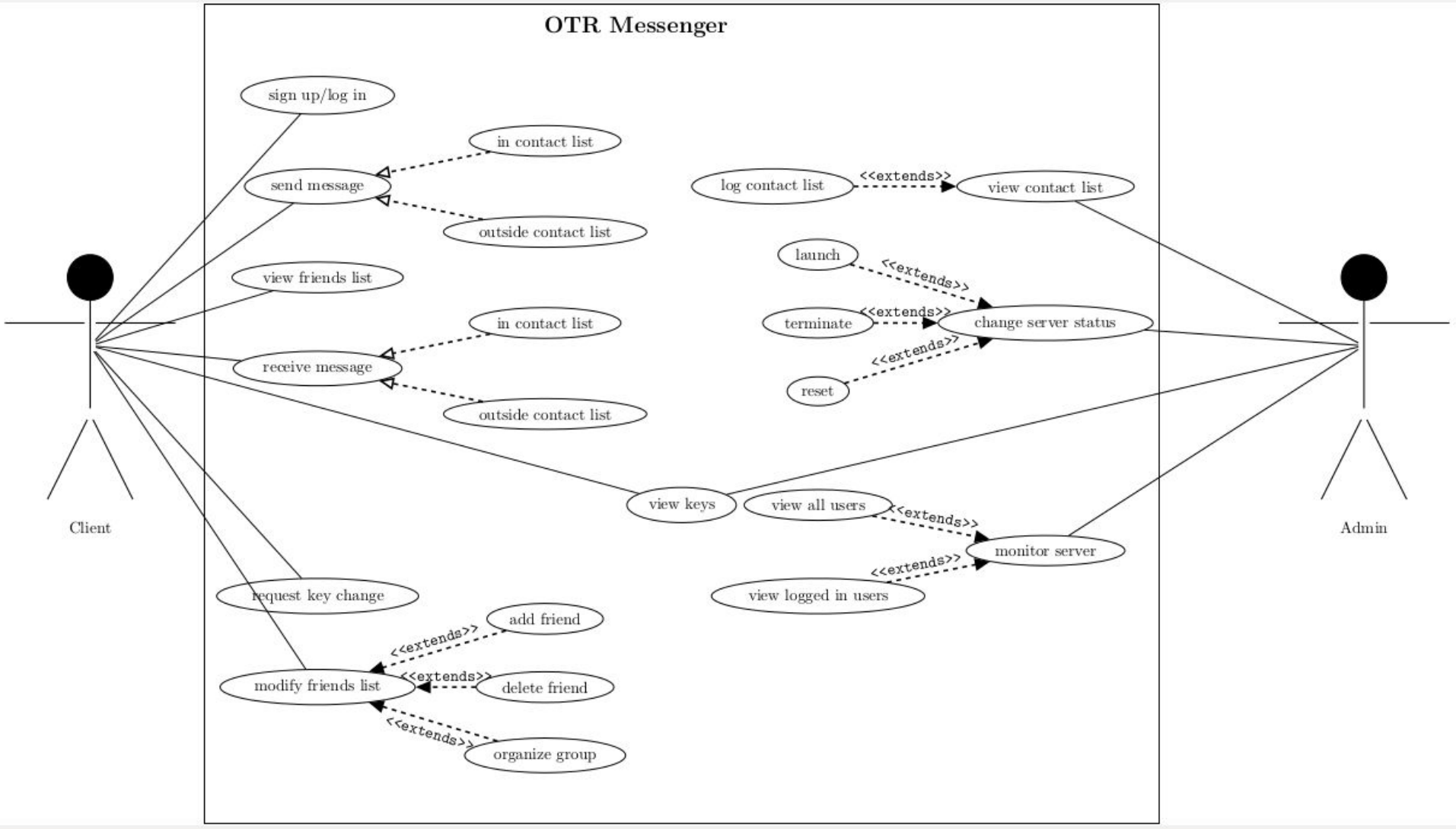
