UML class for Message

Siqing Xu, Chuanrun Zhang, Christine Chen

Message

- SenderID: String
- ReceiverID: String
- Content: String
- Time: String
- Creation: Schedule

+ Message(senderID: String, receiverID: String, content: String)
+ getSenderID(): Sting
+ getReceiverID(): String
+ getCreation(): Schedule
+ getContent(): String
+ getTime(): String
+ setCreation(creation: Schedule): void
+ setSenderID(senderID: String): void
+ setReceiverID(receiverID: String): void
+ setContent(content: String): void
+ toString(): String

«interface»
Serializable

- writeObject(out: java.io.ObjectOutputStream): void
- readObject(in: java.io.ObjectInputStream): void
- readObjectNoData(): void

MessagePublisher

+ listeners: HashMap<String, ArrayList<MessageListener>>

+ MessagePublisher(operations: ArrayList<String>)
+ subscribe(eventType: String, listener: MessageListener): void
+ notify(actionType: String, message: Message): void

MessageSender

+ events: MessagePublisher

+ MessageSender(operations: ArrayList<String>)
+ sendMsg(actionType: String, message: Message): void

SendMsgListener

- mm: MessageManager

+ update(actionType:String, message: Message): void

<<interface>>
MessageListener

+ update(actionType:String,message: Message): void

MessageUI

- facade: Facade
- sc: Scanner
- userId: String
- sm: ScheduleManager
- mm: MessageManager
- sui: ScheduleUI

+ MessageUI(facade: Facade, scanner: Scanner, userId: String)-
- attachSchedule(msg: Message): void
+ runByRegular(): void
- sendByReg(): void
+ runByAdmin(): void
- replyByAdmin(): void
- replyByReg(): void
- sendByAdmintoSome(): void
- deleteMsgByAdmin(): void
- deleteMsgByReg(userId: String): void
- linkSchedule(String: userid): void
- ChooseSchedule(): void
+ saveToFrw(): void

MessageManager

+ um: UserManager
- chathistory: HashMap<String, ArrayList<Message>>
- commonInbox: ArrayList<Message>

+ MessageManager(um: UserManager)
+ createMessage(senderID: String, receiverID: String, content: String): Message
+ setCommonInbox(commonInbox: ArrayList<Message>): void
+ setChatHistory(chathistory: HashMap<String, ArrayList<Message>>): void
+ sendToAdmin(message: Message): void
+ admintoSomeUsers(message: Message): void
+ sendtoAllUser(message: Message): void
+ sendMessage(message: Message): void
+ makeMessageCopy(message: Message): Message
+ deleteMessage(message: Message, ownerid: String): void
+ deleteCommonInbox(message: Message): void
+ getChatHistory(): HashMap<String, ArrayList<Message>>
+ getCommonInbox(): ArrayList<Message>
+ commonInboxToString(): void
+ UserChatHistoryString(userid: String): void
+ AllChatHistoryString(): void
+ getUserEmail(id: String):String