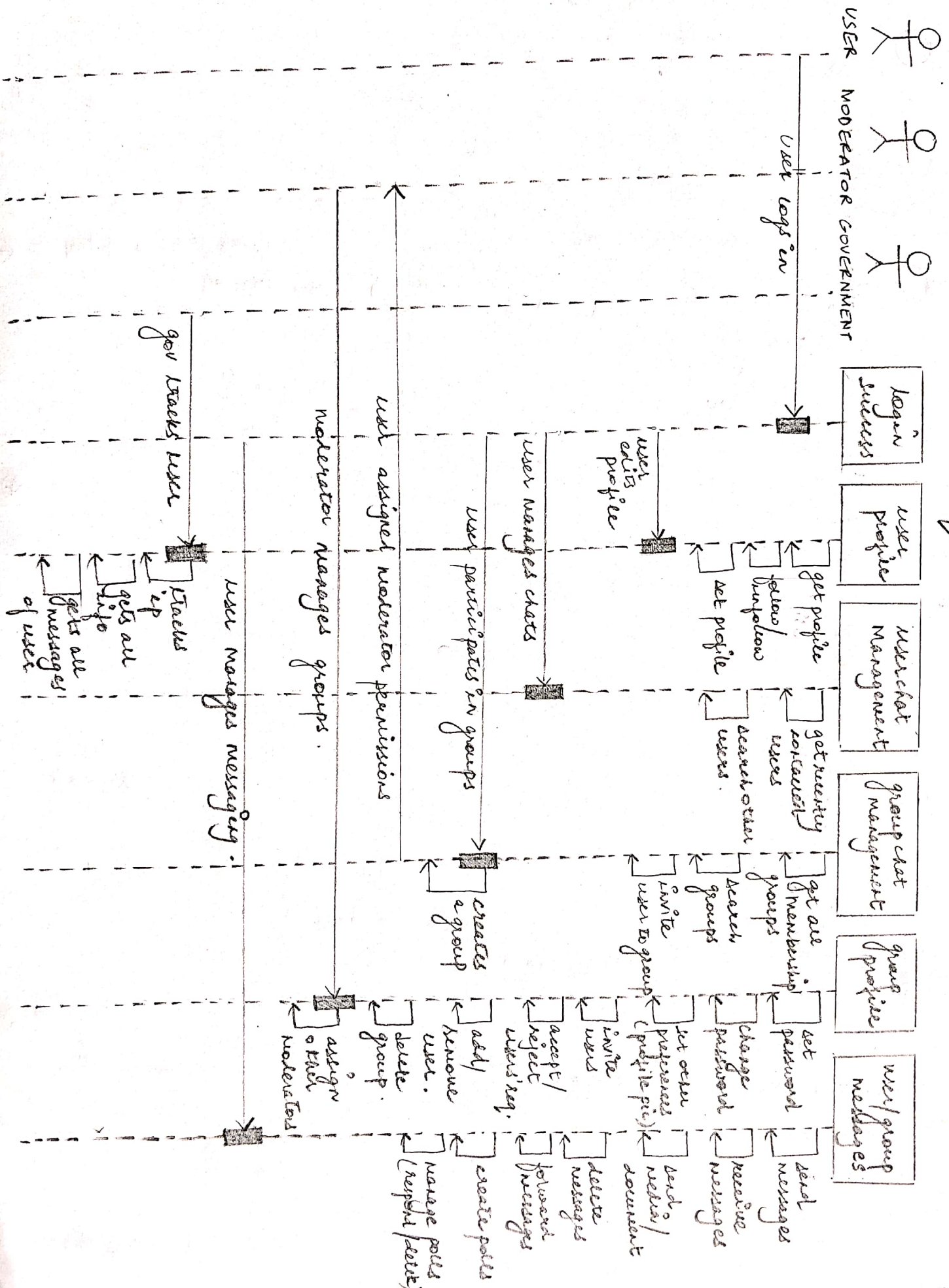