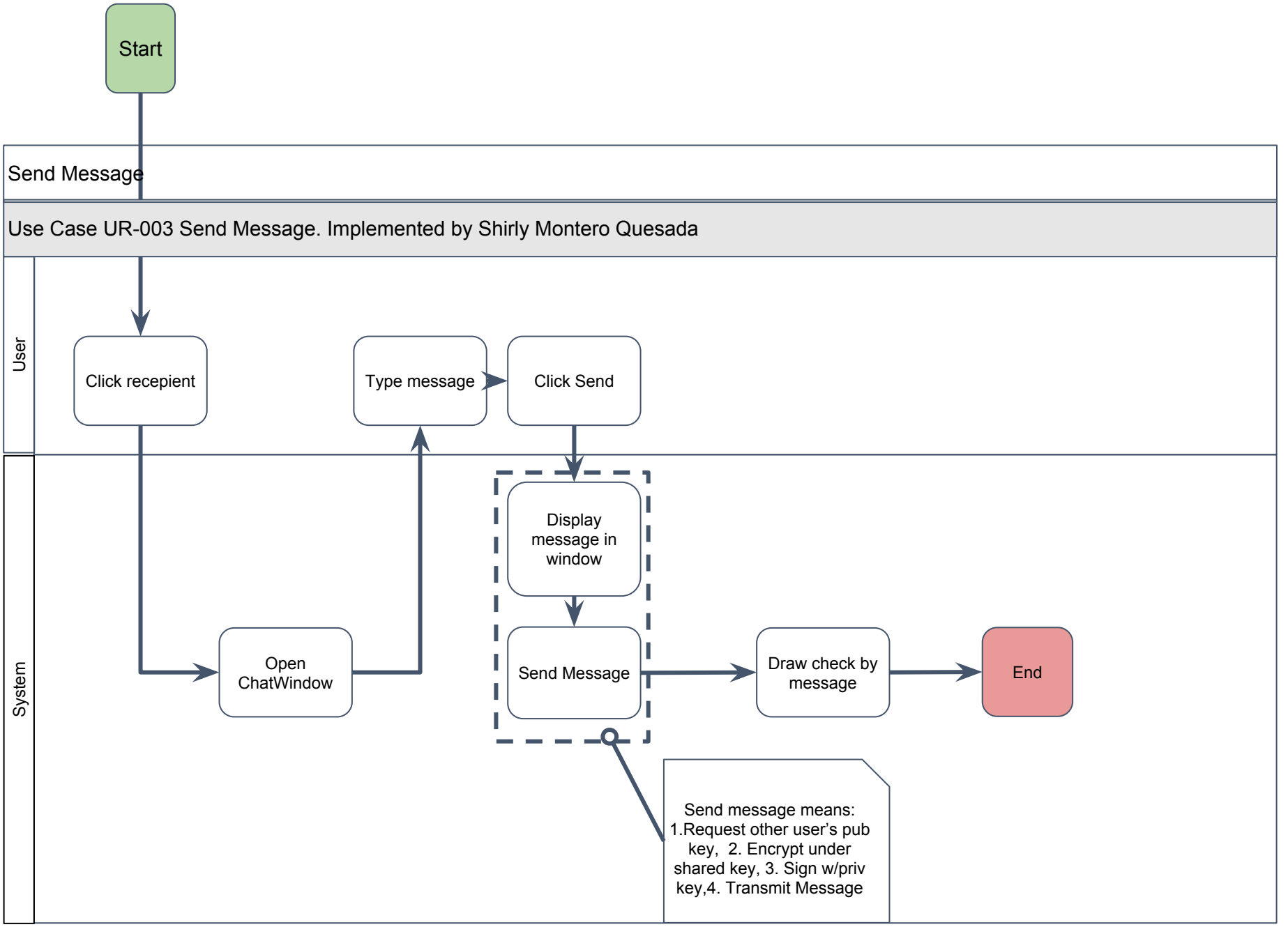
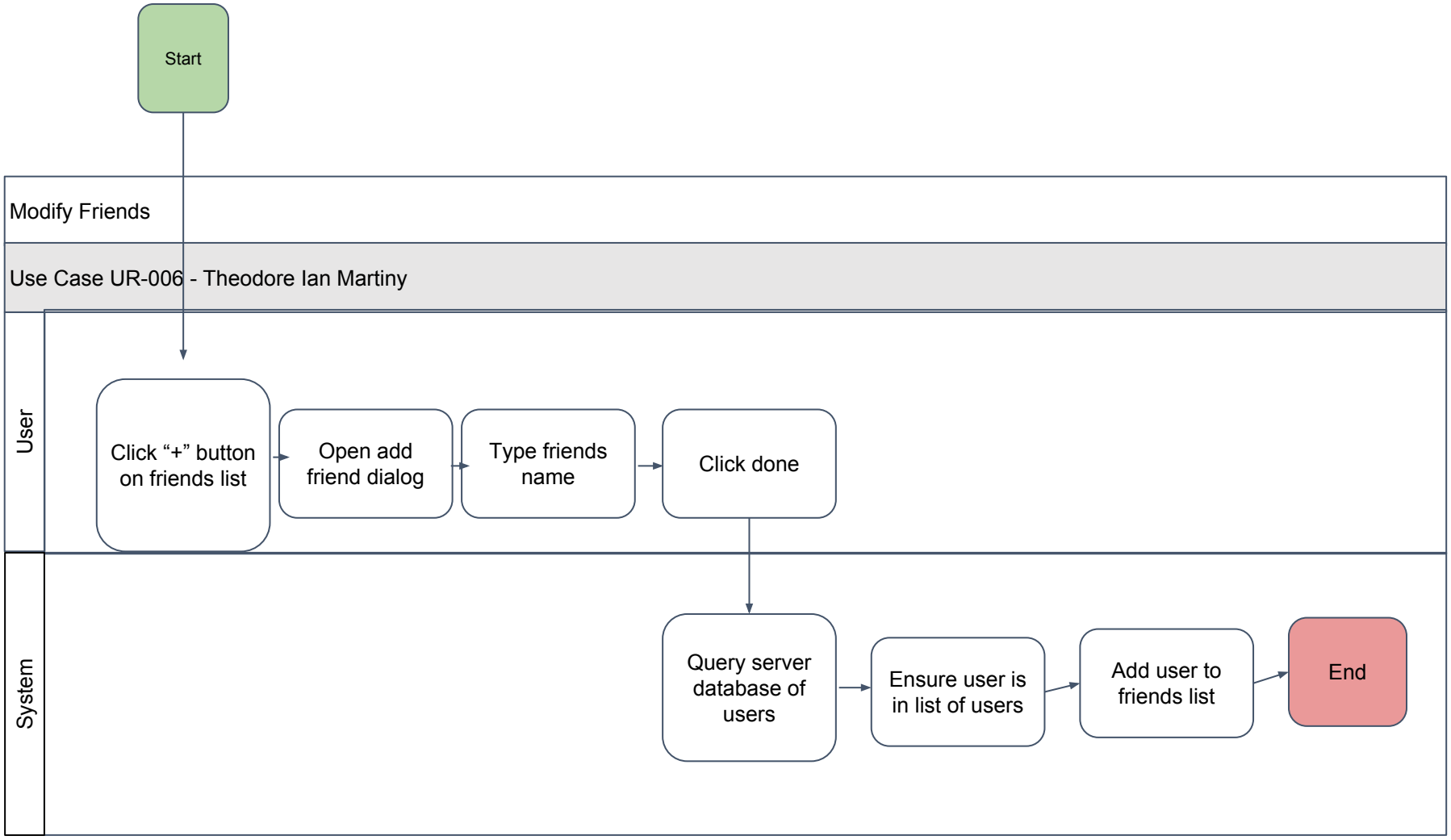
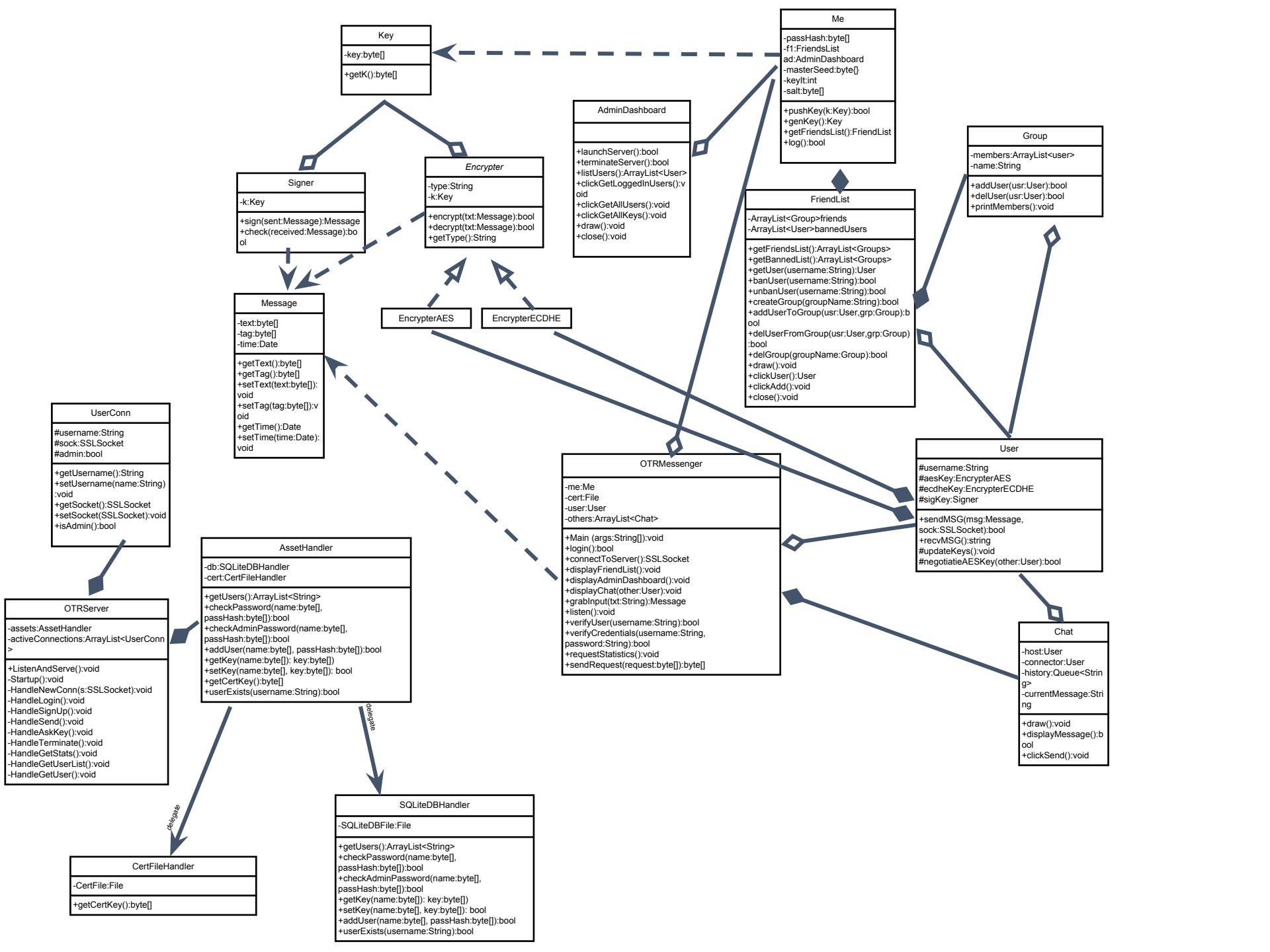
35_OTR-Messenger

