ОООП

Звіт

Лабораторна робота № 1b

Додаткове завдання № 1 b

Моделювання з використанням UML

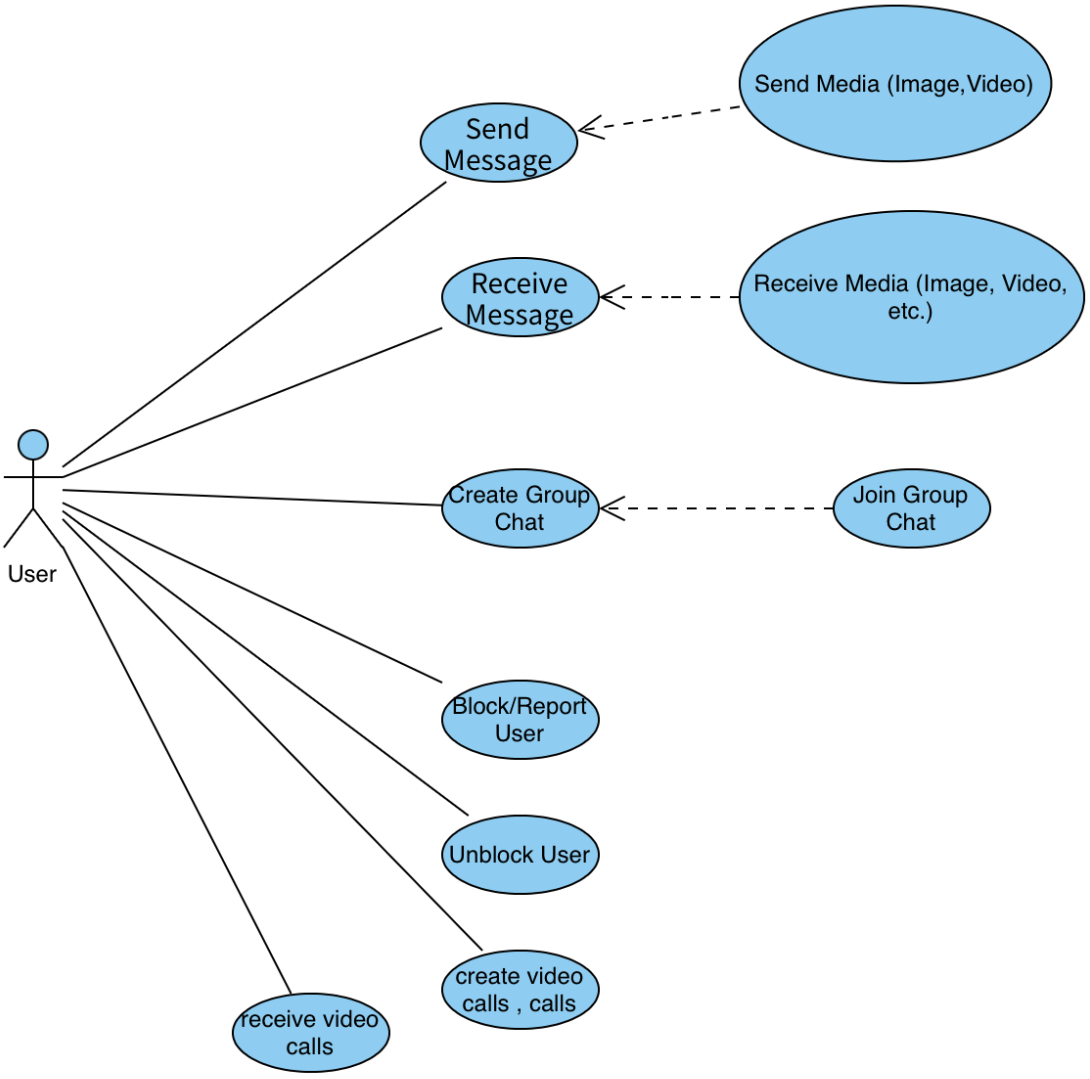
(Conceptual/Domain Modeling)

Мандріченко Ксенія ІПС-22

В даному файлі прикріплені діаграми та текстовий опис до кожної діаграми.

