Class Messenger extends Component{

State = {

Authenticated: null,

userID: null

usersChatRoomID: [],

currentChatRoom: {},

currentChatRoomID: null

currentChatRoomName: “blaine”,

username: this.props.username

}

ComponentDidMount(){

setAuthentication()

checkAuthentication()

setusersChatRoomsID()

}

setAuthentication(){

if this.props.authenticated ? this.state.authenticated = this.props.authenticated : this.state.authenticated = false

if this.props.uID ? this.state.userID = this.props.uID : this.state.userID = null

if this.props.uname ? this.state.username = this.props.uname

}

checkAuthentication(){

if this.state.authenticated ? return : redirect to authentication Component

}

setUsersChatRoomID(){

/\*send SQL command to get different chatrooms for each different user\*/

sql = select \* from usersChatRooms where userID = this.state.userID

set this.state.usersChatRoomID = the sql results

}

/\*setCurrentChatRoom( chatRoomID ){

Sql Select \* from chatRoomID

this.state.currentChatRoom = to sql results

this.state.currentChatRoomID = chatRoomID

setCurrentChatRoomName()

}

setCurrentChatRoomName(){

sql = select \* from chatRoomsUsers where chatRoomID === this.state.currentChatRoomID

delete this.state.userID from the returned array

getUsernameByID(sql result)

this.state.currentChatRoomName = sql result

}

getUsernameByID( userID ){

send SQL command to get username from the userID passed.

Sql = select userName from users where UserID === userID

return the sql result

}

newMessage(message){

if message is > 200 characters && !null ?

/\*!!!!!escape data for MySQL and HTML!!!!!\*/

Insert {message} into this.state.currentChatRoomID where userID===this.state.userID

}

} /\* end of class\*/