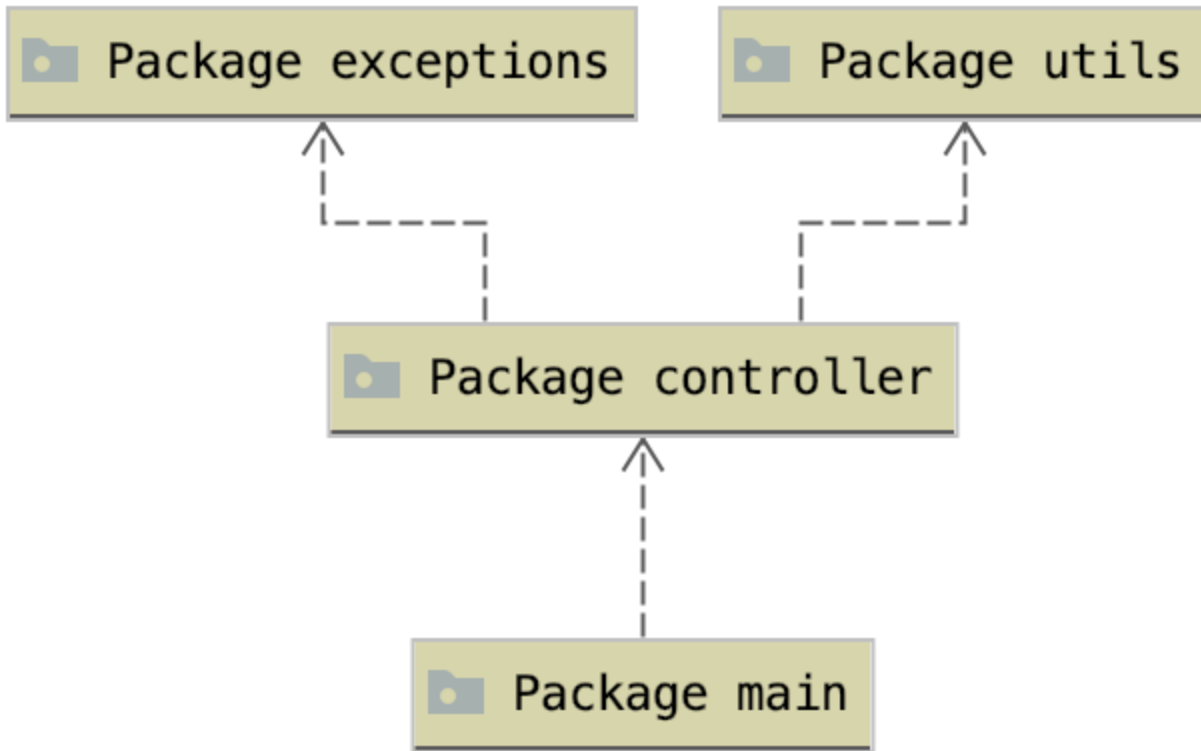