Вибраний сервіс для моделювання :

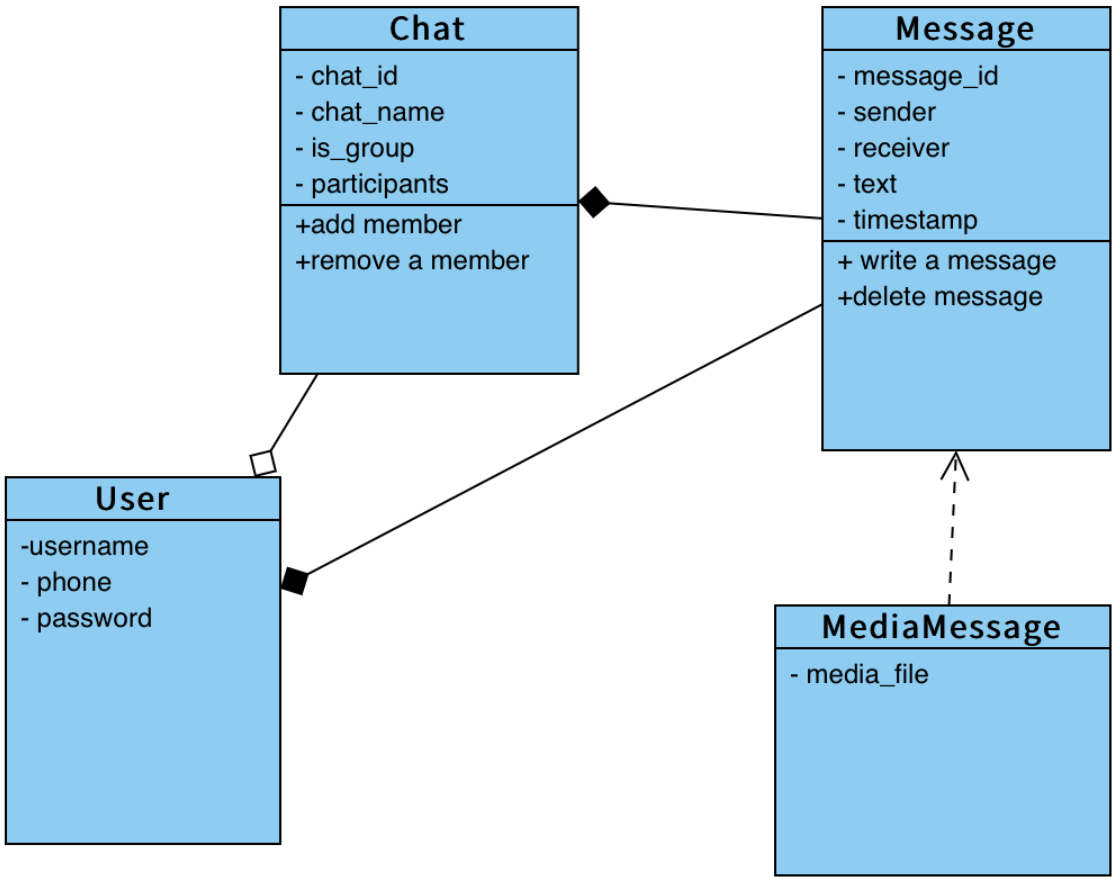
Use Case



In this Use Case diagram, there are four main use cases that involve the actions of the Viber user, which are sending and receiving messages, sending and receiving media (images, videos, etc.), creating and joining group chats, and blocking/reporting or unblocking users.

These use cases have actors that interact with the Viber messaging service. The Viber user can send messages, media, create a group chat or block/report users. Meanwhile, the other Viber users can receive messages, media, and join a group chat.

Class



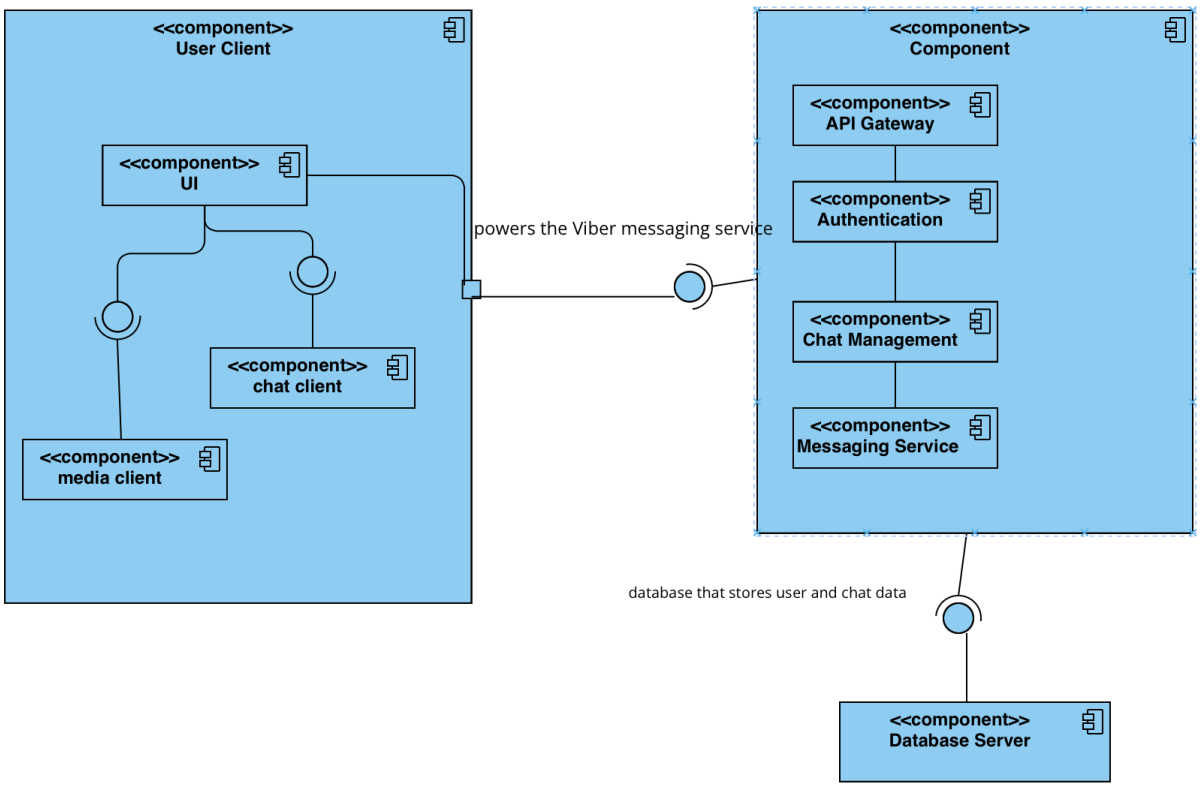
In this Class diagram, there are three main classes that represent the key entities in the Viber messaging service. The User class represents a user of the messaging service, the Chat class represents a chat (which can be a one-on-one chat or a group chat), and the Message class represents a message that is sent or received in a chat.

