### Create GitClient Module

#### File: src/GitClient.js

import axios from 'axios';

class GitClient {

async getRepositories(username) {

const response = await axios.get(`https://api.github.com/users/${username}/repos`);

return response.data.map(repo => repo.name);

}

}

export default GitClient;

Modify App Component

#### File: src/App.js

import React, { useEffect, useState } from 'react';import GitClient from './GitClient';

function App() {

const [repos, setRepos] = useState([]);

useEffect(() => {

const client = new GitClient();

client.getRepositories('techiesyed').then(setRepos);

}, []);

return (

<div>

<h1>GitHub Repositories</h1>

<ul>

{repos.map((repo, idx) => <li key={idx}>{repo}</li>)}

</ul>

</div>

);

}

export default App;

Create Unit Test for GitClient

#### File: src/GitClient.test.js

import axios from 'axios';import GitClient from './GitClient';

jest.mock('axios');

describe('Git Client Tests', () => {

test('should return repository names for techiesyed', async () => {

const mockData = [

{ name: 'RepoOne' },

{ name: 'RepoTwo' }

];

axios.get.mockResolvedValue({ data: mockData });

const client = new GitClient();

const repos = await client.getRepositories('techiesyed');

expect(repos).toEqual(['RepoOne', 'RepoTwo']);

});

});

OUTPUT:

