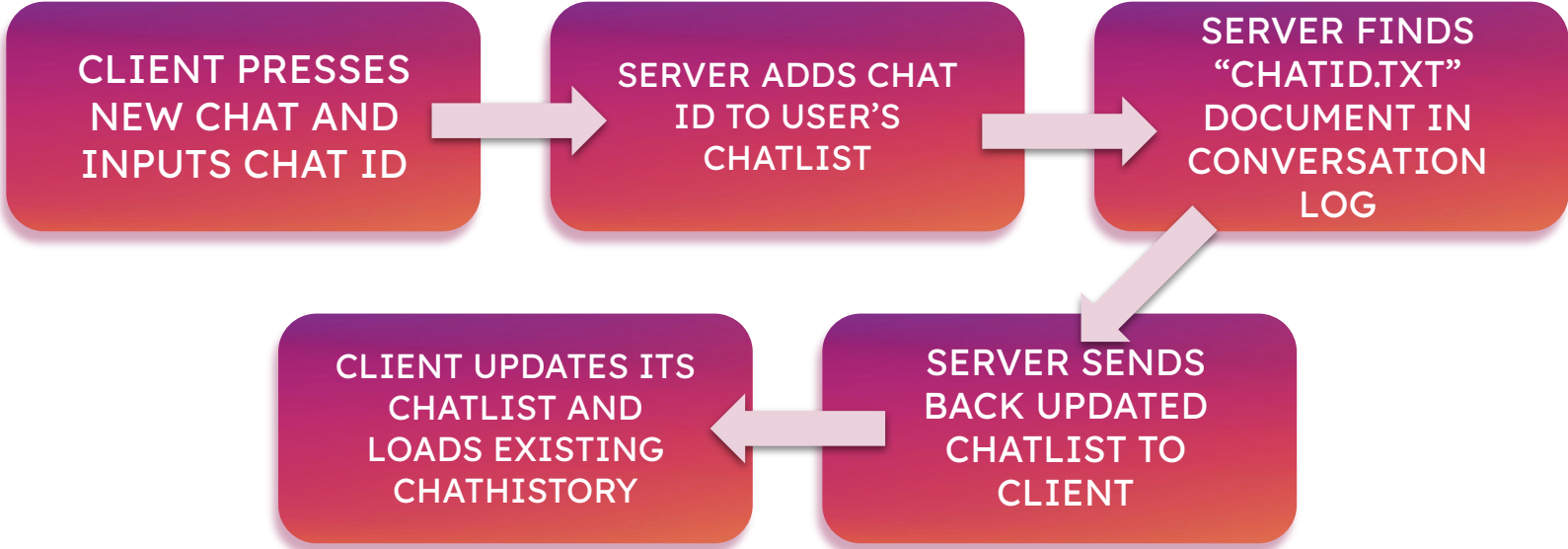
# CHAT CREATION:

IF CHAT DOES NOT EXIST



# CHAT CREATION:

IF CHAT DOES EXIST



**Chat** HOLDS A UNIQUE **ConversationHistory** OBJECT FOR MESSAGE STORAGE



**ChatRoom** DISPLAYS A **Chat's ConversationHistory** TO USERS

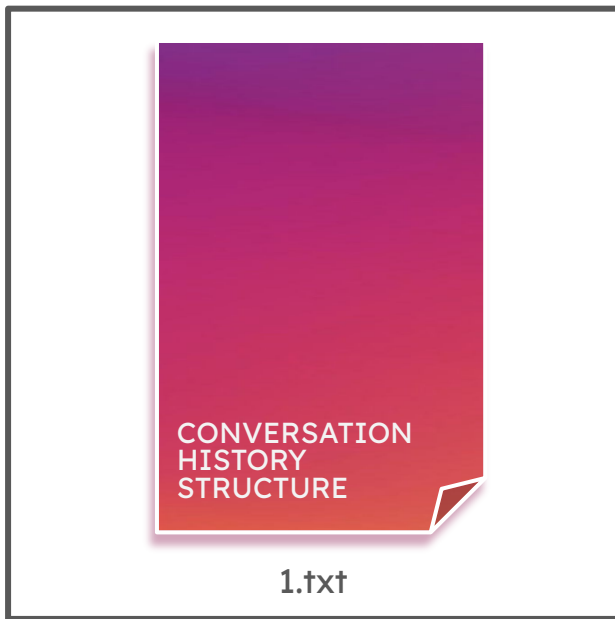


**ChatList** CONTAINS A **<List>** OF **Chat** OBJECTS A **UserAccount** HAS

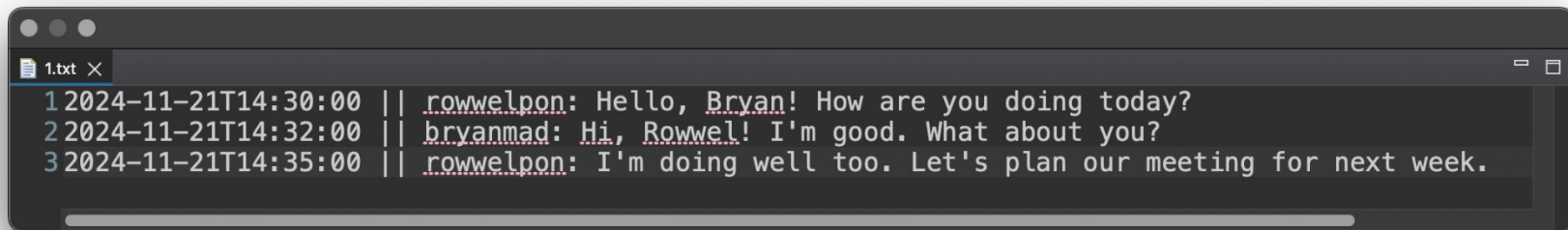


**ChatListManager** MANAGES CENTRAL DATABASE OF ALL **UserAccount's Chat** MEMBERSHIP

# CONVERSATION MANAGEMENT ARCHITECTURE



- ❖ OWNED BY A SINGLE **Chat** OBJECT
- ❖ CONTAINS **<LIST>** OF MESSAGES
- ❖ TRACKS **MESSAGE** TIMESTAMPS
- ❖ MAINTAINS SENDER INFORMATION
- ❖ A NEW **Chat** CREATES A NEW **ChatHistory** THAT CREATES A NEW TEXT FILE IN THE SERVER



```
1 2024-11-21T14:30:00 || rowwelpon: Hello, Bryan! How are you doing today?
2 2024-11-21T14:32:00 || bryanmad: Hi, Rowwel! I'm good. What about you?
3 2024-11-21T14:35:00 || rowwelpon: I'm doing well too. Let's plan our meeting for next week.
```

