We have decided to go with the object-oriented programming design. We chose it because our project, Educord which is a chatting platform similar to Discord but dedicated for students, consists of the following components: user profiles with a personal calendar and a to-do list for each, servers that represent courses/classes, chatrooms within each server, and messages inside of these chatrooms.

We thought that since for Project 4, we need to fully implement the functionalities, we needed to have a backend server to store all the data, and so the team agreed to write classes for each of the previously mentioned components of the project, and constantly add instances of these classes to the backend server or remove from it as needed.

Since we are using JavaScript, it is natural that a lot of functions will be needed to dictate the flow of the program as well as keep track of the different data. In order to avoid the use of global variables or minimize it as much as possible, it was ideal to go with classes instead. With them, we are able to easily create new instances of a user whenever a new one signs in, and each user will have their own list of servers/courses to which they can add or delete. Also, since the messages displayed in any given chatroom contain information other than the text itself, like the user who sent them and at what time, it is appropriate to write a class for these messages that includes all that information as member variables. This makes storing the chat history in the backend much easier and more convenient.