The User class has attributes such as username, phone, and password, which represent the user's account information. The Chat class has attributes such as chat_id, chat_name, and is_group, which represent the unique identifier for the chat, the name of the chat, and whether the chat is a group chat or a one-on-one chat. The participants attribute in the Chat class is a list of User objects that represent the participants in the chat, functions such as add member and remove member.

The Message class has attributes such as message_id, sender, receiver, text, and timestamp, which represent the unique identifier for the message, the sender of the message, the recipient of the message, the text of the message, and the time the message was sent, functions such as write a message and delete a message .

The MediaMessage class is a subclass of the Message class, which includes an additional attribute called media_file that represents the media (such as an image or video) attached to the message.

Component



In this Component diagram, there are four main components that represent the key parts of the Viber messaging service: the User Client, the Server, the Database Server, and the external services.

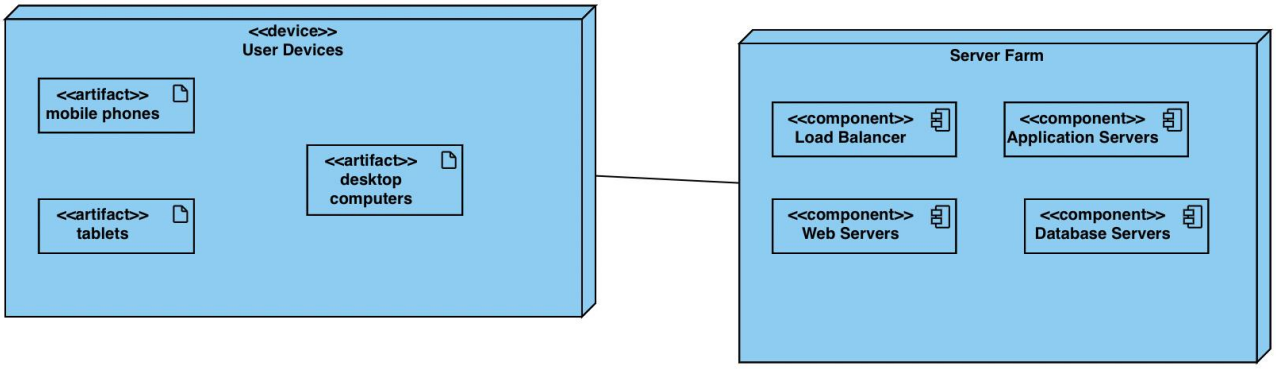
The User Client component represents the client-side application that Viber users use to interact with the messaging service. It includes components for the user interface (UI), the chat client (which handles sending and receiving messages), and the media client (which handles sending and receiving media, such as images and videos).

The Server component represents the server-side application that powers the Viber messaging service. It includes components for the API Gateway (which handles requests and responses between the user client and the server), Authentication (which verifies user identities), Chat Management (which handles creation and management of chats), and Messaging Service (which handles sending and receiving messages).

The Database Server component represents the server-side database that stores user and chat data, including user account information and chat histories.

Finally, the external services component represents any external services that the Viber messaging service may rely on, such as third-party APIs for media hosting or authentication.