Sequence





Package websockets

Package websocket

Package service

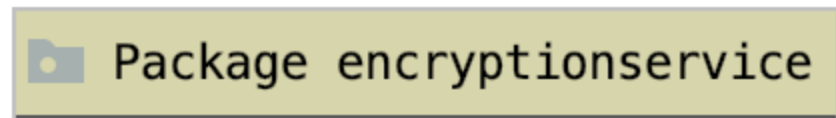
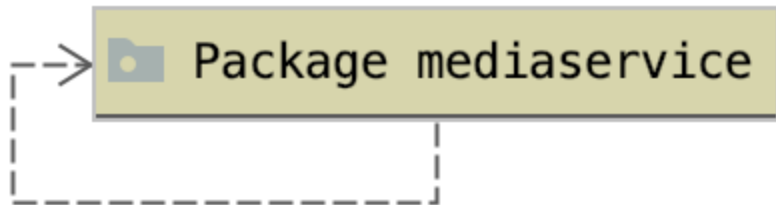
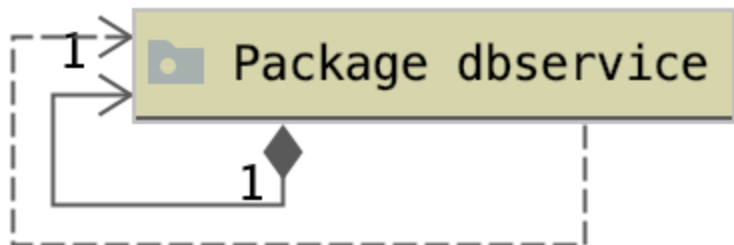
UserController		
m 🔒	findUserByUsername(String)	UserModel
m 🔒	searchUsersByUsername(String)	List<UserModel>
m 🔒	createUserAccount(UserModel, String)	Response
m 🔒	getIdfromAuthHeader(String)	String
m 🔒	trackUserWhileLogin(UserModel, String)	void
m 🌱	login(UserCredentialsParams, HttpServletRequest)	Response
m 🌱	logout(String, HttpServletRequest)	Response
m 🌱	signup(UserCredentialsParams)	Response
m 🌱	deleteUserAccount(String)	Response
m 🌱	searchUser(String)	Response
m 🌱	follow(String, String)	Response
m 🌱	unfollow(String, String)	Response
m 🌱	findUserById(String, String)	Response
m 🌱	findUserById2(String, String)	Response
m 🌱	updateUser(UserModel)	Response
m 🌱	getRecentlyInteractedUsers(String)	Response
m 🌱	getUserChats(String)	Response
m 🌱	getPreference(String, String)	Response
m 🌱	setPreference(String, String, String)	Response
m 🌱	setProfilePicture(String, ProfilePictureParams)	Response

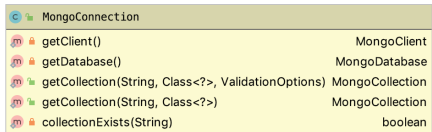
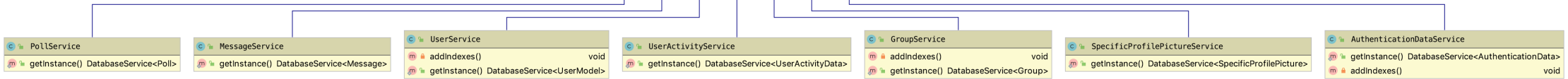
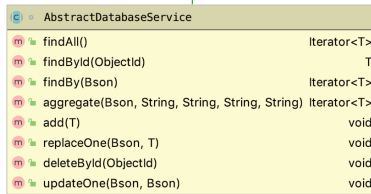
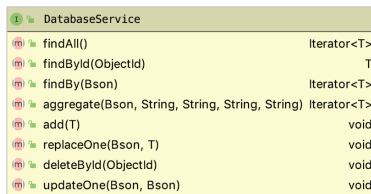
GroupController		
m 🔒	getIdfromAuthHeader(String)	String
m 🌱	createGroup(String, String)	Response
m 🌱	deleteGroup(String, String)	Response
m 🌱	searchGroup(String)	Response
m 🌱	getAllGroups(String)	Response
m 🌱	getAllUsersOfGroup(String)	Response
m 🌱	inviteUserToGroup(String, String)	Response
m 🌱	getCurrentInvitedUsers(String, String)	Response
m 🌱	rejectInvitedUser(String, String, String)	Response
m 🌱	addUserToGroup(String, String, String)	Response
m 🌱	removeUserFromGroup(String, String, String)	Response
m 🌱	assignModerator(String, String, String)	Response
m 🌱	createSecureGroup(UserCredentialsParams, String)	Response
m 🌱	accessGroup(UserCredentialsParams, String)	Response
m 🌱	setGroupPassword(UserCredentialsParams, String)	Response
m 🌱	changePassword(UserCredentialsParams, String)	Response
m 🌱	fetchAllGroups(String)	Response
m 🔒	createGroupAccount(Group, String)	Response
m 🔒	modifyGroupAccount(Group, String, String)	Response