STORES MESSAGES CHRONOLOGICALLY



MANAGES MESSAGE DISPLAY ORDER



PROVIDES HISTORY RETRIEVAL METHODS



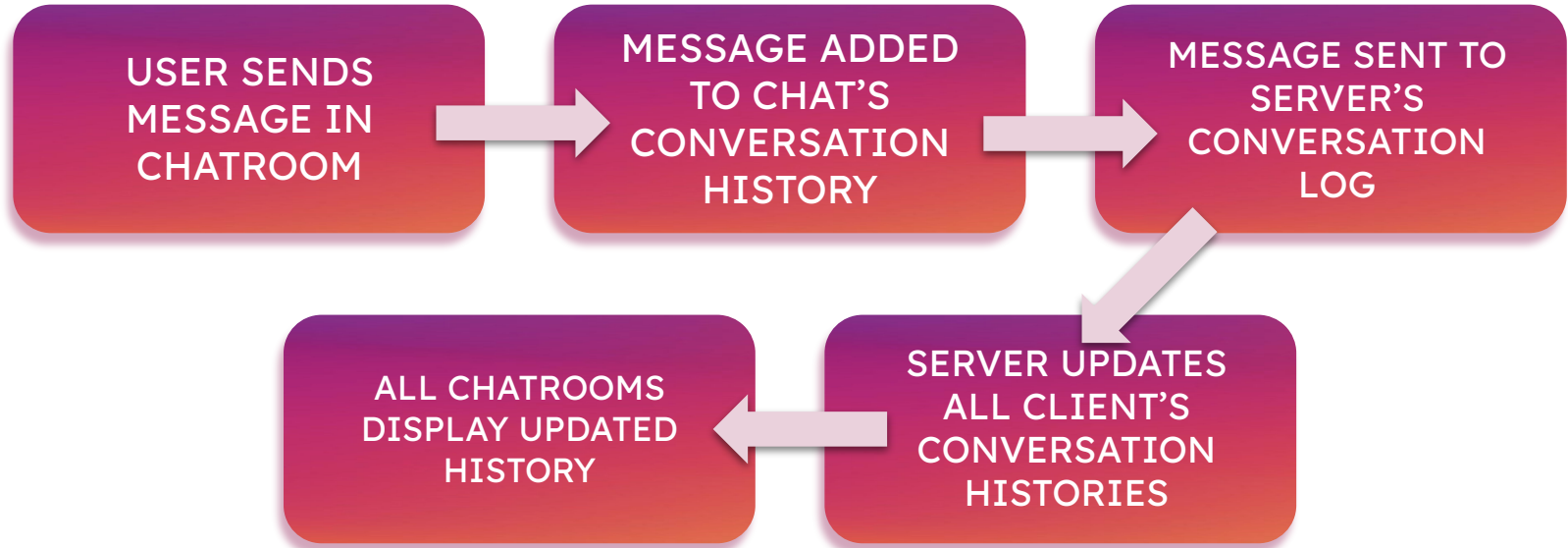
UPDATES IN REAL-TIME

# CONVERSATION MANAGEMENT ARCHITECTURE



- ❖ CENTRAL DATABASE OF ALL CONVERSATIONS
- ❖ STORES ALL **ChatHistories**
- ❖ MAPS HISTORIES TO **Chat** ID KEY(HISTORY):VALUE(CHATID)
- ❖ ENSURES DATA PERSISTENCE
- ❖ HANDLES **Message** OBJECTS SENT WITH TYPE.HISTORYREQUEST