Deployment

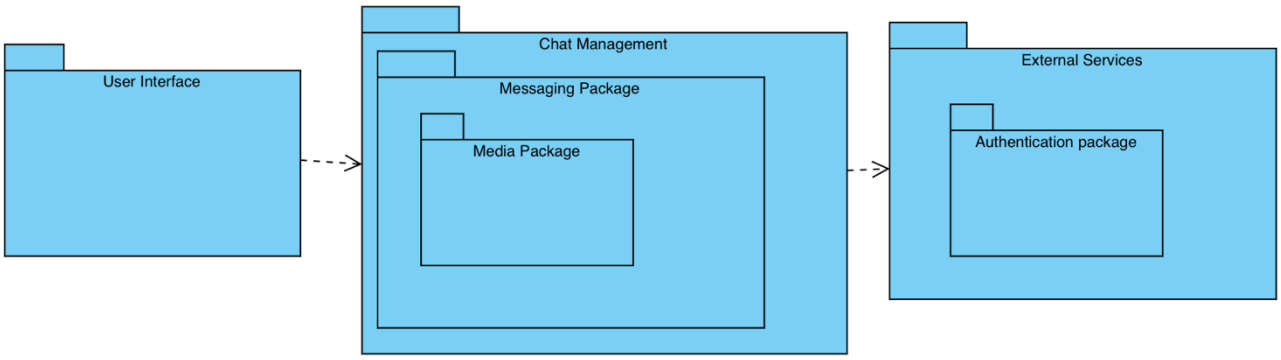


In this Deployment diagram, there are two main nodes that represent the key parts of the Viber messaging service: the User Devices and the Server Farm.

The User Devices node represents the various client devices that Viber users may use to interact with the messaging service, including mobile phones, tablets, and desktop computers.

The Server Farm node represents the collection of servers that power the Viber messaging service. It includes components for a Load Balancer (which distributes incoming requests among multiple web servers to balance the load), Web Servers (which handle requests and responses between the user client and the server), Application Servers (which provide the logic for handling user requests), and Database Servers (which store user and chat data, including user account information and chat histories).

Package



In this Package diagram, there are several packages that represent the major functional areas of the Viber messaging service:

The User Interface package represents the package that includes all of the components that are involved in presenting the user interface of the Viber app to users, such as screens, buttons, and menus.

The Messaging Package package represents the package that includes all of the components that are involved in the sending and receiving of messages, such as message composition, encryption, and transport.

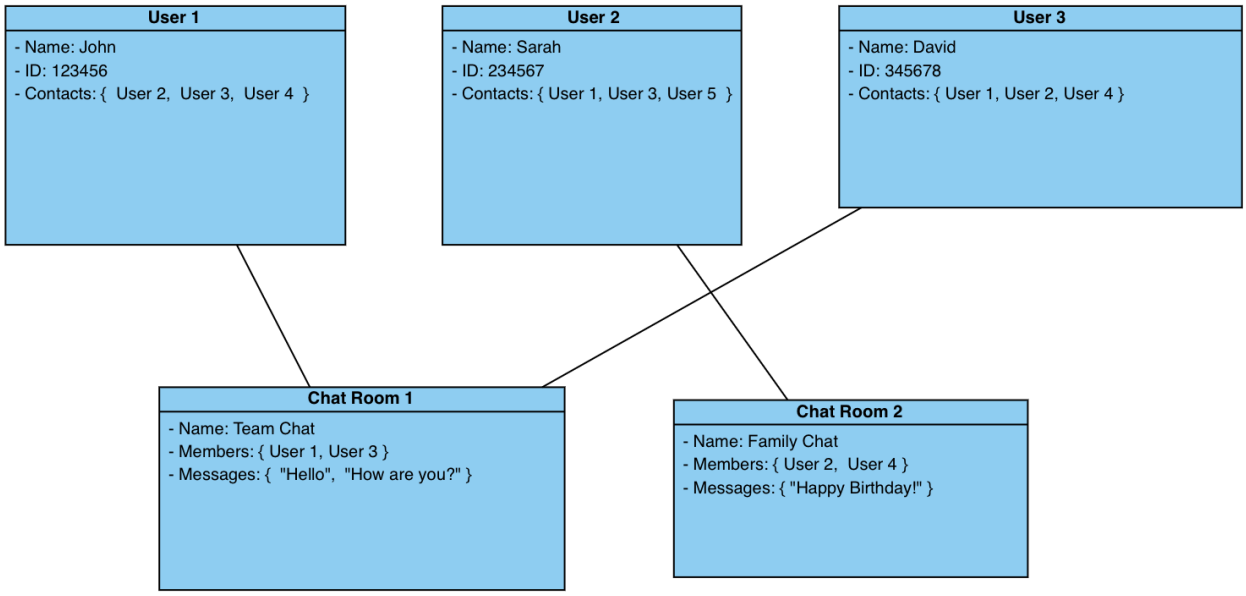
The Authentication package represents the package that includes all of the components that are involved in verifying the identity of Viber users, such as login and password management.

