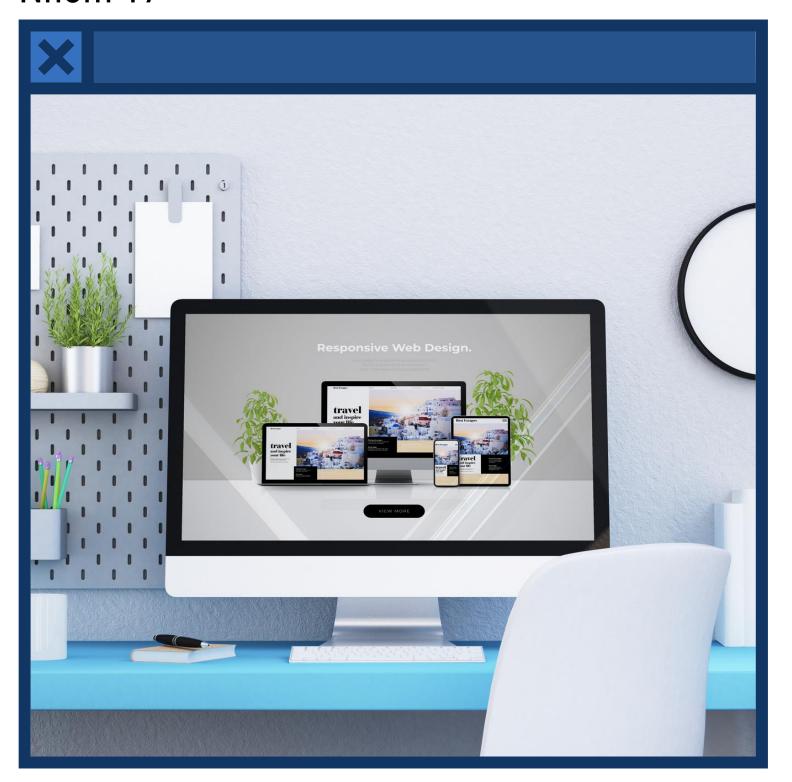
Úng dụng Nhán tin



Nhóm 17



Những phần chính

01 KHẢO SÁT, ĐẶC TẢ YÊU CẦU BÀI TOÁN

02 PHÂN TÍCH THIẾT KẾ BÀI TOÁN

03 CÔNG NGHỆ VÀ THUẬT TOÁN SỬ DỤNG

04 XÂY DỰNG CHƯƠNG TRÌNH MINH HOẠ

Giới thiệu thành viên



Hồ Xuân Thái

Trưởng nhóm, Code chính

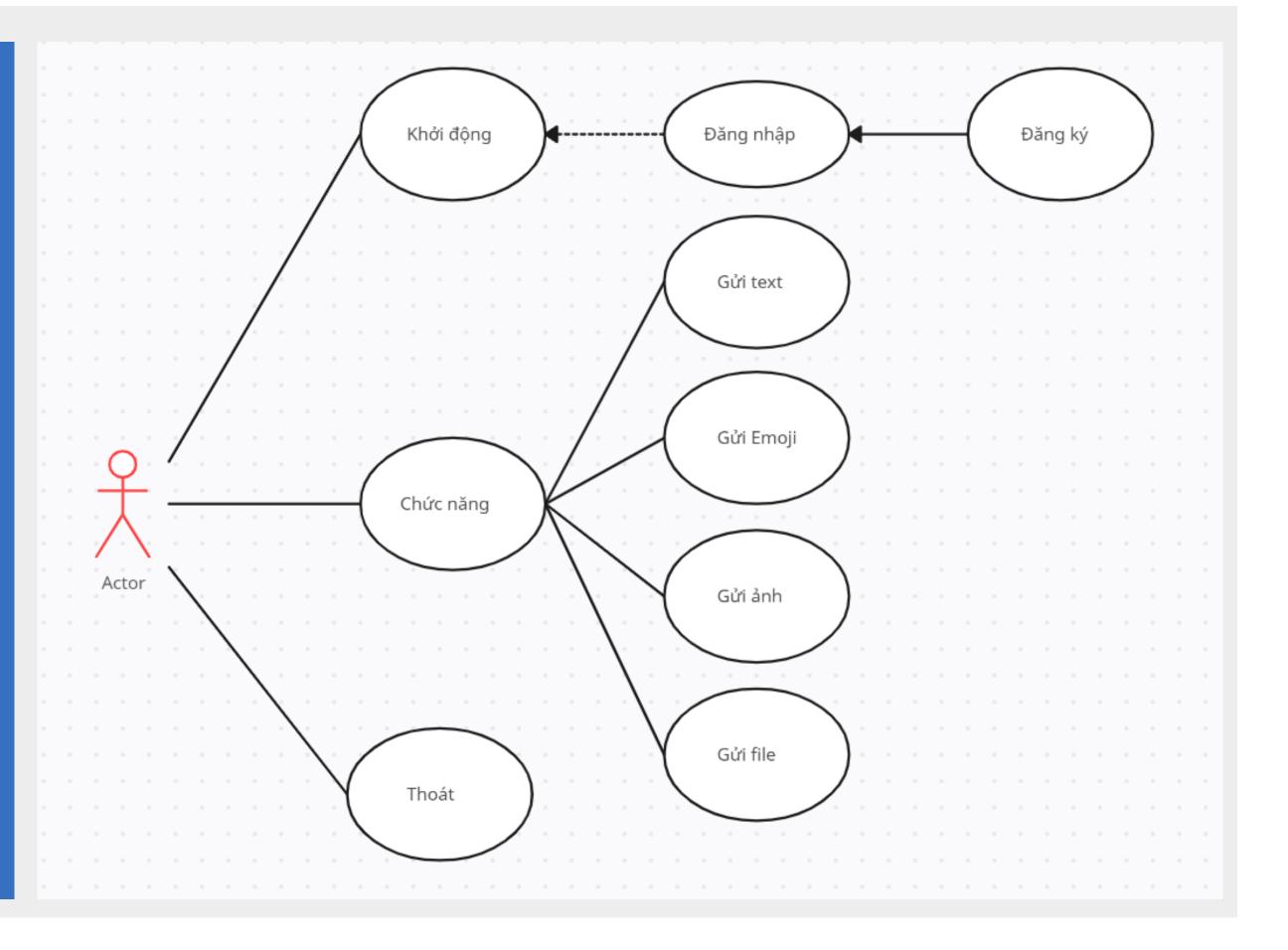


Lê Việt Hoàng

Hỗ trợ, code phụ, làm slide

BIẾU ĐỒ USE CASE

BIẾU ĐÔ PHÂN RÃ



PHÂN TÍCH THIẾT KẾ BÀI TOÁN

```
+-----+
| Tables_in_chatapp |
+-----+
| files |
| user |
| user_account |
```