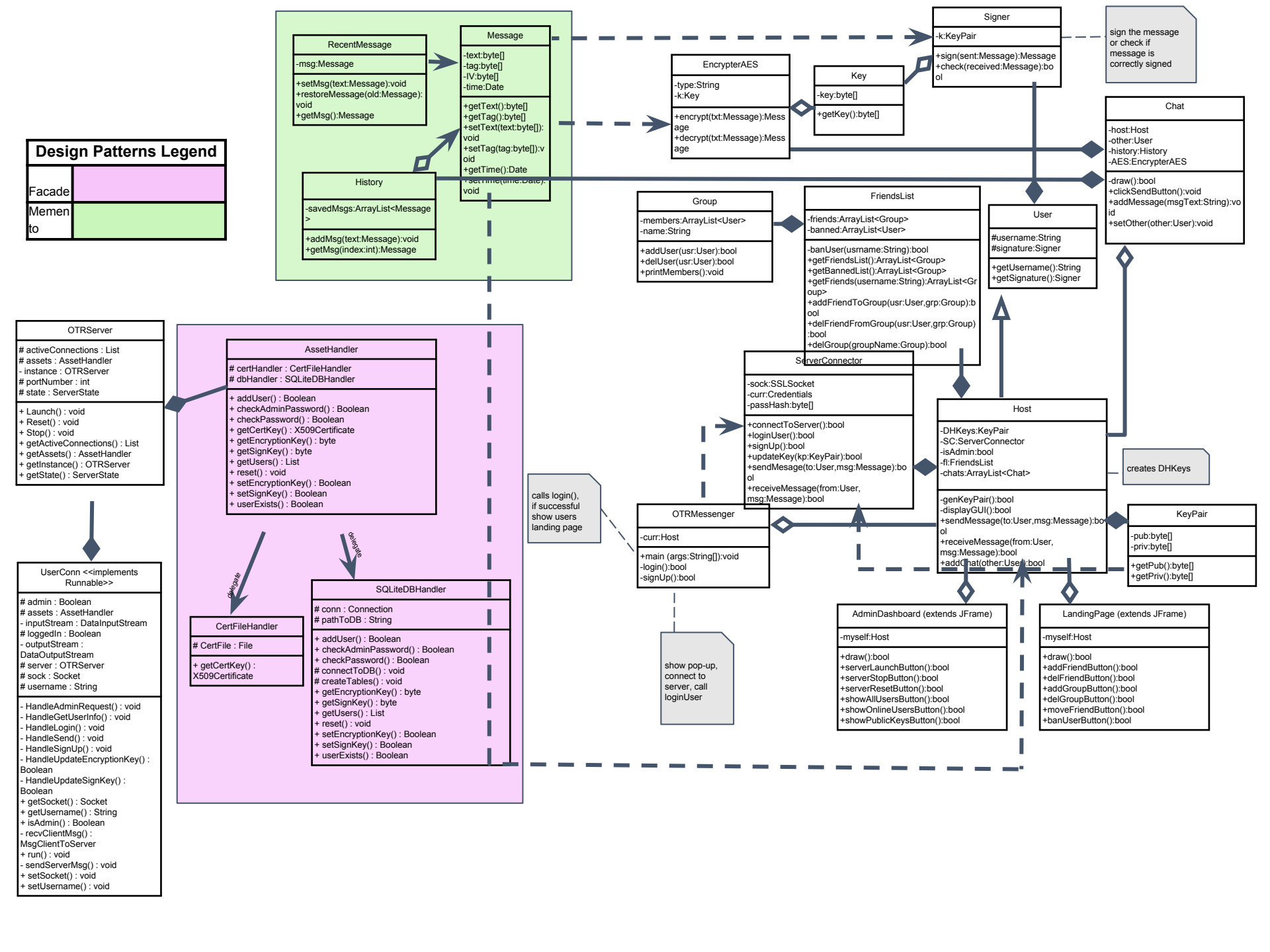
Ian Martiny, Sergey Frolov, Shirly Montero Quesada


