

Subject: Axios Error Handling Implementation

Dear [Recipient's Name],

I hope this message finds you well.

I am writing to discuss the implementation of error handling in our Axios HTTP requests. To ensure robust functionality and improve user experience, I have outlined a template for error handling that we can adopt in our project.

```
```javascript
```

```
import axios from 'axios';
const apiClient = axios.create({
 baseURL: 'https://api.example.com',
 timeout: 10000, // Set a timeout for requests
});
// Interceptor for handling errors
apiClient.interceptors.response.use(
 response => response,
 error => {
 let errorMessage = 'An unexpected error occurred.';
 if (error.response) {
 // The request was made and the server responded with a status code
 switch (error.response.status) {
 case 400:
 errorMessage = 'Bad Request: Please check your input.';
 break;
 case 401:
 errorMessage = 'Unauthorized: Please log in.';
 break;
 case 404:
 errorMessage = 'Not Found: The requested resource was not found.';
 break;
 case 500:
 errorMessage = 'Server Error: Please try again later.';
 break;
 default:
 errorMessage = 'An error occurred: ' + error.response.statusText;
 }
 } else if (error.request) {
 // The request was made but no response was received
 errorMessage = 'No response received from the server.';
 } else {
 // Something happened in setting up the request that triggered an Error
 errorMessage = error.message;
 }
 // You can customize your error handling further here (e.g., logging)
 alert(errorMessage); // Replace with your preferred method of displaying errors
 return Promise.reject(error);
 }
);
export default apiClient;
```
```

Please let me know your thoughts or if there are any additional considerations we should include.

Best regards,

[Your Name]

[Your Position]

[Your Contact Information]