+		+	
UserID	UserName	Password	
+		+	
16	user1	1	
17	user2	22	
18	user3	333	
19	user4	4444	
20	user5	55555	
21	user5.1	515151	
22	user6	666666	
23	user0	0000	
+		+	

<u>USER</u>

+ UserID	 UserName 	Gender	Image	ImageString	Status
16	user1		NULL		1
17	user2		NULL		1
18	user3		NULL		1
19	user4		NULL		1
20	user5		NULL		1
21	user5.1		NULL		1
22	user6		NULL		1
23	user0		NULL		1
+					+

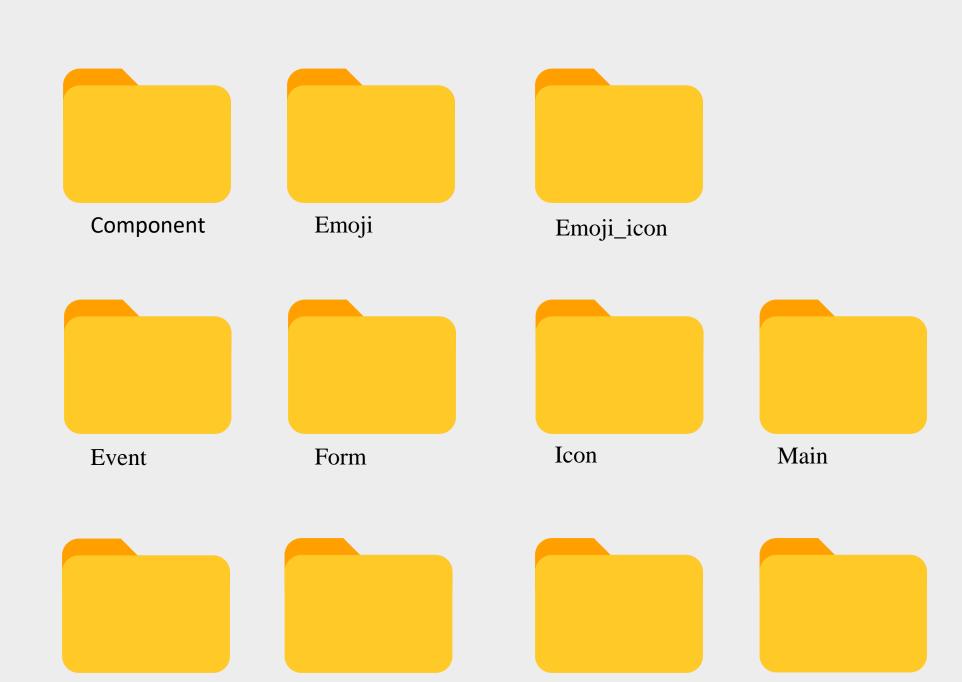
USER ACCOUNT

++ FileID	FileExtension	BlurHash	Status
52 53 54 55 56 57 58 59	.png .jpg .jpg .jpg .jpg .png .jpg .jpg .jpg	L4SY{q9F~q%M?bofWBj[-;WBxuof L7ABneE.8xyC~ptmNKRjIC?^IURP L7ABneE.8xyC~ptmNKRjIC?^IURP LC9j} M_4T%MITWAjYWB00xv?wM L5Aw@:tmEn~okCxtxsE.ng-gIV L6S?DVHtIEAV?wWqRjnkNgbbaer@ LC9j} M_4T%MITWAjYWB00xv?wM LC9j} M_4T%MITWAjYWB00xv?wM L5Aw@:tmEn~okCxtxsE.ng-gIV	1 1 1 1 1 1 1

