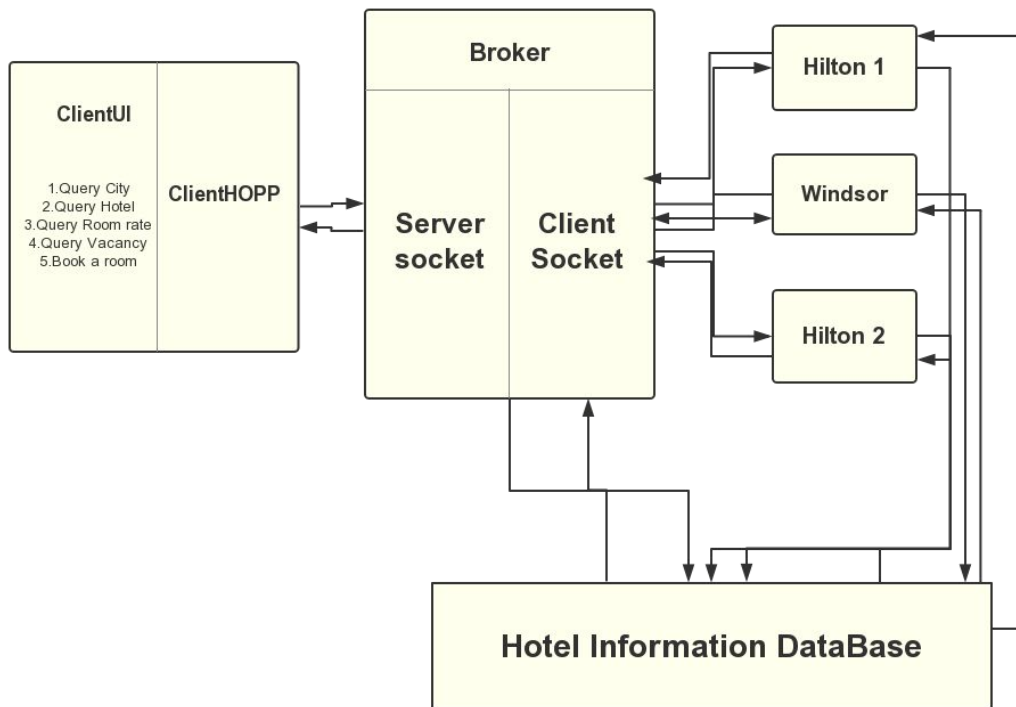


❖ The system architecture



❖ Description of each component:

This system contains 4 main components. Client part is designed to receive the user's command and show the results in the screen.

Broker part is designed to receive the message from client part and handle the different request in two ways. For example, the query requests are sent directly to the database to get the results but the booking request is sent to the hotel server then sent to the database. Broker also takes responsibility to return the results to the client part.

Hilton1, Windsor and Hilton2 are 3 different hotel servers to handle the booking request. Once they get the booking request, they send the insert sentence request to the database then return the booking condition to the broker.

Hotel information database is designed to carry out the SQL sentence and handle the messages from broker and hotel servers, then return the execution result to them.

❖ Specification.

Messages between client(s) and broker:

1. Query city: "CITY"+message+"\r\n";

The Client add the "CITY" prefix in the message and broker delete the prefix "CITY" in the message.

2. Query hotel: "HOTEL" + message + "\r\n";

The Client add the "HOTEL" prefix in the message and broker delete the prefix "HOTEL" in the message. The middle message is in the form of " : : :", so the client and broker will split the message by acknowledge the mark ":".

3. Query room rate: "ROOMRATE"+message+"\r\n";

The Client add the "ROOMRATE" prefix in the message and broker delete the prefix "ROOMRATE" in the message. The middle message is in the form of " : : :", so the client and broker will split the message by acknowledge the mark ":".

4. Query vacancy: "QUERY"+message+"\r\n";

The Client add the "QUERY" prefix in the message and broker delete the prefix "QUERY" in the message. The middle message is in the form of " : : :", so the client and broker will split the message by acknowledge the mark ":".

5. Exit: "EXIT"+"\r\n";

Once accept the "EXIT", the connection is broken.

6. Error: "ERROR"+"\r\n";

Once accept the "ERROR", the client show some hint message.

Message between client, broker and hotel servers:

Book a room: "BOOK" + message + "\r\n";

The Client add the "BOOK" prefix in the message and broker delete the prefix "BOOK" in the message. The middle message is in the form of " : : :", so the client and broker will split the message by acknowledge the mark ":". The first part and second part of the message will determine which hotel server to accept the message.

❖ UML class diagrams

