

FileLoader Component

Framework

React

Goal

Use test doubles to isolate and test a component that fetches and displays data.
Replace real API calls with a stub to ensure reliability and speed.

Instructions

1. Create a `fileService` module with a `fetchFile(id)` method.
2. In your test file, use `jest.mock()` to stub the module and return `{ content: 'Sample text' }`.
3. Write a test that renders the component, waits for the content to load, and verifies it appears.
4. Write another test where `fetchFile` returns an empty string or error — verify fallback UI is displayed.
5. Ensure tests check for initial loading state using `screen.getByText('Loading...')`.
6. Use `findByText` to wait for async rendering.

Tech & Tools

- `jest.mock()` for service stubbing
- React Testing Library's `findByText`, `queryByText`
- `useEffect` and `useState` for the component

What to Test

- "Loading..." appears initially
- Fetched content is displayed after loading
- Alternative UI appears if `fetchFile` fails or returns empty
- No actual HTTP requests are made