# MESSAGE FLOW





EACH Chat HAS ONE(1) ConversationHistory



Server HAS ONE(1) ConversationLog FOR ALL CHATS

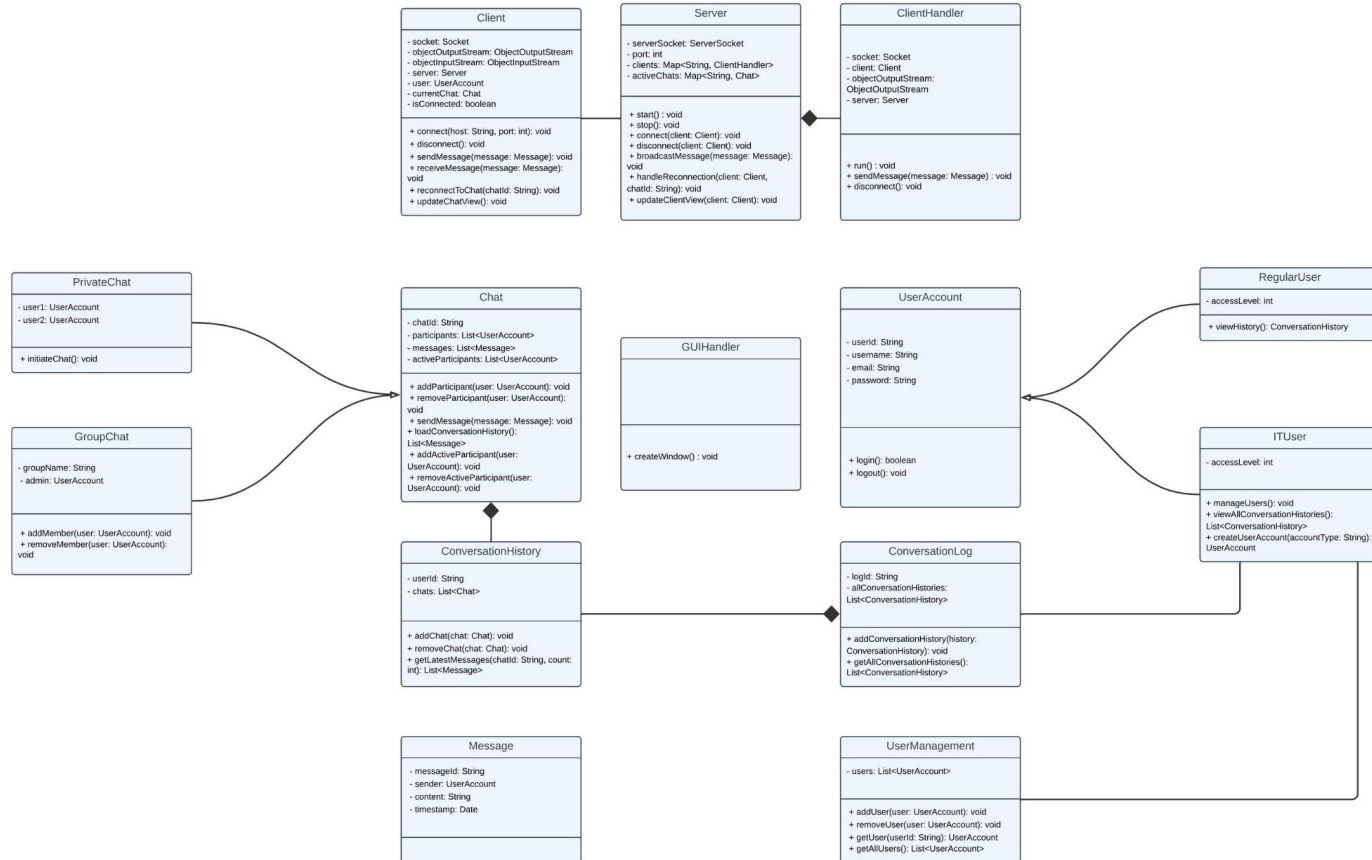


AUTOMATIC UPDATES ACROSS ALL CONNECTED CLIENTS

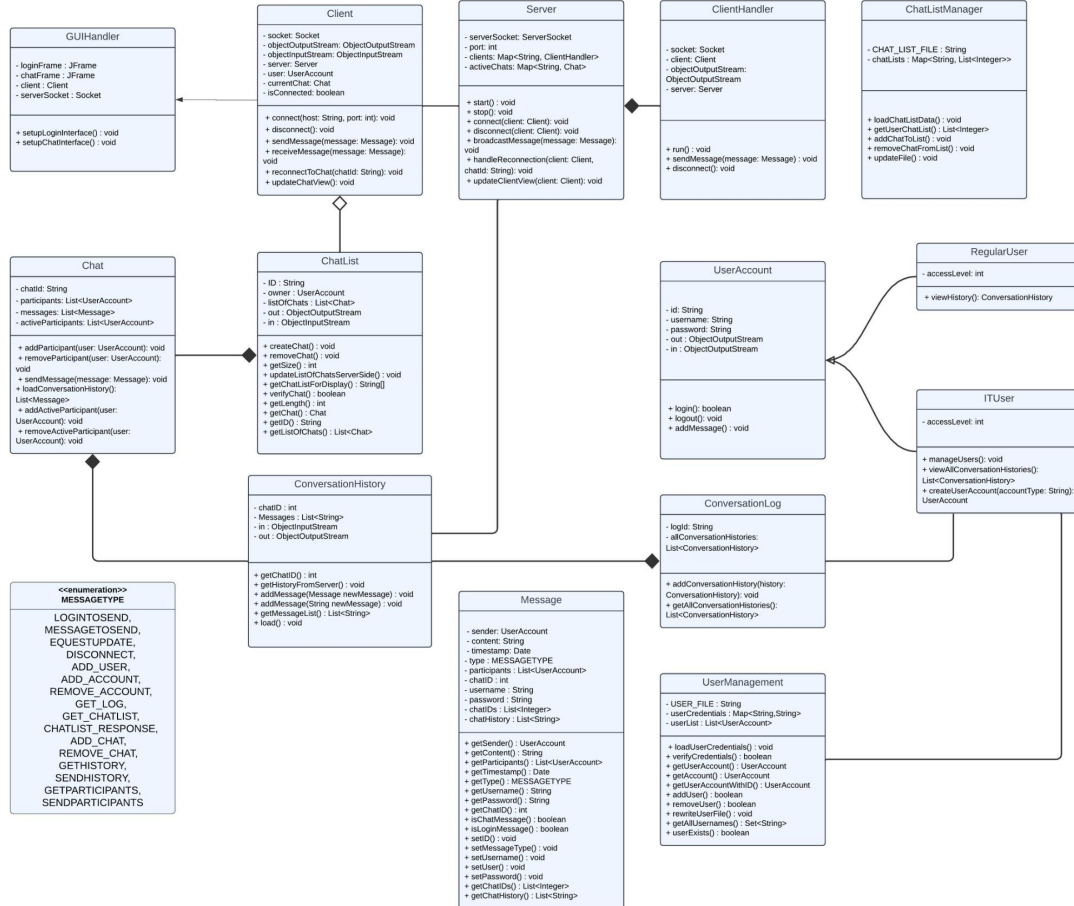


REAL-TIME SYNCHRONIZATION BETWEEN HISTORY AND LOG

# Communications System Class Diagram



# Communications System Class Diagram



# com.com

[com]pany  
[com]munications



# Group 4: Communications

Kai Abillar  
Johnny Hoang

Bryan Madrigal  
John Huynh

Rowwel Ponesto