

Hashim Aziz Muhammad

Task 5 - React - Back-end Integration

Course: Full Stack Development

Reviewed By: Sam Baloyi

Date Submitted: May 22, 2024, 8:46 a.m.

Date Reviewed: May 22, 2024, 3:14 p.m.

Student number: HA24010012783

Scores

Completeness: 4/4

Efficiency: 4/4

Style: 4/4

Documentation: 4/4

Positive aspects of the submission

Hello, Hashim. I hope you are doing well. Thank you for your submission.

Backend Integration

You've added a new endpoint `/api/message` in your backend server to handle requests for custom messages. The endpoint responds with a custom message defined in the server code.

Frontend Integration

In your frontend code, you've imported the necessary libraries and set up state variables to store the fetched data and custom message.

You've used `useEffect()` to fetch both the data and custom message from their respective endpoints when the component mounts.

The fetched custom message is stored in the `customMessage` state variable.

Your backend and frontend code are logically separated, making it easier to manage and maintain. You've correctly utilised React's state management to store and update data fetched from the server.

You've implemented error handling for fetching both data and the custom message, which is essential for a robust application.

Aspects of the submission that could be improved

Hashim, it would be beneficial to update your `README.md` file in your project. Your `README` will serve as a guide for users and potential contributors, providing essential information about your application. In the `README`, you can detail how to install and run the application, explain its purpose and features, list any API endpoints available, mention the technologies used, and provide any other relevant information. A well-written `README` enhances the usability and accessibility of your project, making it easier for others to understand and engage with. Consider crafting a concise `README` to accompany your application.

Overall feedback

Overall, Hashim, you did a great job with your backend integration, and your frontend code demonstrates a solid understanding of React's state management and `useEffect` hook. Keep up the good work, and don't hesitate to reach out if you need further assistance or feedback.