OTR Messenger
sign up/log in
in contact list
send message
outside contact list
view friends list
in contact list
receive message
outside contact list
request key change
add friend
<<extends>>
modify friends list
<<extends>>
delete friend
<<extends>>
organize group
log contact list
<<extends>>
view contact list
launch
<<extends>>
terminate
<<extends>>
change server status
<<extends>>
reset
view keys
view all users
<<extends>>
monitor server
<<extends>>
view logged in users
Client
Admin


Start
Send Message
Use Case UR-003 Send Message. Implemented by Shirly Montero Quesada
User
Click recepient
Type message
Click Send
System
Display message in window
Open ChatWindow
Send Message
Draw check by message
End
Send message means: 1.Request other user's pub key, 2. Encrypt under shared key, 3. Sign w/priv key,4. Transmit Message


Start
Modify Friends
Use Case UR-006 - Theodore Ian Martiny
User
Click “+” button on friends list
Open add friend dialog
Type friends name
Click done
System
Query server database of users
Ensure user is in list of users
Add user to friends list
End

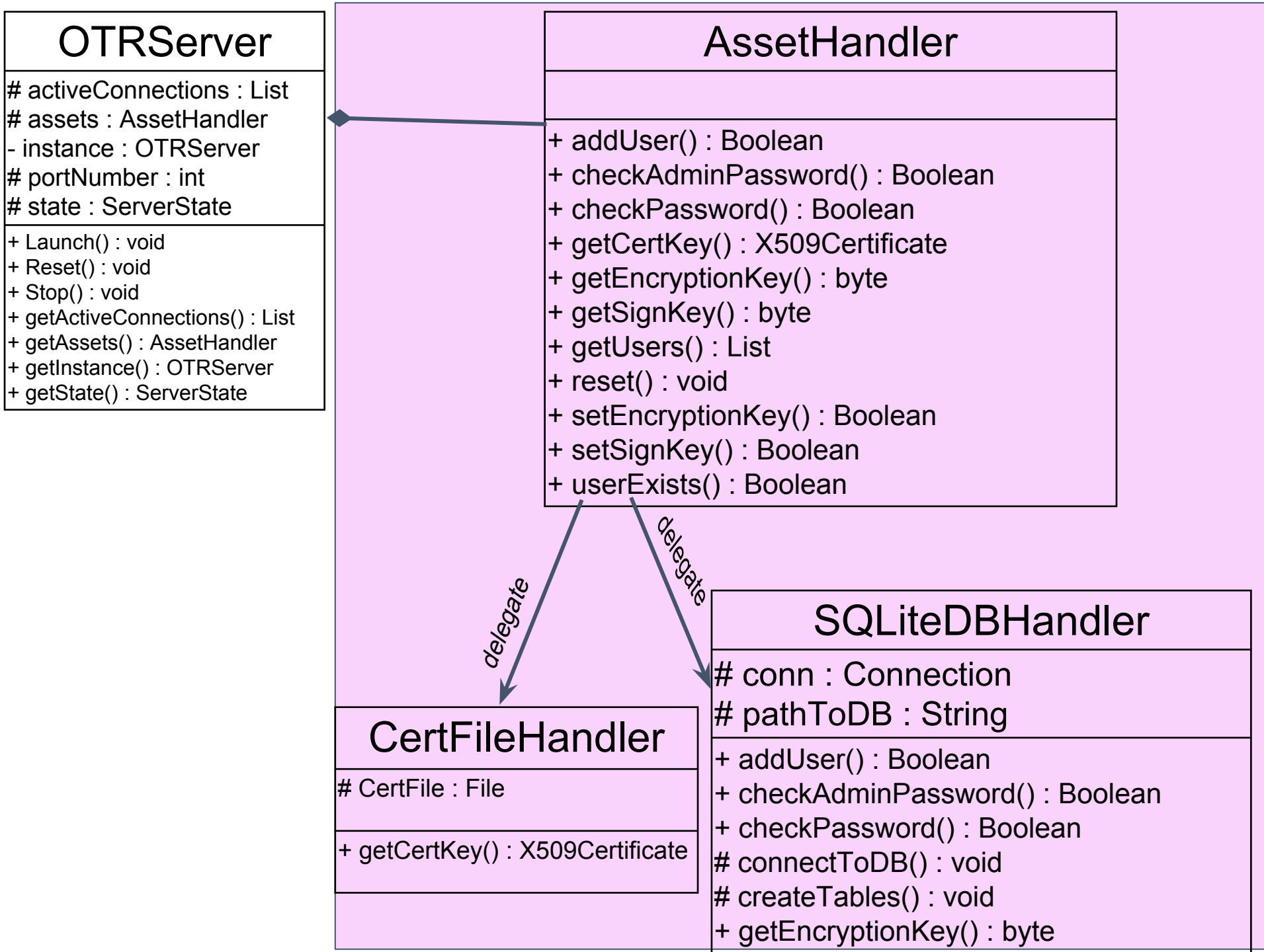

Key
-key:byte[]
+getK():byte[]
Me
-passHash:byte[]
-f1:FriendsList
ad:AdminDashboard
-masterSeed:byte{}
-keyIt:int
-salt:byte[]
+pushKey(k:Key):bool
+genKey():Key
+getFriendsList():FriendList
+log():bool
AdminDashboard
+launchServer():bool
+terminateServer():bool
+listUsers():ArrayList<User>
+clickGetLoggedInUsers():void
+clickGetAllUsers():void
+clickGetAllKeys():void
+draw():void
+close():void
Group
-members:ArrayList<user>
-name:String
+addUser(usr:User):bool
+delUser(usr:User):bool
+printMembers():void
Signer
-k:Key
+sign(sent:Message):Message
+check(received:Message):bool
Encrypter
-type:String
-k:Key
+encrypt(txt:Message):bool
+decrypt(txt:Message):bool
+getType():String
FriendList
-ArrayList<Group>friends
-ArrayList<User>bannedUsers
+getFriendsList():ArrayList<Groups>
+getBannedList():ArrayList<Groups>
+getUser(username:String):User
+banUser(username:String):bool
+unbanUser(username:String):bool
+createGroup(groupName:String):bool
+addUserToGroup(usr:User,grp:Group):bool
+delUserFromGroup(usr:User,grp:Group):bool
+delGroup(groupName:Group):bool
+draw():void
+clickUser():User
