* What you like about the features and functionality of the library so far.

I like the features and the functionality of the library so far. Allowing users to choose different input options is a useful feature. I like the way you structured the input container and the message container. But I guess there aren’t many features implemented for me to comment on. I am looking forward to seeing the final submission.

* Your thoughts on the examples in their deployed webpage. Were they easy to understand and use based on their explanations in their written work and/or the page itself? Is it clear how the DOM is being manipulated? Is the benefit clear to you as the user?

I think the examples in the webpage is easy to understand and follow through.

Suggestion: It is mentioned that developers are able to customize the option bar. So, it might be a good idea to provide some buttons in the webpage to change the view of the option bar in real time such that people are aware of different customization options.

I think the use case of the library is pretty clear. I like how you separate the input container and the message container. The DOM manipulation is clear. The benefit is clear to me as a user. But improvements can be made: the messages overflow when it exceeds the message container window. It is a good idea to fix it in final submission.

* What you think they could work to improve on, and/or things you didn't quite understand about the library functionality or the potential use cases. As a peer, are there any concerns you might have about the complexity or originality of their library?

A suggestion about importing the js library: in the individual proposal handout, it is recommended that your library code should be included in the <script> tag, but you import the library code in your examples.js which is not recommended. So it might be nice to have it fixed for the final release.

I think the functionality and potential use cases are clear.

Complexity: I am a little concerned about the complexity of the project based on what I have seen so far. Even if you implemented all the features you proposed the complexity might still not be enough. So, I proposed a couple of features at the end. If you managed to implement all the proposed features. Take a look at those ones!

Originality: I think the originality is good. I did a rough search on messaging/commenting js library but was not able to find anything similar. So good job on coming up with this idea.

* Come up with a feature that you think would be a good addition to this library that they didn't talk about and explain why you think it would be nice to have.

Features for the message container:

* + Allowing developers to customize the position of the messages based on users. For example, your messages are on the right-hand side of the window and others are on the left-hand side of the chat window. (why good to have: Users can distinguish between messages sent from different users without looking at the sender)
  + Allowing developers to customize the background color of the messages based on users (why good to have: Users can distinguish between messages sent from different users without looking at the sender)
  + Allowing developers to specify the width and height of the container window. So not let the messages overflow. Changing the view port such that the latest messages are shown. (why good to have: Users care more about the latest messages)
  + Hiding username of the message (why good to have: If the message is coming from yourself, it is not necessary to show it)

Features for the input container:

* + Adding font and font size to the options (why good to have: users might want to change font size and font)