Setting a new project up notes:

When stating a new project I spend some time researching the problem, and this time I have spend a fraction of what I would usually do.

I have had a brief scan at the following links:

* <https://engineering.riotgames.com/news/chat-service-architecture-servers>
* <https://softwareengineering.stackexchange.com/questions/339882/system-design-scalable-chat-server>
* <https://www.isode.com/whitepapers/xmpp-bosh.html>
* https://codeabout.wordpress.com/2011/03/06/building-a-simple-server-client-application-using-c/

I then have a look at the technologies available and decide what to use, in this case I would use a multi layer application compose of a UI using MVC, the UI would use SignalR to talk to the controllers, as this has a fail over mechanism, from web sockets to JavaScript pulling, this would communicate with the server using WCF and this would talk to a business layer that in terns would talk to a repository layer, and there would be an interface between each layer so we can do mocking and use DI. There are cases where the Service layer could talk straight to the repository this in cases where the call would be a pass-through call.

Then I create mocks using balsamiq or a similar UI modelling tool.

I then think about the entities that will be involve in the solution, essentially the problem domain and try and find a domain expert so I can get all the detail and exemptions in the system.

I will pick up the task from the instructions given, I have copy here for ease of reading and as I will be updating this as I go “normally” so here goes:

Task 3 description:

- is the original task, I have made notes and extended some of the concepts.

-Write the domain model for a chat service.

-Designing a chat service is certainly far beyond the scope of what could be completed in a recruitment test.

-Please focus only in what is a fair representation of its model.

Entities:

- Users

User

Id

UserName

Email

Status

UserId

Status

StatusMessage

- Conversations

Group

GroupId

GroupName

OwnerUserId

DateCreated

GroupUserMap

GroupId

UserId

DateCreated

ActiveConversations

Id

ConversationType (group/private)

Message

MessageSent

DateTimeCreated

ActiveUser

UserId

ActiveConversationId

MessageReceived

DateTimeCreated

- Contacts

Contact

UserId

TargetUserId

TargetAccepted

DateTimeCreated

-Behaviour:

- Get user profile.

- Add contact request.

- Update status message (user is online, offline, custom message)

- Creating private and group conversations.

- Send messages to private and group conversations.

-Code requirements:

-1. The code should fulfil OOP and SOLID principles.

-2. The code should be maintainable.

-Hint:

- This task evaluates design & clean code over algorithmic.

- There is not exact correct answer. Feel free to add more Entities and/or Behaviour