+clickAdd():void
+close():void
Message
-text:byte[]
-tag:byte[]
-time:Date
+getText():byte[]
+getTag():byte[]
+setText(text:byte[]):void
+setTag(tag:byte[]):void
+getTime():Date
+setTime(time:Date):void
EncrypterAES
EncrypterECDHE
UserConn
#username:String
#sock:SSLSocket
#admin:bool
+getUsername():String
+setUsername(name:String):void
+getSocket():SSLSocket
+setSocket(SSLSocket):void
+isAdmin():bool
OTRMessenger
-me:Me
-cert:File
-user:User
-others:ArrayList<Chat>
+Main (args:String[]):void
+login():bool
+connectToServer():SSLSocket
+displayFriendList():void
+displayAdminDashboard():void
+displayChat(other:User):void
+grabInput(txt:String):Message
+listen():void
+verifyUser(username:String):bool
+verifyCredentials(username:String, password:String):bool
+requestStatistics():void
+sendRequest(request:byte[]):byte[]
User
#username:String
#aesKey:EncrypterAES
#ecdheKey:EncrypterECDHE
#sigKey:Signer
+sendMSG(msg:Message, sock:SSLSocket):bool
+recvMSG():string
#updateKeys():void
#negotiatieAESKey(other:User):bool
AssetHandler
-db:SQLiteDBHandler
-cert:CertFileHandler
+getUsers():ArrayList<String>
+checkPassword(name:byte[], passHash:byte[]):bool
+checkAdminPassword(name:byte[], passHash:byte[]):bool
+addUser(name:byte[], passHash:byte[]):bool
+getKey(name:byte[]): key:byte[])
+setKey(name:byte[], key:byte[]): bool
+getCertKey():byte[]
+userExists(username:String):bool
Chat
-host:User
-connector:User
-history:Queue<String>
-currentMessage:String
+draw():void
+displayMessage():bool
+clickSend():void
OTRServer
-assets:AssetHandler
-activeConnections:ArrayList<UserConn>
+ListenAndServe():void
-Startup():void
-HandleNewConn(s:SSLSocket):void
-HandleLogin():void
-HandleSignUp():void
-HandleSend():void
-HandleAskKey():void
-HandleTerminate():void
-HandleGetStats():void
-HandleGetUserList():void
-HandleGetUser():void
delegate
delegate
CertFileHandler
-CertFile:File
+getCertKey():byte[]
SQLiteDBHandler
-SQLiteDBFile:File
+getUsers():ArrayList<String>
+checkPassword(name:byte[], passHash:byte[]):bool
+checkAdminPassword(name:byte[], passHash:byte[]):bool
+getKey(name:byte[]): key:byte[])
+setKey(name:byte[], key:byte[]): bool
+addUser(name:byte[], passHash:byte[]):bool
+userExists(username:String):bool


