

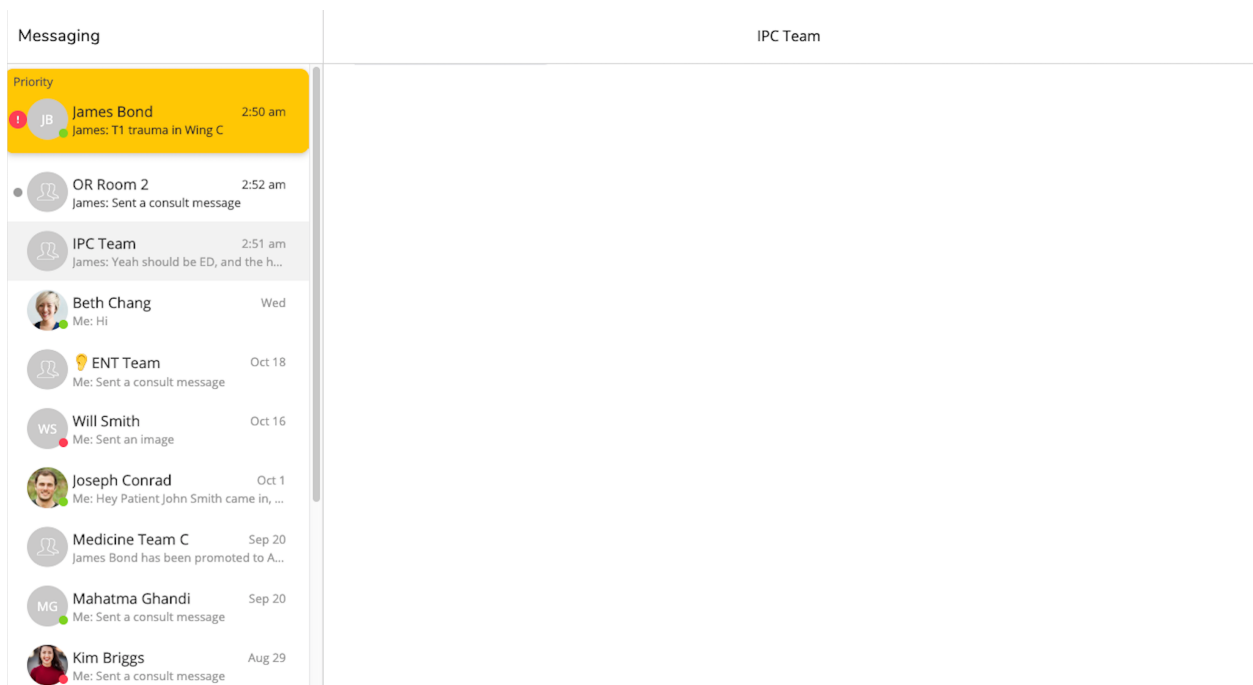
Hypercare

Take Home Challenge!

Objective:

Implement the User Interface based on the attached images and the guidelines below. The purpose of the assignment is to hit our API, and after fetching the data, present the chats data in a list view.

Using Typescript & React is a must. You also don't need to worry about responsive design. There are also some bonus tasks at the end if you want to really stand out.



Specifications

Chat List

- The chats should be listed from most recent to least recent
 - In the response you will see a "lastMessage" property for each chat. Use the "dateCreated" property of the "lastMessage" property to sort your chats
- Ignore the top section, which they contains priority chats (it is a bonus), only render chat where isPriority is false

For Each Chat

- Show the avatar of the recipient if it is a single chat or the group icon if it is a group chat
 - There is a "type" property for each chat. The possible values for this are "self", "single", or "group". You can use this property to determine the type of chat
 - Ignore the colored dot at bottom right of the avatar
- Show the title of the chat
 - Each chat has a title property
 - If the title property is NULL the chat title should be a concatenation of the first names of the members
 - If the generated title is longer than 25 characters (this might differ based on screen resolution. Use your best discretion) then it should trim it to say "+ {remaining_users} people"
- Show the last message sent
 - If the message was sent by you then it should be followed by a "me: "
 - If the message was not sent by you then it should be, "{first_name}:" followed by the message
 - The message content should take no more than 1 line
- Show the last message sent timestamp, the time should be displayed next to the title
- Clicking on a chat should change its background color to a darker grey color as an indication for active chat, and the onclick event can change the right side page to any information you would like to show(e.g. Chat Title on top)
- Once finished, please upload code to a github repo and share the repository link with Hamad (hamad@hypercare.com)

Technical Aspect

Constants:

- **AccessToken:** `3a1aeb259e5987a8be639d05714cc1dd2f87ff15`
- **hypercare-scope:** `eyJvcmdhbml6YXRpb25JZCI6MX0=`