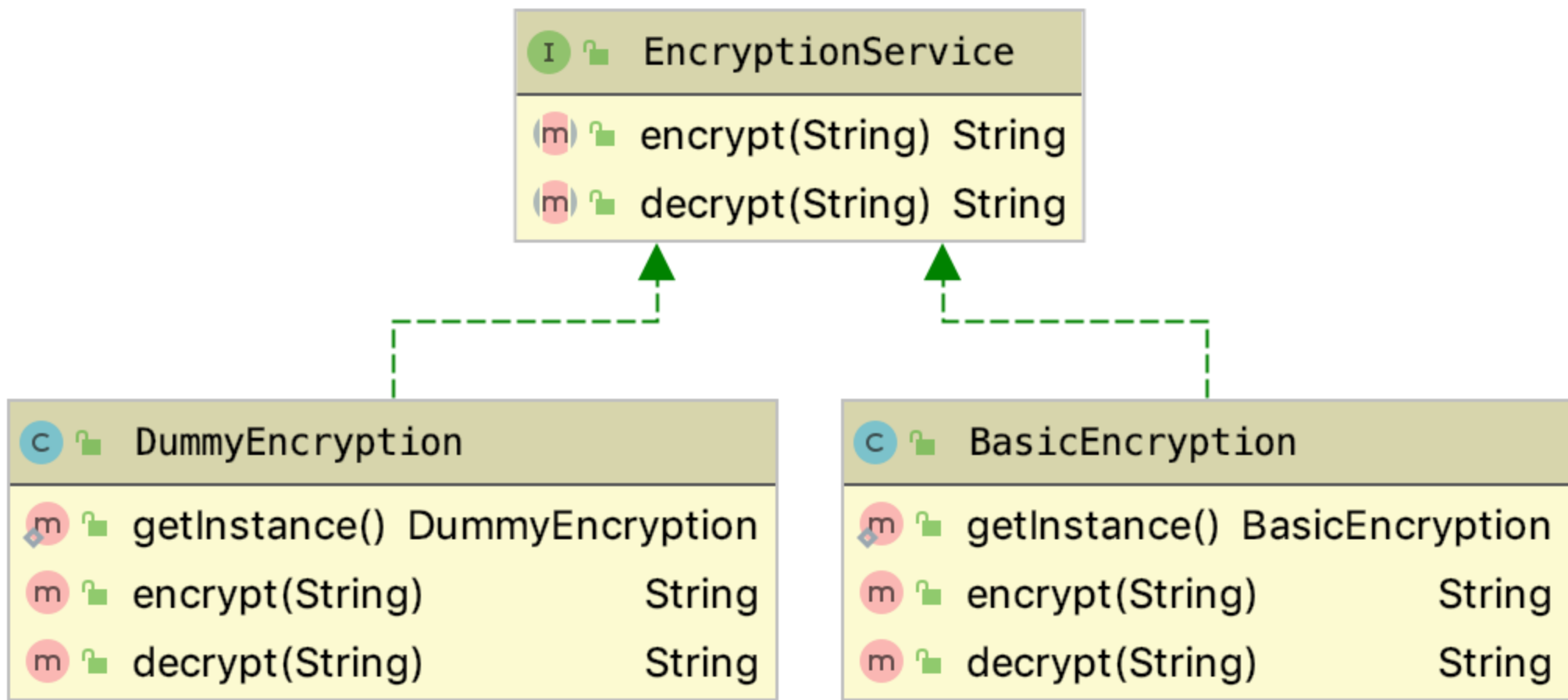
MessageController		
m 🔒	getIdfromAuthHeader(String)	String
m 🌱	deleteMessage(String, String)	Response
m 🌱	getAllMessages(String, String)	Response
m 🌱	getAllGroupMessages(String, String)	Response
m 🌱	createMediaMessage(CreateMediaMessageParams)	Response
m 🌱	createPollMessage(CreatePollMessageParams)	Response
m 🌱	addPollResponse(AddPollResponseParams)	Response
m 🌱	removePollResponse(RemovePollResponseParams)	Response
m 🌱	getPoll(String)	Response
m 🔒	getAllMessagesHelper(String, String)	List<Message>
m 🔒	getDecryptedMessages(List<Message>)	List<Message>