Design Patterns Legend
Facade
Memento
RecentMessage
-msg:Message
+setMsg(text:Message):void
+restoreMessage(old:Message):void
+getMsg():Message
Message
-text:byte[]
-tag:byte[]
-IV:byte[]
-time:Date
+getText():byte[]
+getTag():byte[]
+setText(text:byte[]):void
+setTag(tag:byte[]):void
+getTime():Date
History
-savedMsgs:ArrayList<Message>
+addMsg(text:Message):void
+getMsg(index:int):Message
Signer
-k:KeyPair
+sign(sent:Message):Message
+check(received:Message):bool
sign the message or check if message is correctly signed
EncrypterAES
-type:String
-k:Key
+encrypt(txt:Message):Message
+decrypt(txt:Message):Message
Key
-key:byte[]
+getKey():byte[]
Chat
-host:Host
-other:User
-history:History
-AES:EncrypterAES
-draw():bool
+clickSendButton():void
+addMessage(msgText:String):void
+setOther(other:User):void
Group
-members:ArrayList<User>
-name:String
+addUser(usr:User):bool
+delUser(usr:User):bool
+printMembers():void
FriendsList
-friends:ArrayList<Group>
-banned:ArrayList<User>
-banUser(usrname:String):bool
+getFriendsList():ArrayList<Group>
+getBannedList():ArrayList<Group>
+getFriends(username:String):ArrayList<Group>
+addFriendToGroup(usr:User,grp:Group):bool
+delFriendFromGroup(usr:User,grp:Group):bool
+delGroup(groupName:Group):bool
User
#username:String
#signature:Signer
+getUsername():String
+getSignature():Signer
OTRServer
activeConnections : List
assets : AssetHandler
- instance : OTRServer
portNumber : int
state : ServerState
+ Launch() : void
+ Reset() : void
+ Stop() : void
+ getActiveConnections() : List
+ getAssets() : AssetHandler
+ getInstance() : OTRServer
+ getState() : ServerState
AssetHandler
certHandler : CertFileHandler
dbHandler : SQLiteDBHandler
+ addUser() : Boolean
+ checkAdminPassword() : Boolean
+ checkPassword() : Boolean
+ getCertKey() : X509Certificate
+ getEncryptionKey() : byte
+ getSignKey() : byte
+ getUsers() : List
+ reset() : void
+ setEncryptionKey() : Boolean
+ setSignKey() : Boolean
+ userExists() : Boolean
ServerConnector
-sock:SSLSocket
-curr:Credentials
-passHash:byte[]
+connectToServer():bool
+loginUser():bool
+signUp():bool
+updateKey(kp:KeyPair):bool
+sendMesage(to:User,msg:Message):bool
+receiveMessage(from:User, msg:Message):bool
Host
-DHKeys:KeyPair
-SC:ServerConnector
-isAdmin:bool
-fl:FriendsList
-chats:ArrayList<Chat>
-genKeyPair():bool
-displayGUI():bool
+sendMessage(to:User,msg:Message):bool
+receiveMessage(from:User, msg:Message):bool
+addChat(other:User):bool
creates DHKeys
KeyPair
-pub:byte[]
-priv:byte[]
+getPub():byte[]
+getPriv():byte[]
calls login(), if successful show users landing page
OTRMessenger
-curr:Host
+main (args:String[]):void
-login():bool
-signUp():bool
show pop-up, connect to server, call loginUser
AdminDashboard (extends JFrame)
-myself:Host
+draw():bool
+serverLaunchButton():bool
+serverStopButton():bool
+serverResetButton():bool
+showAllUsersButton():bool
+showOnlineUsersButton():bool
+showPublicKeysButton():bool
LandingPage (extends JFrame)
-myself:Host
+draw():bool
+addFriendButton():bool
+delFriendButton():bool
+addGroupButton():bool
+delGroupButton():bool
+moveFriendButton():bool
+banUserButton():bool
UserConn <<implements Runnable>>
admin : Boolean
assets : AssetHandler
- inputStream : DataInputStream
loggedIn : Boolean
- outputStream : DataOutputStream
server : OTRServer
sock : Socket
username : String
- HandleAdminRequest() : void
- HandleGetUserInfo() : void
- HandleLogin() : void
- HandleSend() : void
- HandleSignUp() : void
- HandleUpdateEncryptionKey() : Boolean
- HandleUpdateSignKey() : Boolean
+ getSocket() : Socket
+ getUsername() : String
+ isAdmin() : Boolean
- recvClientMsg() : MsgClientToServer
+ run() : void
- sendServerMsg() : void
+ setSocket() : void
+ setUsername() : void
delegate
delegate
CertFileHandler
CertFile : File
+ getCertKey() : X509Certificate
SQLiteDBHandler
conn : Connection
pathToDB : String
+ addUser() : Boolean
+ checkAdminPassword() : Boolean
+ checkPassword() : Boolean
# connectToDB() : void
# createTables() : void
+ getEncryptionKey() : byte
+ getSignKey() : byte
+ getUsers() : List
+ reset() : void
+ setEncryptionKey() : Boolean
+ setSignKey() : Boolean
+ userExists() : Boolean