The Media Package package represents the package that includes all of the components that are involved in the sending and receiving of media, such as images and videos.

The Chat Management package represents the package that includes all of the components that are involved in the creation, management, and deletion of chats in Viber, such as chat invitations and notifications.

The External Services package represents the package that includes all of the components that are involved in interfacing with external services, such as media hosting or authentication.

Object

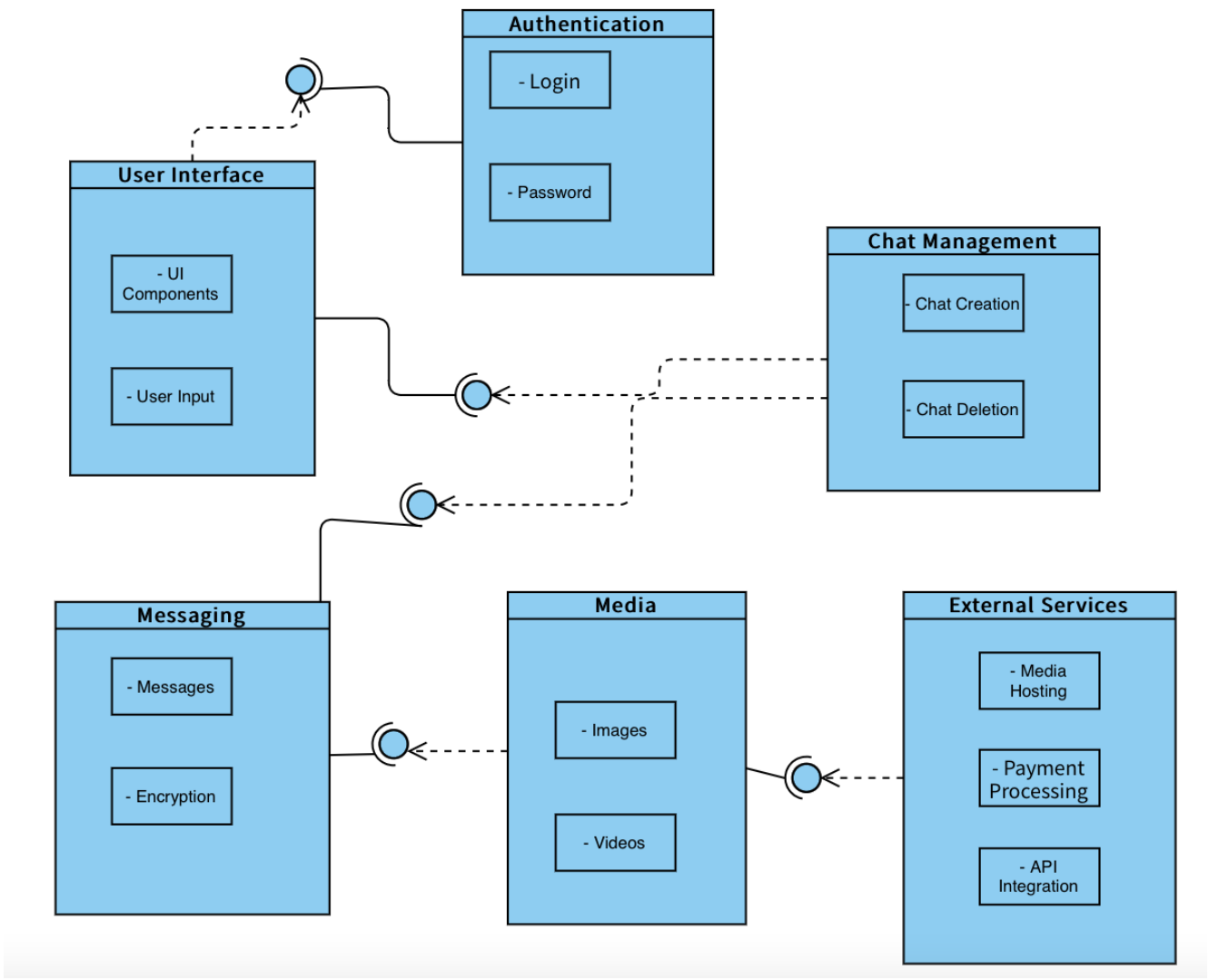


In this Object diagram, there are several objects that represent key entities in the Viber messaging service, including users and chat rooms.

Each User object represents a Viber user and includes several attributes such as the user's name, user ID, and contact list. In this example, there are three user objects representing User 1, User 2, and User 3.

Each Chat Room object represents a chat room in Viber and includes several attributes such as the chat room name, members, and message history. In this example, there are two chat room objects representing Chat Room 1 and Chat Room 2.

