

- lines: BlockingQueue<String> closed: boolean backgroundReaderThread: Thread + NonBlockingBufferedReader(input: InputStream) + NonBlockingBufferedReader(reader: Reader) - SwingComponents + readLine(): String + close(): void username: String - passwort: String to: String - c: Client + UI + generateUI(): void Client log(log: String): void - hosts: HashMap<Host> printMessages(- username: String newMessages: List<Message>): void password: String + Client(username: String, password: String) + addHost(hostname, String, port: int): void + login(): String + getChatHistory(chatPartner: String): List<Message> + sendMessage(receiver: String, message: String): String + getUpdates(chatPartner: String): List<Message> + register(): String getDataPackageList(socketFromServer: Socket): List<DataPackage> sendDataPackage(socketToServer: Socket, CLI dataPackage: DataPackage): int c: Client closeSocket(socket: Socket): void reader: BufferedReader selectHost(): Host exit: boolean randomNumber(max: int): int + CLI() - waitMillis(millis: int): void modifySendMessageText(message: String): String + login: void modifyReceivedMessage(message: String): String printMessages(messageList: List<Message>): void

NonBlockingBufferedReader

Host

+ Host(hostname: String, port: int)

+ setHostname(hostname: String): void

- hostname: String

+ getPort(): int

+ getHostname(): String

+ setPort(port: int): void

port: int