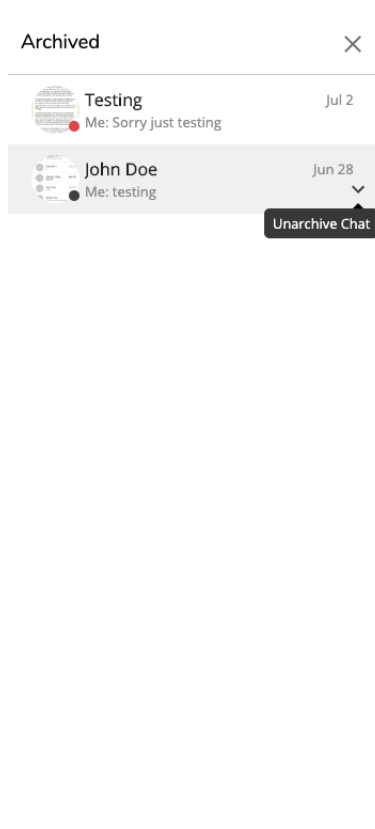
Network Calls:

- (replace constants from above)
- **Fetch Chats**
- `curl --location 'https://api.staging.hypercare.com/graphql/private' \`
- `--header 'hypercare-scope: {hypercare-scope}' \`
- `--header 'Content-Type: application/json' \`
- `--header 'Authorization: Bearer {AccessToken}' \`
- `--data '{"query":"query organizationChats($continuationId: ID, $limit: Int, $isPriority: Boolean) {\n chatsForOrganization(continuationId: $continuationId, limit: $limit, isPriority: $isPriority) {\n chats {\n ...basicChatFields\n unreadPriorityMessages\n }\n }\n}\n\nfragment basicChatFields on Chat {\n chatId: id\n title\n type\n members {\n ...chatMemberFields\n }\n lastMessage {\n ...messageFields\n }\n muted\n status\n dateCreated\n isArchived\n unreadPriorityMessages\n}\n\nfragment chatMemberFields on ChatMember {\n id\n firstName\n lastName\n username\n role\n profilePic {\n url\n }\n status\n privilege\n workStatus\n statusExpiryDate\n statusDescription\n workStatusProxy {\n ...publicUserStatusFields\n }\n}\n\nfragment messageFields on Message {\n id\n priority\n message\n image\n # attachment {\n # url\n # mimeType\n # fileName\n }\n type\n dateCreated\n sender {\n ...publicUserFields\n }\n readBy {\n ...readReceiptFields\n }\n data {\n __typename\n ... on ConsultMessageData {\n mrn\n firstName\n lastName\n details\n }\n }\n}\n\nfragment readReceiptFields on ReadReceipt {\n messageId\n user {\n ...publicUserFields\n }\n timestamp\n}\n\nfragment publicUserFields on PublicUser {\n id\n firstName\n lastName\n username\n role\n profilePic {\n url\n }\n workStatus\n statusDescription\n workStatusProxy {\n ...publicUserStatusFields\n }\n}\n\nfragment publicUserStatusFields on PublicUser {\n id\n firstName\n lastName\n username\n role\n profilePic {\n url\n }\n}\n\n","variables":{"isPriority":false,"limit":20}}'`

Bonus

1) Archive and unarchive chat list, please see example below:

- Create any UI you want for user to navigate and view archived / unarchived chat list
- Each chat item will have a UI toggle for user to archive and unarchive (does not need to be look exactly like the one shown below)
- Archive and unarchive chat should remove them from archive/unarchive chat list to another



Network request for archived chats

```
curl --location 'https://api.staging.hypercare.com/graphql/private' \  
--header 'hypercare-scope: {hypercare-scope}' \  
--header 'Content-Type: application/json' \  
--header 'Authorization: Bearer {AccessToken}' \  
--data '{"query": "query organizationChats($continuationId: ID, $limit: Int, $priority: MessagePriority) {\n  archivedChats(continuationId: $continuationId, limit: $limit, priority: $priority) {\n    chats {\n      ...basicChatFields\n    }\n  }\n}"}
```

```

}\n}\n\nfragment basicChatFields on Chat {\n id\n title\n type\n lastMessage {\n ...messageFields\n }\n muted\n dateCreated\n isArchived\n}\n\nfragment messageFields on Message {\n id\n priorityType\n message\n image\n#\n attachment {\n# url\n# mimeType\n# fileName\n# }\n type\n dateCreated\n sender {\n ...publicUserFields\n }\n readBy\n {\n ...readReceiptFields\n }\n data {\n __typename\n ... on ConsultMessageData {\n mrn\n firstname\n lastname\n details\n }\n }\n}\n\nfragment readReceiptFields on ReadReceipt {\n messageId\n user {\n ...publicUserFields\n }\n timestamp\n}\n\nfragment publicUserFields on PublicUser {\n id\n firstname\n lastname\n username\n role\n profilePic {\n url\n }\n workStatus\n statusDescription\n workStatusProxy {\n ...publicUserStatusFields\n }\n}\n\nfragment publicUserStatusFields on PublicUser {\n id\n firstname\n lastname\n username\n role\n profilePic\n {\n url\n }\n}\n\n\n","variables":{"isPriority":false,"limit":12}}'

```

Notes

Please follow best coding practices and make appropriate use of design and architectural patterns when appropriate. If there are any shortcuts you take or any decisions you might have made that weren't clearly specified please record them down to be discussed during the interview. Best of luck.

Feel free to reach out to me (Hamad) via phone or email if you have any questions. I can be reached at (416) 270-4857 or at hamad@hypercare.com