<u>FILE</u>

Service

CÁC PACKAGE



Swing

Model

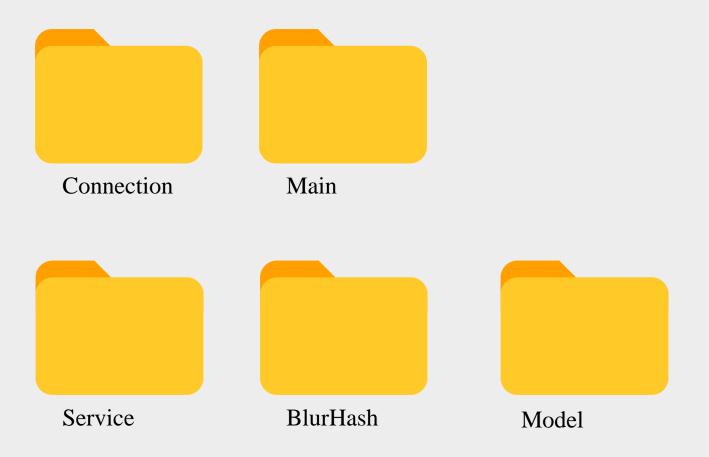
BlurHash



CLIENT

CÁC PACKAGE

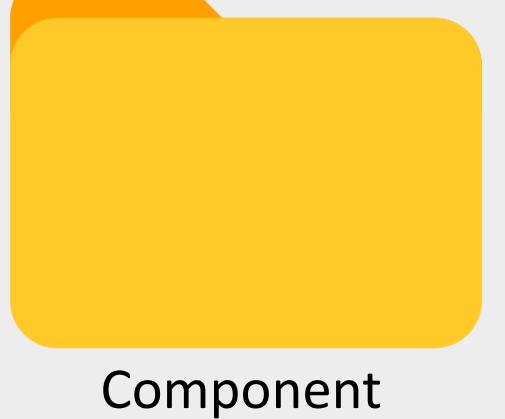


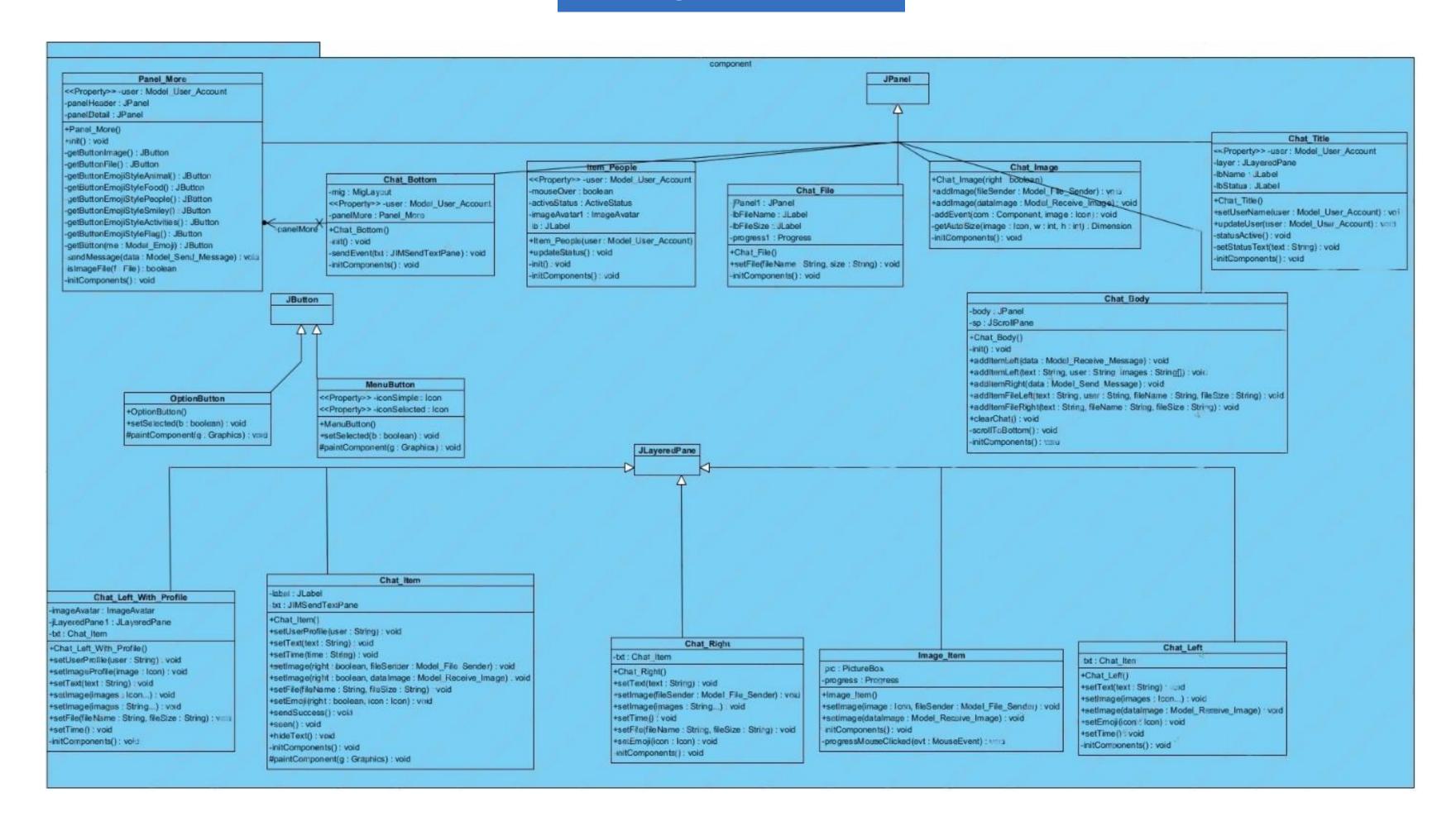


CÁC PACKAGE



CLIENT





emoji

Model_Emoji

<<Pre><<Pre>roperty>> -id : int

<Property>> -icon : Icon

+Model_Emoji()

+Model_Emoji(id : int, icon : Icon)

+getID(): int +setID(): void

+getIcon() : Icon

+setIcon(): void

Emoji

Property>> -instance : Emoji

-Emoji()

+getStyleAnimal(): List<Model_Emoji>

+getStyleFood(): List<Model_Emoji>

+getStylePeople(): List<Model_Emoji>

+getStyleSmiley(): List<Model_Emoji>

+aetStyleActivities(): List<Model_Emoji>

+getStyleFlag(): List<Model_Emoji>

+getEmoji(id : int) : Model_Emoji

а

event

+register(data: Model_Register, mes: EventMessage): void

+goRegister() : void +goLogin() : void

a <<Interface>>
EventChat

+sendMessage(data: Model_Send_Message): void

+receiveMessage(data: Model_Receive_Message): void

a <<Interface>>
EventMenuLeft

+newUser(users : List<Model_User_Account>) : void

+userConnect(userID : int) : void +userDisconnect(userID : int) : void

<<Interface>>
EventMain

+showLoading(show : boolean) : void

+initChat(): void

+selectUser(user : Model_User_Account) : void +updateUser(user : Model User Account) : void

E

PublicEvent

<< Property>> -instance : PublicEvent

<< Property>> -eventImageView : EventImageView

<<Pre><<Pre><<Pre><<Pre><<Pre><<Pre><<Pre><<Pre><</pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre><<pre></p

<<Pre><<Pre>roperty>> -eventMenuLeft : EventMenuLeft

-PublicEvent()

+addEventMenuLeft(event : EventMenuLeft) : void +addEventImageView(event : EventImageView) : void

+addEventChat(event : EventChat) : void +addEventLogin(event : EventLogin) : void +addEventMain(event : EventMain) : void a <<Interface>>