Composite Structure

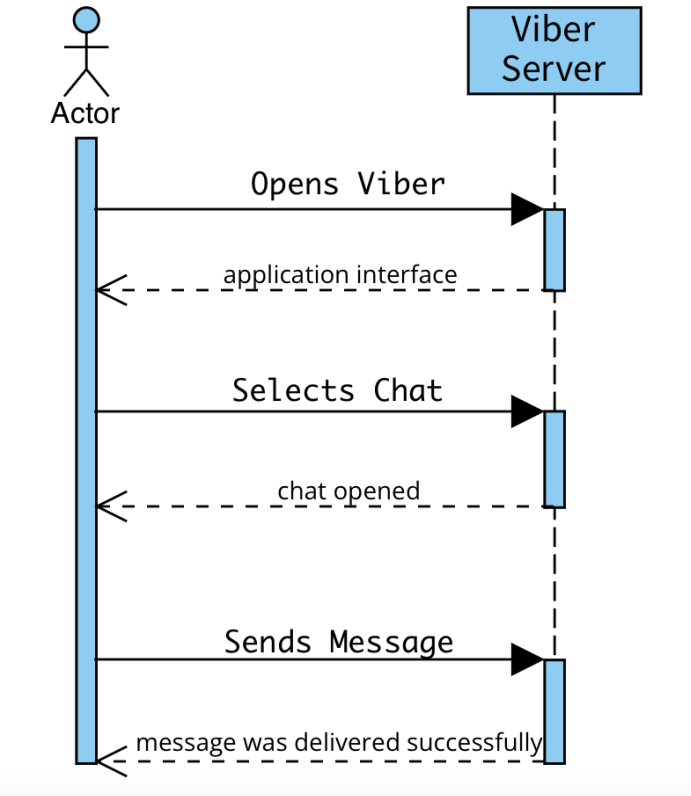


In this Composite Structure diagram, the Viber messaging service is depicted as a composition of several major components, including the User Interface, Messaging, Authentication, Media, Chat Management, External Services, and others.

Each component is represented as a composite object with its own internal structure, which can include subcomponents and interfaces. For example, the Messaging component includes subcomponents for messages and encryption, while the External Services component includes subcomponents for media hosting, payment processing, and API integration.

The components are connected through interfaces, which define the services provided by each component and the way that components interact with each other. For example, the Messaging component can send and receive messages through the Messaging interface, while the User Interface component can display messages on the user interface through the User Interface interface.

Sequence

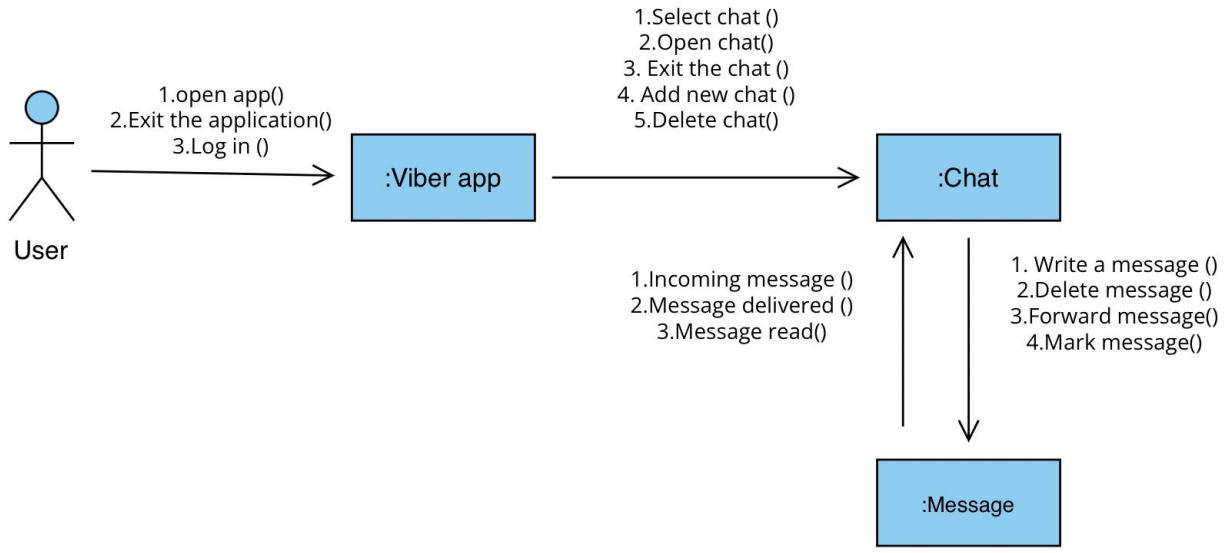


In this Sequence diagram, the steps for sending a message on Viber are depicted, showing the communication between the User and the Viber Server.

The User starts by opening the Viber app , selecting the chat they want to send a message to , typing the message , and then sending it . The Viber Server receives the message and processes it, sending it to the appropriate chat recipient. Once the message is received by the recipient, the Viber Server sends a confirmation message back to the user, indicating that the message was delivered successfully .

