

|  |  |
| --- | --- |
| Name | Anas Ilyas |
| Course Title | OOP |
| Date | 11/10/2024 |
| Reg no | SP24-BSE-60 |
| Submitted to | Shahid Bhatti |

**Overview Main Class:**

The Main class serves as the entry point for our program. It allows users to interact with the messaging system by performing various actions such as adding contacts, sending messages, displaying message histories, deleting messages, and searching for contacts or messages. This interaction happens through a menu-based system, where the user selects an action from the available options.

**Overview of the SMS Class:**

The SMS class represents individual text messages that are exchanged between contacts. Each message contains the following details:

The sender's phone number.

The receiver's phone number.

The message content.

The message ID.

The timestamp of when the message was sent.

The SMS class is used by the Message class to store and manage messages between contacts.

Message Class Functions

**General Goal:**

The Message class manages contacts and messages between users. It stores contact names, phone numbers, and message histories. This class allows users to add contacts, send messages, display message histories, delete messages, and search contacts and messages.

**1. addcontact:**

The addcontact function allows the user to add a contact by providing both a name and a phone number. It checks if there is space in the contact list before adding the new contact. If the list is full, it shows an error message.

2. **sendMessage**:

The sendMessage function is responsible for sending a message from a sender to a receiver. It verifies whether the sender exists in the contact list and finds an empty slot to store the message along with the time it was sent.

3. **display**:

The display function shows the message history for a specific contact. It checks for the contact's number and displays all messages associated with that contact, including the sender and timestamp.

4. **delete:**

The delete function removes a message by its ID for a given contact. It searches for both the contact and the message, and if found, deletes the message. If the contact or message ID is invalid, an error message is displayed.

5. **searchContact**:

The searchContact function searches for a contact by their phone number. If the contact is found, it prints the contact's name and number. If the contact is not found, it displays a "contact not found" message.

6. **searchMessage**:

The searchMessage function allows the user to search for a specific message by its message ID. It verifies if the message exists for the given contact, and if found, displays the message.