EventFileSender

+onSending(percentage : double) : void

+onStartSending(): void

+onFinish(): void

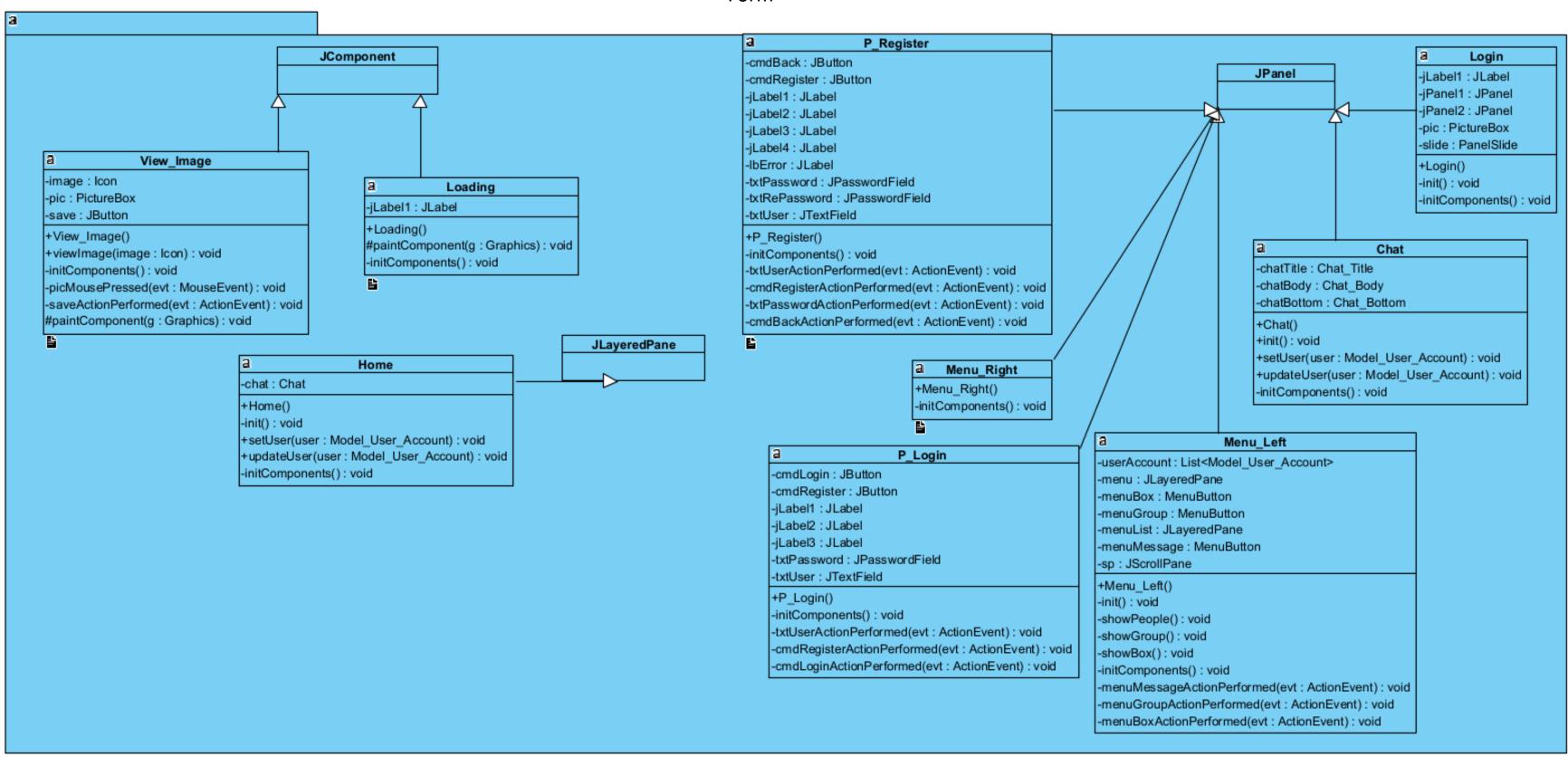
<<Interface>>

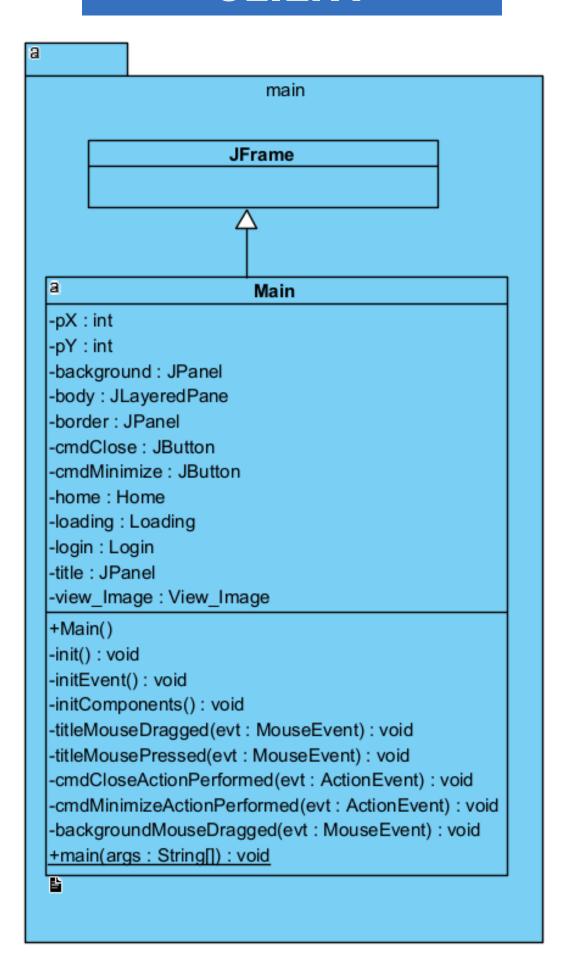
EventFileReceiver

+onReceiving(percentage : double) : void

+onStartReceiving(): void +onFinish(file: File): void

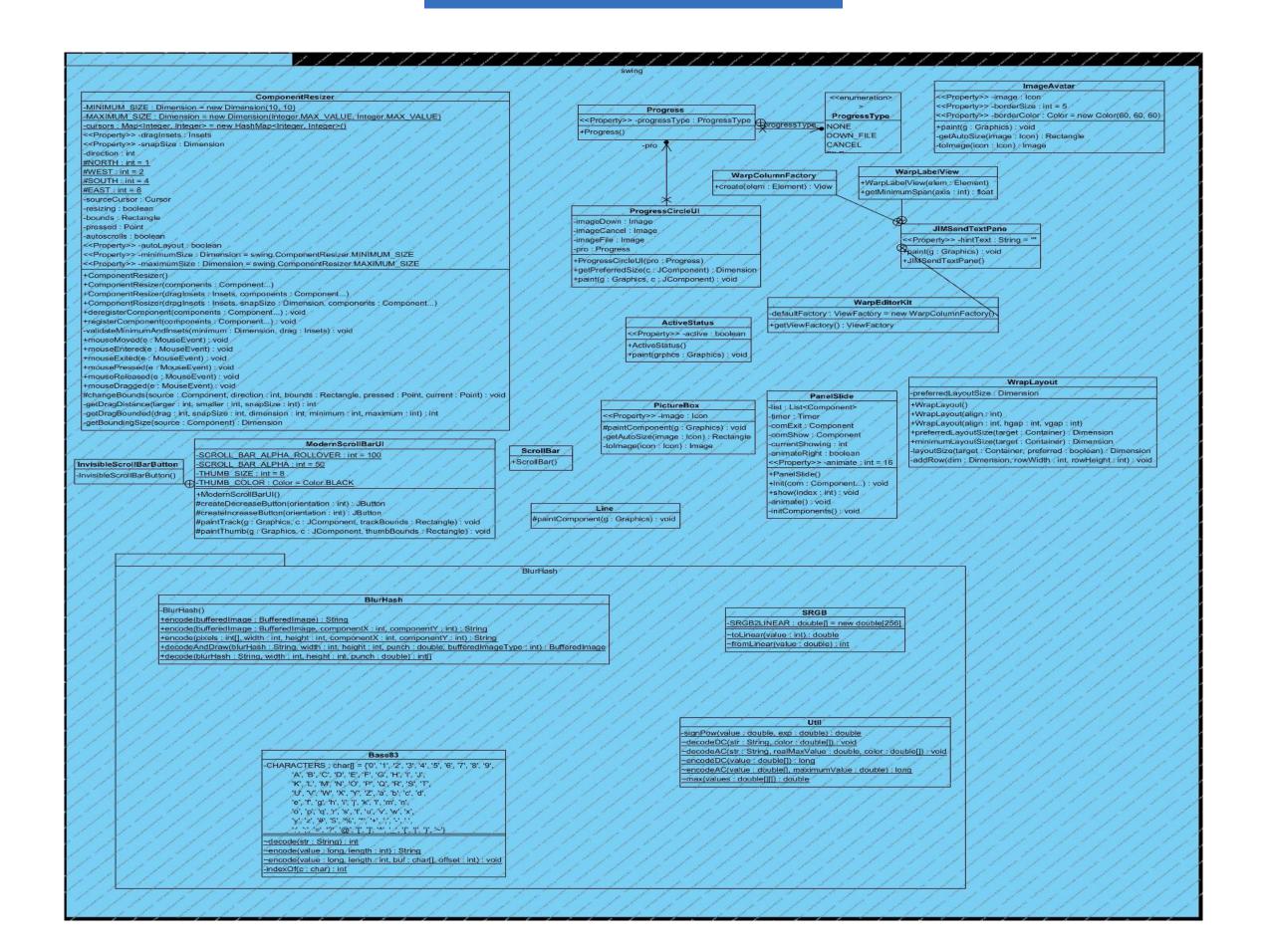
Form





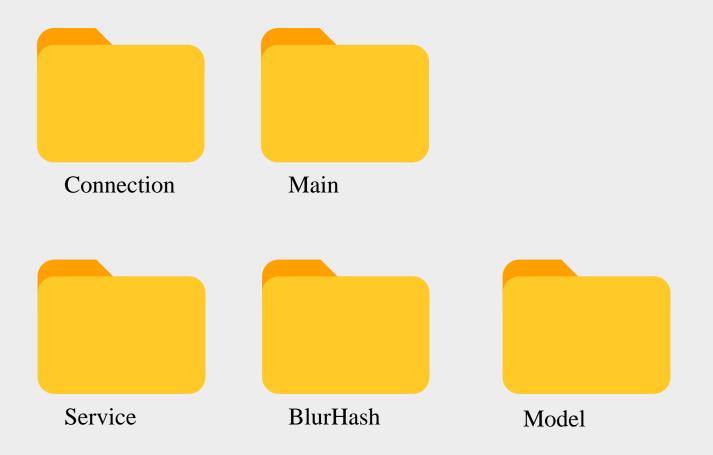
```
Model Send Message
                                                                                                                                                                  model
                                                                           << Property>> -fromUserID: int
                        Model Package Sender
                                                                           << Property>> -toUserID : int
     << Property>> -fileID: int
                                                                           << Property>> -text : String
    << Property>> -data : byte[]
                                                                           << Property>> -messageType : int
    << Property>> -finish : boolean
                                                                           << Property>> -file : Model_File_Sender
                                                                           +Model_Send_Message(fromUserID : int, toUserID : int, text : String, messageType : int)
    +Model_Package_Sender()
    +Model_Package_Sender(fileID : int, data : byte[], finish : boolean)
                                                                           +Model_Send_Message()
    +toJSONObject(): JSONObject
                                                                           +toJsonObject(): JSONObject
    Model_User_Account
                      Model Login
                                                              << Property>> -userName : String
    << Property>> -userName : String
                                                            << Property>> -userID : int
    << Property>> -password : String
                                                             model y>> -gender: String
                                                             << Property>> -image : String
    +Model_Login(userName : String, password : String)
                                                             << Property>> -status : boolean
    +Model_Login()
                                                             +Model User_Account(json : Object)
    +toJSONObject(): JSONObject
                                                             +Model User Account(userName: String, userID: int, gender: String, image: String, status: boolean)
                                                                                                               Model File Sender
                                                                                  << Property>> -fileID : int
                              Model File Receiver
                                                                                  << Property>> -fileExtension : String
    << Property>> -fileID : int
                                                                                  <<Pre><<Pre>roperty>> -file : File
    << Property>> -file : File
                                                                                  << Property>> -fileSize : long
    << Property>> -fileSize : long
                                                                                 << Property>> -accessFile : RandomAccessFile
    << Property>> -fileExtension : String
                                                                                 << Property>> -socket : Socket
    << Property>> -socket : Socket
                                                                                  -event : EventFile Sender
    << Property>> -accessFile : RandomAccessFile
                                                                                  -message: Model Send Message
    << Property>> -event : EventFileReceiver
                                                                                  +Model File Sender()
    -PATH_FILE : String = "client_data/"
                                                                                  +Model_File_Sender(message: Model_Send_Message, file: File, socket: Socket)
    +Model_File_Receiver()
                                                                                  +readFile(): byte[]
    +Model_File_Receiver(fileID: int, socket: Socket, event: EventFileReceiver)