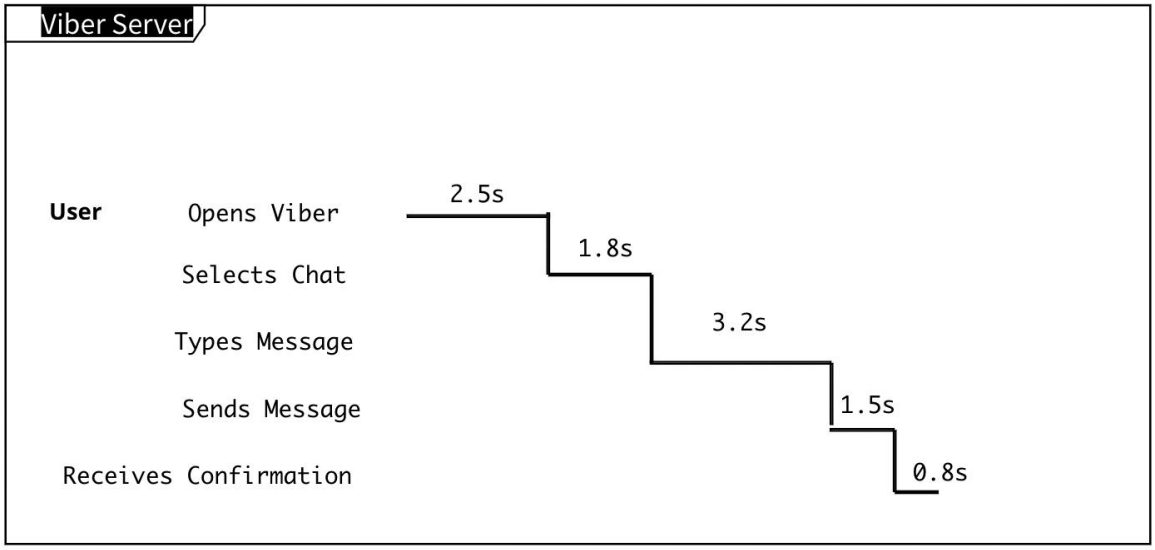
The Communication diagram shows the flow of communication between the User and the Viber Server during the process of sending a message, which can be used to understand the interactions and dependencies between the different components of the Viber messaging system.

Communication



A Communication diagram is a diagram that shows the interactions between elements at run-time in much the same manner as a Sequence diagram. However, Communication diagrams are used to visualize inter-object relationships, while Sequence diagrams are more effective at visualizing processing over time.

Timing

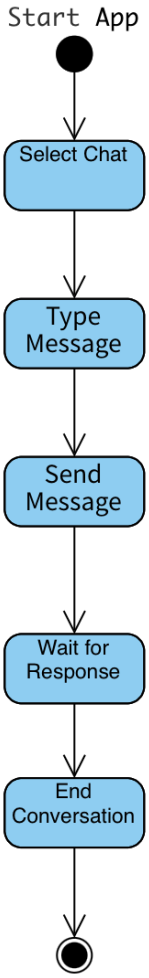


In this Timing diagram, the steps for sending a message on Viber are depicted, showing the duration of each step.

The User starts by opening the Viber app and this takes 2.5 seconds. The user then selects the chat they want to send a message to , which takes 1.8 seconds. After selecting the chat, the user types the message , which takes 3.2 seconds. Once the message is typed, the user sends it , which takes 1.5 seconds. The Viber Server receives the message and processes it, sending it to the appropriate chat recipient. Once the message is received by the recipient, the Viber Server sends a confirmation message back to the user , which takes 0.8 seconds.

The Timing diagram shows the duration of each step involved in sending a message on Viber, which can be used to identify potential bottlenecks or areas for improvement in the Viber messaging system.