GovernmentRegulationController		
m 🔒	getUsernameFromUserId(ObjectId)	String
m 🔒	getGroupNameFromGroupId(ObjectId)	String
m 🔒	getIdfromAuthHeader(String)	String
m 🔒	checkGovernmentUser(String)	boolean
m 🌱	trackUser(String, String)	Response
m 🌱	getTrackedUserInfo(String)	Response
m 🌱	untrackUser(String, String)	Response
m 🌱	getAllInformation(String)	Response
m 🌱	getMessageInfoForUser(String, String)	Response


CorsFilter		
m 🌱	filter(ContainerRequestContext, ContainerResponseContext)	void










I  **MediaService**


(m)  upload(byte[], String) String


(m)  delete(String) void


(m)  mediaExists(String) boolean




C  **AWSMediaService**

(m)  getInstance() AWSMediaService

(m)  upload(byte[], String) String

(m)  delete(String) void

(m)  mediaExists(String) boolean

UserCredentialsParams	
getId()	ObjectId
setId(ObjectId)	void
getUsername()	String
setUsername(String)	void
getPassword()	String
setPassword(String)	void
getName()	String
setName(String)	void
getGroupname()	String
setGroupname(String)	void
getOldPassword()	String
setOldPassword(String)	void
getNewPassword()	String
setNewPassword(String)	void
getOAuth()	String
setOAuth(String)	void

UserInformation	
getId()	String
setId(String)	void
getUsername()	String
setUsername(String)	void
getName()	String
setName(String)	void
getFollowers()	List<String>
setFollowers(List<String>)	void
getFollowing()	List<String>
setFollowing(List<String>)	void
getGroups()	List<String>
setGroups(List<String>)	void
getIsTracked()	String
setIsTracked(String)	void

UserMessagesInformation	
getUsername()	String
setUsername(String)	void
getSender()	String
setSender(String)	void
getReceiver()	String
setReceiver(String)	void
getGroup()	String
setGroup(String)	void
getMessage()	String
setMessage(String)	void

CreatePollMessageParams	
getSenderId()	String
setSenderId(String)	void
getReceiverId()	String
setReceiverId(String)	void
getOptions()	List<String>
setOptions(List<String>)	void
getPollQuestion()	String
setPollQuestion(String)	void

CreateMediaMessageParams	
getSenderId()	String
setSenderId(String)	void
getReceiverId()	String
setReceiverId(String)	void
getFileName()	String
setFileName(String)	void
getFileContents()	String
setFileContents(String)	void

AddPollResponseParams	
getMessageId()	String
setMessageId(String)	void
getUserId()	String
setUserId(String)	void
getResponse()	String
setResponse(String)	void

RemovePollResponseParams	
getMessageId()	String
setMessageId(String)	void
getUserId()	String
setUserId(String)	void

UserMessageParams	
getUserId()	String
setUserId(String)	void
getMessageId()	String
setMessageId(String)	void

SenderReceiverParams	
getSenderId()	String
setSenderId(String)	void
getReceiverId()	String
setReceiverId(String)	void

ProfilePictureParams	
getProfilePictureContents()	String
getForUser()	String
getFileName()	String

