PART 1

1.

clientconsole and use args[1] instead of 0 that is being assigned

and just pass that new port

ClientConsole chat= new

ClientConsole(host, port);

protected synchronized void handleMessageFromClient  
(Object message, ConnectionToClient client)  
{  
String message1 = (String) message;  
if(message1.charAt(0) == '#') {  
String command[] = message1.split(" ");  
  
if(command[0].equals("#number")) {  
setChanged();  
notifyObservers(getNumberOfClients());  
}  
else if(command[0].equals("#all\_users")) {  
setChanged();  
notifyObservers(getClientConnections());  
}  
}  
//setChanged();  
//notifyObservers(message);  
}

add these methods in echoServer,

clientConnected(ConnectionToClient client) {

System.out.println("Client connected..");

}

clientDisconnected(ConnectionToClient client){

System.out.println("Client disconnected..");

}

2.

**#Functional requirment:**

**open GUI on telephone**

**make window with all content fields and pictures and fastens**

**test legitimacy of all client entered fields**

**confirm all fields are entered**

**show yield to screen about result of confirmation**

**3.Send information to data set**

**interface with information base**

**combine all content fields into a string**

**send string to information base**

**parse string to information base fields**

**send affirmation back to application**

**show on screen achievement liveliness**

**4.Open Check In App on transport**

**make window with text field and catch**

**Submit button is pushed**

**test legitimacy of client entered field**

**show yield to screen about result of confirmation**

**5.Phone application interfaces with client**

**Utilizing contact input, clients select a content field, the telephone OS gives information section techniques**

**Sources of info:**

**First Name - String, 30 characters max, not case delicate**

**Last Name - String, 30 characters max, not case delicate**

**Email - String, 30 characters max, not case delicate, requires precisely 1 "@" sign and at any rate 1 "." image**

**Ticket Type - Drop down menu with "Single Use," "1 Day Pass," "3 Day Pass,"Week Pass." Defaults to "Single Use"**

**Visa type - Drop down menu with "Visa," "Find," "Mastercard," Defaults to "Select sort."**

**Visa number - String, Exactly 16 numeric characters**

**Lapse date month - String, Exactly 2 numeric characters**

**Lapse date year - String, Exactly 4 numeric characters**

**6.Phone application interfaces with information base**

**When sending yield "information," information base returns 6 digit string "affirmation code"**

**Input:Using contact input, clients contact a picture button "Buy", "Information" yield is just sent once**

**Outputs:Data - String, all content fields entered as data sources connected together, isolated by semicolons**

**7.Database interfaces with telephone application**

**information base gets "information" string**

**Input:Data - String, all content fields entered on telephone connected together, isolated by semicolons**

**information base sends "affirmation code" string back to telephone application**

**Output:Confirmation Code - String, 6 digit numeric code allocated by information base as a key while accepting effective "information" contribution from Phone App.**

**8.Bus application interfaces with client**

**Utilizing console, clients enter "affirmation code" gave upon effective buy from telephone application**

**Info:**

**Affirmation Code - String, 6 digit numeric code appointed by information base as a key while accepting fruitful "information" contribution from Phone App.**

**9.Bus application interfaces with information base**

**When sending yield "code," information base will restore a Boolean demonstrating effective expulsion of given passage from information base**

**Information:**

**Achievement - Boolean, got back from information base, assuming "Valid" at that point show achievement message to client. Assuming "Bogus" show bomb message to client.**

**10.Database interfaces with Bus application**

**information base gets "code" string**

**Information:**

**Code - String, 6 digit numeric code, used to check if string as of now exists in information base as a key. At the point when discovered, that section in information base is eliminated, and yield "achievement" is returned.**

**information base sends "achievement" Boolean back to transport application**

**Yield:**

**Achievement - Boolean, "valid" if section determined by "code" was found and effectively eliminated. "Bogus" if passage determined by "code" was not found. Information base is unaltered if "Achievement" is bogus**

**#5 Non - functional requirment (Quality):::- -**

**After every update the framework will be investigated and contrasted and the framework necessities assigned by the customer. We will then attempt to break the program with experiments.**

**We will all be liable for quality affirmation. We will cross-investigate the framework execution with the System Requirements to check framework exactness.**

**All last reports will be surveyed for spelling blunders, broken connections, and whatever other issues that influence the nature of the record. Any blunders found in either the framework or supporting archives will be fixed at the time mistakes are found.**

**Security is utilized to guarantee Mastercard data as private and isn't put away anyplace. The main area any client information is noticeable is when clients enter in their data. Framework Administers are prepared on client security.**

**All information will be put away as Strings in the information base. The main estimations that will be performed is the hunt executed to check the "code" with all the passage Keys in the information base. The main different techniques that are run are the add and eliminate strategies for sections in the information base.**

**#3 Non - Functional Requirement (Platform):- -**

**Telephone App**

**Framework upholds clients with capacity to utilize essential cell phone abilities.**

**General comprehension of explicit telephone strategies for text info and catch determination**

**For clients with helpless vision**

**Android OS locally underpins zoom-in/zoom-out capacities**

**Transport App**

**Framework upholds clients with capacity to type 6 digit number**

**Information base Administration**

**Full admittance to physically add/eliminate/alter information fields and passages.**

**#2 Non – functional requirement (process):- -**

**Telephone Application**

**must work in demand for android based telephone OS Version 2.x. May take a shot at form 1.x yet isn't upheld.**

**telephone must approach web by means of either WiFi or Mobile Broadband**

**Transport Application**

**must work in demand for windows or macintosh based PC with console and mouse access.**

**PC must approach web through either WiFi or Mobile Broadband**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MODE | Experience | Accidents | Rush hours | Overtime |
| Short bus | Less than 15years | Very high | yes | low |
| Long bus | Less that 15years | Very high | yes | high |
| Rail | More than 10 years | Very low | No | low |
| Double decker | Less than 15years | Very low | no | high |