Activity

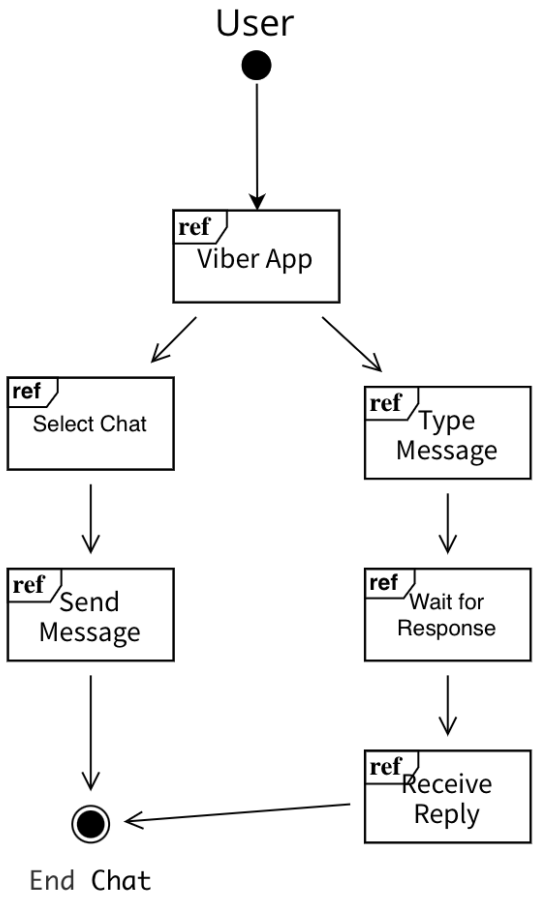


In this Activity diagram, the steps for sending a message on Viber are depicted, showing the flow of control between different activities.

The user starts by opening the Viber app and selecting the chat they want to send a message to. After selecting the chat, the user types the message and then sends it. The system then waits for a response, which can include a confirmation message or a reply from the recipient. Once the response is received, the conversation ends.

The Activity diagram shows the different activities involved in sending a message on Viber and the flow of control between them. It can be used to understand the different steps and decision points in the process, which can be helpful for designing and improving the Viber messaging system.

Interaction Overview

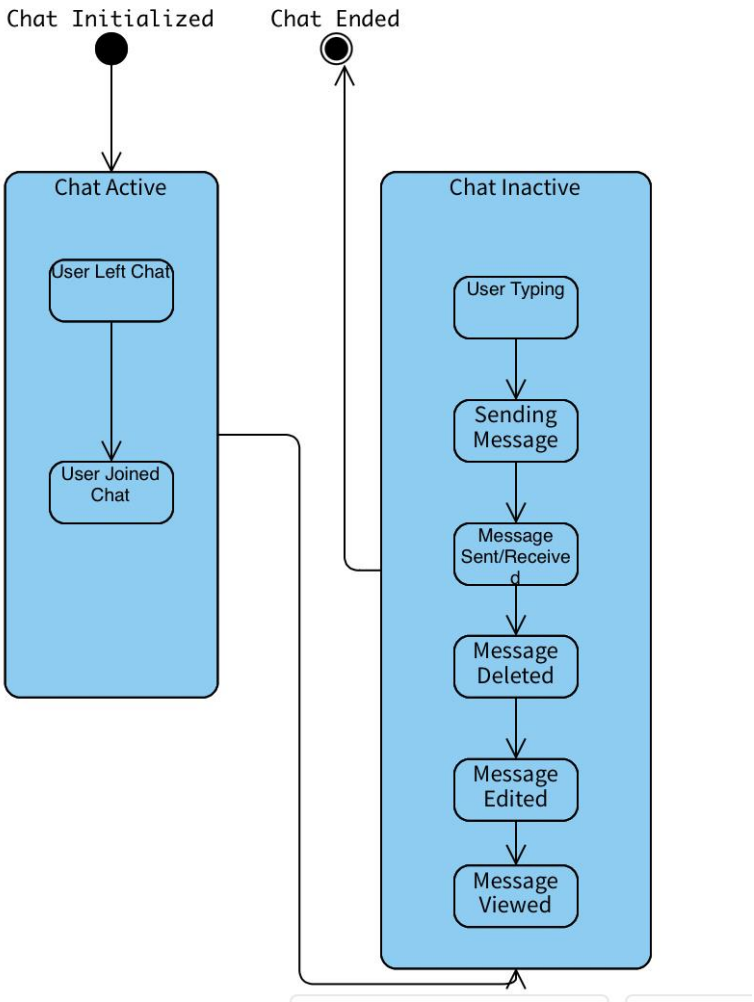


In this Interaction Overview diagram, the steps for sending a message on Viber are depicted, showing the high-level interactions between different components.

The user starts by using the Viber App and selecting the chat they want to send a message to. After selecting the chat, the user types the message and then sends it. The system then waits for a response, which can include a confirmation message or a reply from the recipient. Once the response is received, the conversation ends.

The Interaction Overview diagram shows the different interactions involved in sending a message on Viber and the flow of control between them. It can be used to understand the high-level interactions between different components of the Viber messaging system, which can be helpful for designing and improving the system.

State



In this State diagram, the different states of a Viber chat are depicted, including the different events that can trigger state transitions.

The chat starts in the "Chat Initialized" state and becomes "Chat Active" when a user joins the chat. The chat can transition to the "Chat Inactive" state if all users leave the chat. While the chat is active, different events can trigger state transitions, such as a user typing a message, sending a message, receiving a message, deleting a message, editing a message, or viewing a message.

The State diagram shows the different states and transitions involved in a Viber chat and can be used to understand the different events that can trigger state changes. It can be helpful for designing and improving the Viber messaging system.