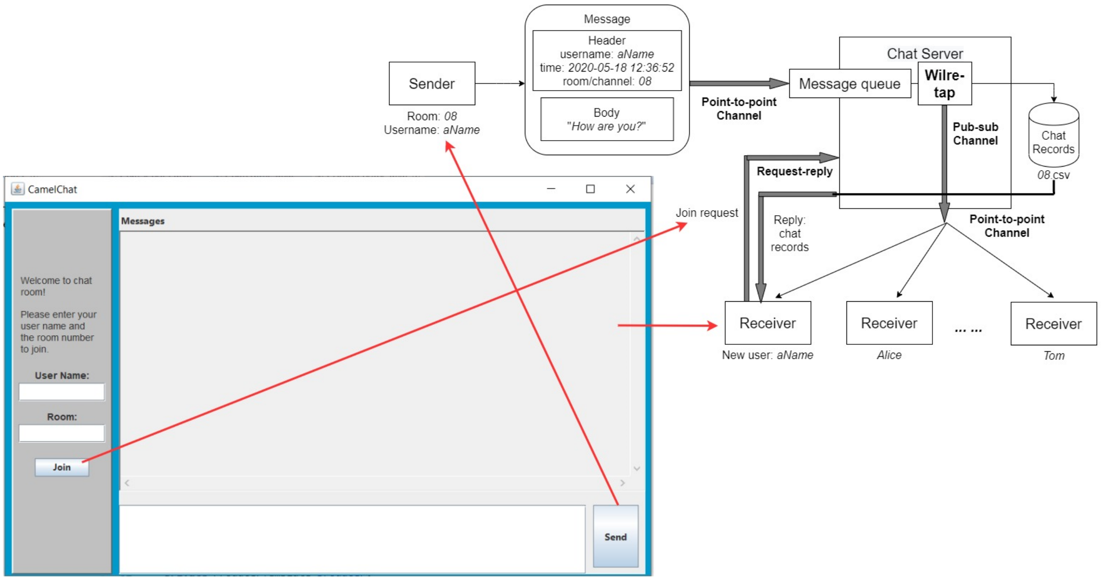

Welcome to chat room!

Please enter your user name and the room number to join.

Design patterns:

1. **Singleton:** Ensure only one instance of ChatSender for one GUI instance (ChatFrame).
2. **Mediator:** The chat server acts as the mediator of sender windows and receiver windows.
3. **Strategy:** Have two classes of RouteBuilder, each implementing its own route configuration.
4. **Layers:** In client side, split the app into 3 layers: GUI layer, function layer and communication layer.
5. **Façade:** Each layer has a Façade class that the upper layer talks to.

EIP patterns:

1. **Point-to-Point Channel:** Used to transmit messages from senders to the server, or to transmit history chat records from the server to newly joined receivers.
2. **Pub-sub Channel:** Used to transmit messages sent from senders to all the receivers in the room.
3. **Wire Tap:** Wire-tap the input message to local persistent csv storage
4. **Transform**: Transform JMS text message to csv byte message for persistent storage
5. **Content-based Router:** Route the message to different topics (chat rooms) based on the header content of input messages.
6. **Request-reply:** Every time a new user wants to join a chat room, it sends a request message to the server; the server replies the message with chat records of the room.
7. **Polling Consumer:** Whenever a join request comes, the server polls the chat records of a chat room from csv file endpoints.