                                                                                  +initSend(): void
    +initReceive(): void
                                                                                  +startSend(fileID : int) : void
    +startSaveFile(): void
                                                                                                                                                                                   Model_Request_File
                                                                                  +sendingFile(): void
    +writeByte(data:byte[]):long
                                                                                                                                                                     << Property>> -fileID : int
                                                                                  +getPercentage(): double
    +getPercentage(): double
                                                                                                                                                                     << Property>> -currentLength : long
                                                                                  -getExtension(fileName : String) : String
    +close(): void
                                                                                                                                                                     +Model Request_File()
                                                                                 +getModel_Send_Message(): Model_Send_Message
                                                                                  +setModel Send Message (message: Model Send Message): void
                                                                                                                                                                     +Model Request File(fileID : int, currentLength : long)
                            Model_Receive_Message
                                                                                                                                                                     +toJsonObject(): JSONObject
                                                                                  +addEvent(event : EventFileSender) : void
    << Property>> -messageType : int
     <<Pre><<Pre>roperty>> -fromUserID : int
     << Property>> -text : String
    << Property>> -dataImage : Model_Receive_Image
    +Model_Receive_Message(fromUserID : int, text : String, messageType : int)
    +Model_Receive_Message(json : Object)
                                                                                                                                       Model Receive Image
     +toJsonObject(): JSONObject
                                                                                                                << Property>> -fileID : int
                                                                                                                << Property>> -image : String
                  Model Register
                                                                                                                 << Property>> -height : int
                                                                          Model_Message
<< Property>> -userName : String
                                                                                                                 << Property>> -width : int
<< Property>> -password : String
                                                         << Property>> -action : boolean
                                                                                                                +Model_Receive_Image(json : Object)
                                                         << Property>> -message : String
                                                                                                                +toJsonOject(): JSONObject
+Model_Register()
                                                         +Model Message(action : boolean, message : String)
                                                                                                                 +Model Receive Image()
+Model_Register(userName : String, password : String)
+toJSonObject(): JSONObject
                                                          +Model Message()
                                                                                                                 +Model Receive Image(fileID: int, image: String, height: int, width: int)
```

