

Github Project React Js -1

You and three other students are working on a React.js frontend project. The project requires you to create a simple task management application. Each student will work on a separate feature of the application. Additionally, there is a condition that all students must commit their code to GitHub simultaneously. Finally, you need to merge all the changes made by the students into the main branch of the repository. Determine which student will work on which feature and explain how you will handle the merging process.

Solution: Assigning Features:

1. **Student A:** Task creation and deletion functionality
2. **Student B:** Task list display and filtering functionality
3. **Student C:** Task editing and updating functionality
4. **Student D:** User authentication and login functionality

Merging Process: To merge the changes made by each student into the main branch, follow these steps:

1. Each student should create a separate branch from the main branch of the repository to work on their assigned feature. For example, Student A can create a branch called "task-creation."
2. Students should regularly commit their changes to their respective branches as they work on their features. It's important to communicate and coordinate with each other to avoid conflicts and ensure smooth collaboration.
3. Once a student completes their feature and commits their changes to their branch, they should create a pull request to merge their branch into the main branch.
4. One student, preferably the team lead or someone responsible for code integration, should review the pull requests. They should carefully examine the changes made in each branch and resolve any conflicts that arise during the merge process.
5. After reviewing and resolving conflicts, the student responsible for code integration can merge the pull requests into the main branch. This can be done using the merge functionality provided by GitHub or through the command line using Git.

Github Project React Js -1

6. Once all the features are merged into the main branch, the project is considered complete. It's essential to test the application thoroughly after merging to ensure all features work together as expected.

Remember to follow good coding practices, maintain clear communication, and regularly synchronize your work to avoid conflicts and streamline the merging process.

Note: The assignment question and solution provided here are just an example. You can modify them based on your specific requirements and the capabilities of the students working on the project.