

## Program 1: NodeJS. Due March 11

You are going to create a NodeJS site with persistent storage with APIs. The site will be a small forum site. The user “logins”, they can post text messages and other users can comment on them. You are not required to create the “front end” website for this. Just the APIs that could be accessed by the React SPA (program 2) and the flutter app (Program 3). This site has Posts and username comment on the posts. You CAN NOT comment on comments.

The data is expected to be persistent across runs, so you can choose to store the information in whichever database (or files) that you want to use.

### Undergraduate:

1. Username is required for all APIs, as you will see below, so it will be a parameter passed to all pages. If it is not there or blank, then an error will be returned.
  - a. Note, I’m not requiring you to “login” a user, ie username and password. You can but it is not required. Instead, the username is just a text name field.
2. A Read posts API. It takes a username and returns the posts (post id, post information, and username.)
3. A Read Comments API. It takes a Post id and returns all the comments for the specific posts (comment id, username, and comment information).
4. A Create Post API. Which takes the username and post information, returns back if it was created or not.
5. A Create Comment API. It takes the Post Id, username, and comment. Returns back if was created or not.
6. An edit post API, which takes the post ID, the username and updated post information. If the username does not match, the update will be denied.
7. An edit comment API. Which takes the post ID, the username, comment id, and updated comment information. If the username does not match, then the update will be denied.
8. Delete a post API It takes the post id information and username. The post and all comments associated with that post ID will be deleted. Again, if the username does not match the Post, then the delete will be denied. The comment username is not relevant to the delete.
9. Delete a comment on a post. Which takes the post id, the comment id, and username. Deletes the comment on that specific post. If the username doesn’t match, the delete is denied.

### Graduate:

1. All the undergraduate version
2. There will be 5 categories. You can name them whatever you want. Each post will be put in a category.
3. You will also need to update all the API above to also include the category ID
4. List category API to return all the categories and category ID.

Testing, I would suggest using something like axios or you can add the necessary website pieces to test all of it.

Use this link to get the GitHub repo: <https://classroom.github.com/a/zmHm8BMM>

Turn in:

1. Make sure the readme has been updated, name, Program 1
2. Update the AI use section
3. Make sure you push everything to GitHub.
4. This will be too difficult to grade without you, so arrangements will be made to have to live demo the site and that everything works. Have a set of tests ready to show the APIs work as well. Or you can create a video showing that everything works.