service Service << Property>> -client : Socket -PORT : int = 9999 -IP : String = "localhost" <<Pre><<Pre>roperty>> -user : Model_User_Account -fileSender : List<Model_File_Sender> -fileReceiver : List<Model_File_Receiver> << Property>> -instance : Service +Service() +startServer(): void +addFile(file: File, message: Model_Send_Message): Model_File_Sender +fileSendFinish(data: Model_File_Sender): void +fileReceiveFinish(data: Model_File_Receiver): void +addFileReceiver(fileID : int, event : EventFileReceiver) : void



CÁC PACKAGE

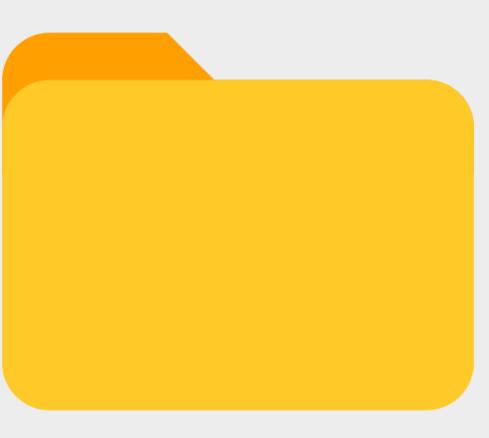




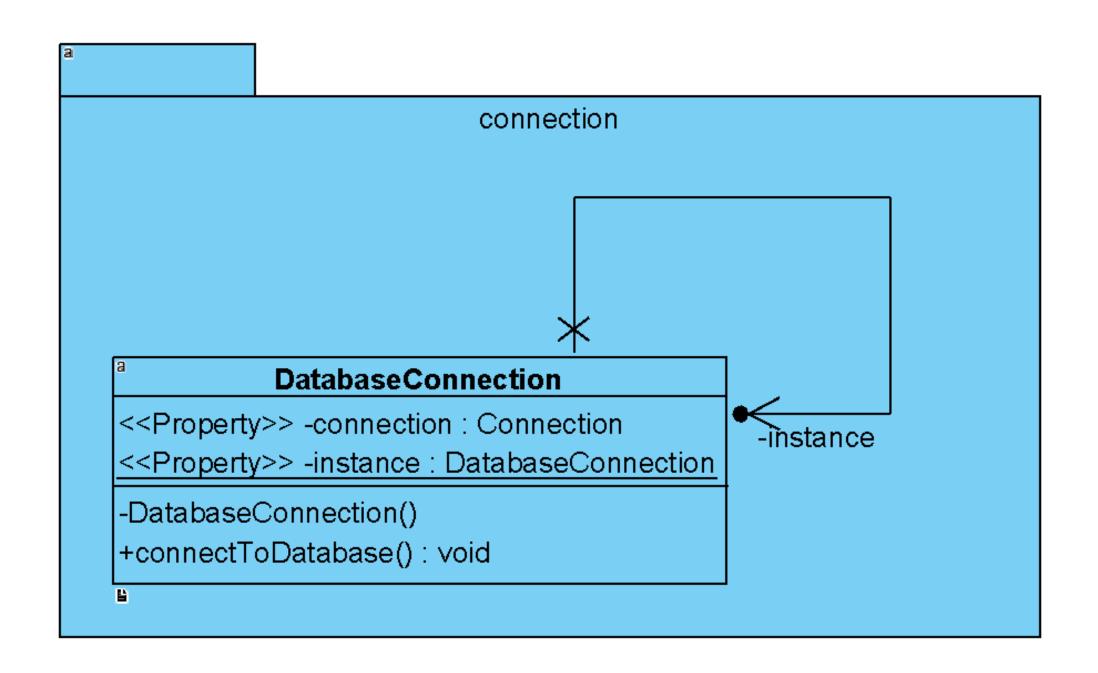
CÁC PACKAGE



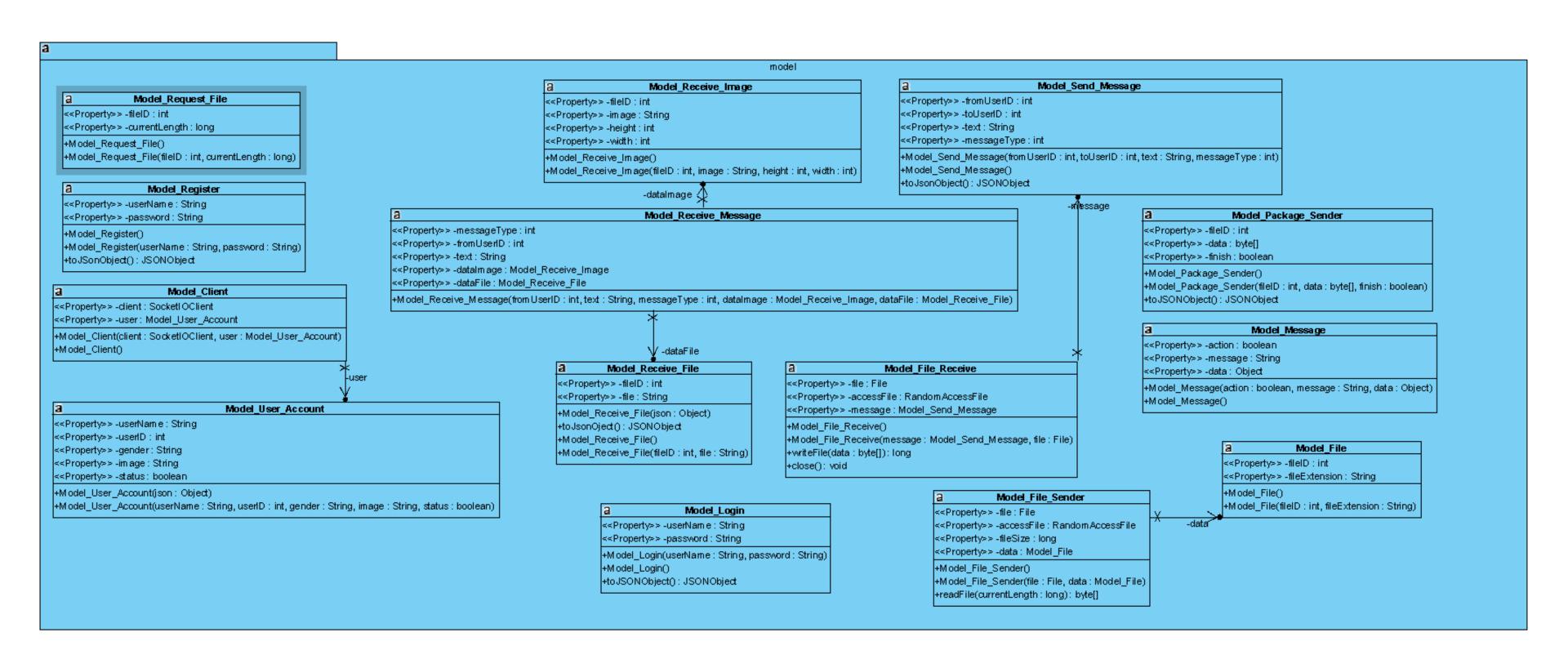
SERVER

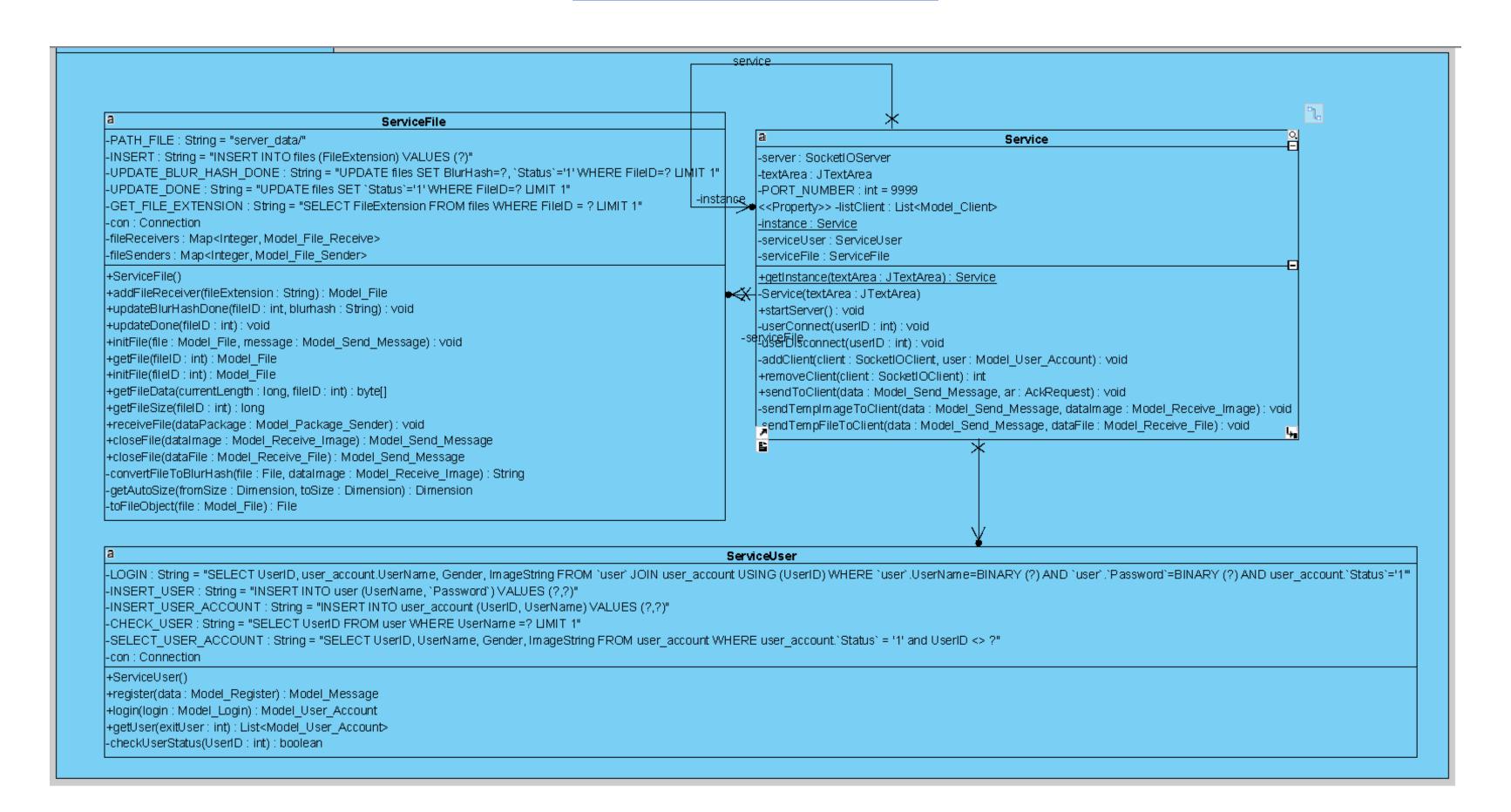


connection



main Main -jLabel1 : JLabel -jScrollPane1 : JScrollPane -txt : JTextArea +Main() -initComponents(): void -formWindowOpened(evt : WindowEvent) : void +main(args : String∏) : void





```
swing
                                                                 BlurHash
                                                          BlurHash
-BlurHash()
+encode(bufferedImage : BufferedImage) : String
+encode(bufferedImage : BufferedImage, componentX : int, componentY : int) : String
+encode(pixels:int∏, width:int, height:int, componentX:int, componentY:int): String
+decodeAndDraw(blurHash: String, width: int, height: int, punch: double, bufferedImageType: int): BufferedImage
+decode(blurHash : String, width : int, height : int, punch : double) : int∏
                                                                                                               Util
                              Base83
-CHARACTERS : char[] = {'0', '1', '2', '3', '4', '5', '6', '7', '8', '9',
                                                                          -signPow(value : double, exp : double) : double
        'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J',
                                                                           ~decodeDC(str : String, color : double∏) : void
                                                                           ~decodeAC(str : String, realMaxValue : double, color : double∏) : void
        'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T',
       'U', 'V', 'W', 'X', 'Y', 'Z', 'a', 'b', 'c', 'd',
                                                                           ~encodeDC(value : double∏) : long
       'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n',
                                                                           ~encodeAC(value : double∏, maximumValue : double) : long
                                                                           ~max(values : double∏) : double
       'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x',
        'y', 'z', '#, '$', '%', '*', '+', ',', '-', '.',
        <u>':', ';', '=', '?', '@', '[', ']', '^', ' ', ' ', '{', '|', '}', '~\}</u>
                                                                                                  SRGB
~decode(str : String) : int
                                                                             -SRGB2LINEAR : double[] = new double[256]
~encode(value : long, length : int) : String
                                                                             ~toLinear(value : int) : double
~encode(value : long, length : int, buf : char∏, offset : int) : void
                                                                             ~fromLinear(value : double) : int
-indexOf(c : char) : int
```

