

AeroAspire -SDE Intern Training

NIRANJAN C N

Week3 Day 3-Oct9

Learning topics:

- Error handling
- Input validation
- Status codes,
- CORS
- connecting with frontend
- testing endpoints

Task:

- Add error handlers (404, 400)
- Validate inputs (e.g. non-empty strings, valid JSON),
- Enable CORS
- From React frontend make calls
- Test from Postman / browser

Questions:

1. How do you test error flows (client sends invalid data)?
 - a. We can use Postman to check for any errors
 - b. Check if there is any error like 400
 - c. Which means the request sent is incorrect
 - d. Perform operations from frontend and backend and check if there is any problem like if there is any issue with input fields that I faced during testing in Postman
2. Describe flow of exception in Flask: what happens if an unhandled exception occurs?
 - a. If there is any unexpected error happens
 - b. And it is not visible with the terminal
 - c. Flask handles these errors and returns respective error codes and we can understand them by the response we get in Postman
 - d. Flask shows a error page in browser which shows all the issues and helps us debug
3. What is CORS? Why browsers block cross origin requests; how to configure CORS in Flask.
 - a. CORS is Cross origin resource sharing
 - b. It is security measure for web browsers
 - c. Browsers block cross origin requests because the server doesn't allow it.

- d. The server doesn't allow any https requests in between websites
 - e. So we use CORS to share the requests
 - f. We can install CORS in Flask by using the package, flask-cors
4. What is the flow of a fetch/Axios request from React to Flask → response → error handling.
- a. First React sends a HTTP request to the Flask API using fetch or axios
 - b. Then Flask receives the request and sends a response that maybe any data or a error message
 - c. Error is handled by Flask and respective error codes and messages are sent in response
 - d. We can handle errors in react by changes in UI like a display of error message
5. How to log or debug failed requests.
- a. We can see the error codes and messages in response to the request we would have sent in postman or browser
 - b. We can check the console tab under inspect to see the errors
 - c. Another method I found it is we can use `console.log()` to print the error messages which is the same we find under the inspect tab