DEMO

Available on our project github page

https://github.com/SergeyFrolov/otr-messenger/blob/master/35_OTR-Messenger_Video.mp4


OTRServer
activeConnections : List
assets : AssetHandler
- instance : OTRServer
portNumber : int
state : ServerState
+ Launch() : void
+ Reset() : void
+ Stop() : void
+ getActiveConnections() : List
+ getAssets() : AssetHandler
+ getInstance() : OTRServer
+ getState() : ServerState
AssetHandler
+ addUser() : Boolean
+ checkAdminPassword() : Boolean
+ checkPassword() : Boolean
+ getCertKey() : X509Certificate
+ getEncryptionKey() : byte
+ getSignKey() : byte
+ getUsers() : List
+ reset() : void
+ setEncryptionKey() : Boolean
+ setSignKey() : Boolean
+ userExists() : Boolean
delegate
delegate
CertFileHandler
CertFile : File
+ getCertKey() : X509Certificate
SQLiteDBHandler
conn : Connection
pathToDB : String
+ addUser() : Boolean
+ checkAdminPassword() : Boolean
+ checkPassword() : Boolean
# connectToDB() : void
# createTables() : void
+ getEncryptionKey() : byte

PROTOBUFF

Protocol Buffers: why?

Protocol buffers is a mechanism to serialize data -- think XML

Protocol Buffers: why?

Protocol buffers is a mechanism to serialize data -- think XML, but

- Smaller
- Faster
- Simpler (except for initial setup)
- Extendable

Protocol Buffers: how?

1. Define protobuf spec once.

Protocol Buffers: spec

```
message Credentials {
    required bytes username = 1;
    required bytes passwordHash = 2;
    required bool  signUp = 3;
    required bool  admin = 4;
}
```

Protocol Buffers: how?

1. Define protobuf spec once.
2. Autogenerate code for any popular language (Java, Python, C++, Golang, Ruby, C#, Rust etc)

Protocol Buffers: how?

1. Define protobuf spec once.
2. Autogenerate code for any popular language (Java, Python, C++, Golang, Ruby, C#, Rust etc)
3. Protobuf!

Protocol Buffers: build message

```
Credentials.Builder msg = Credentials.newBuilder();
```

Protocol Buffers: build message

```
Credentials.Builder msg = Credentials.newBuilder();
msg.setSignUp(false);
```

Protocol Buffers: build message

```
Credentials.Builder msg = Credentials.newBuilder();
msg.setSignUp(false);
msg.setUsername("CookieMonster".getBytes());
// could've used String, if wanted to
```

Protocol Buffers: build message

```
Credentials.Builder msg = Credentials.newBuilder();
msg.setSignUp(false);
msg.setUsername("CookieMonster".getBytes());
// could've used String, if wanted to
msg.build(); // returns bytes to send
```

Protocol Buffers: print message (debug)

```
> System.out.print(creds.toString())
```

Protocol Buffers: print message (debug)

> System.out.print(creds.toString())

```
credentials {
  username: "lan"
  passwordHash: "*TRUNCATED*"
  signUp: false
  admin: false
}
```

Protocol Buffers: SignUp function

```
if (creds.getAdmin()) {
      success = false;//no remote admin signup
} else {
      success = assets.addUser(
creds.getUsername().toByteArray(),
creds.getPasswordHash().toByteArray());
}
```

Thanks!