CÔNG NGHỆ VÀ THUẬT TOÁN SỬ DỤNG

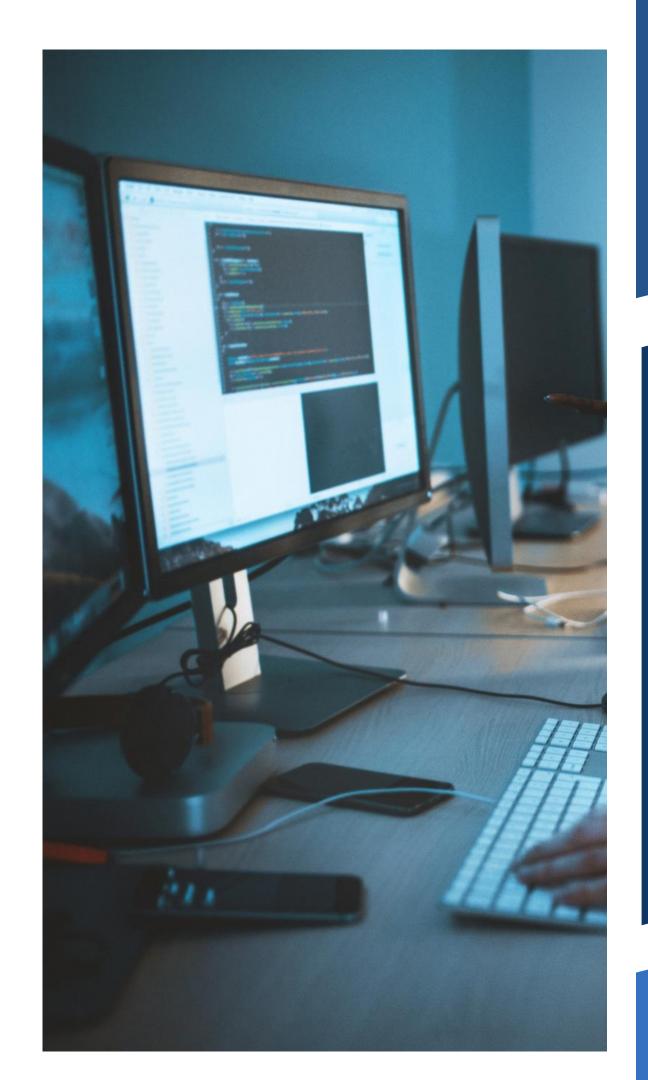
UI: Swing

Service

JSON

Socket Programing: netty

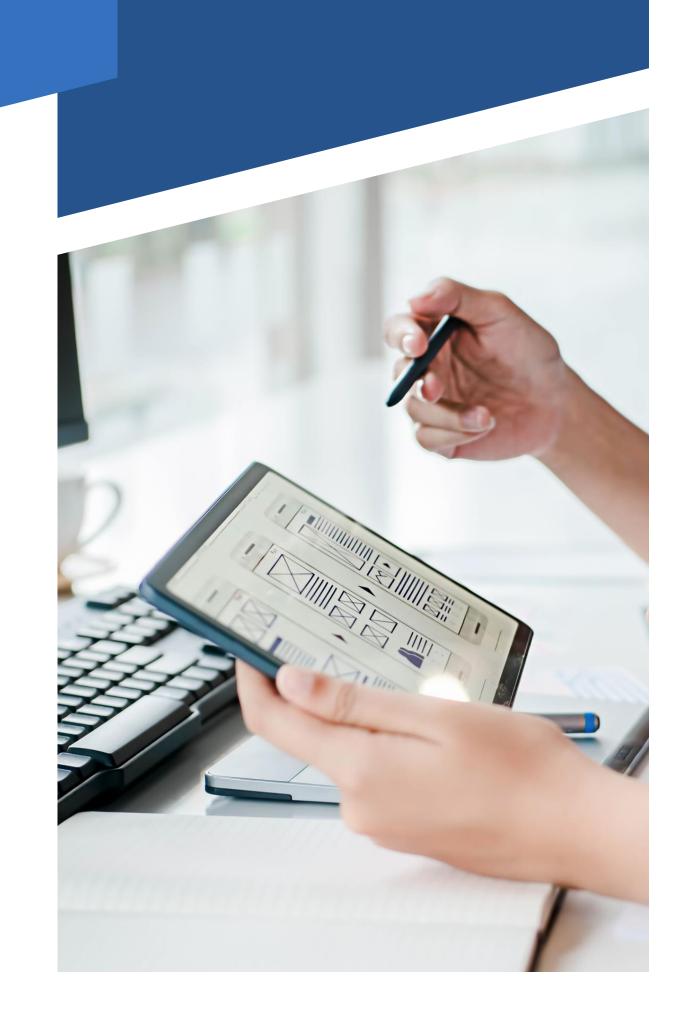
SQL Database



Giao diện và demo



Messenger





Kết luận và hướng phát triển

1.Ưu điểm:

Đáp ứng, hoàn thành đầy đủ các yêu cầu của đề tài, các chức năng cơ bản của 1 ứng dụng nhắn tin

Ứng dụng chạy mượt mà và không gặp các lỗi như crash ứng dụng hay lỗi đồ hoạ.

2. Nhược điểm:

Đồ hoạ chưa thực sự bắt mắt, chưa có tính năng gọi và lưu.

3. Hướng phát triển:

Thêm các thao tác cập nhật ảnh, giới tính cho từng user.

Phát triển thao tác có thể lưu trữ được các tin nhắn vào cơ sở dữ liệu.

Cải thiện giao diện đồ hoạ thân thiện và dễ dàng tiếp cận hơn.

Sửa các lỗi sẵn có.

THANK YOU