## **Objectives**

* Explain how to consume REST APIs from React applications

In this hands-on lab, you will learn how to:

* Construct a React application that invokes the REST API and fetch data from the API

## **Prerequisites**

The following is required to complete this hands-on lab:

* Node.js
* NPM
* Visual Studio Code

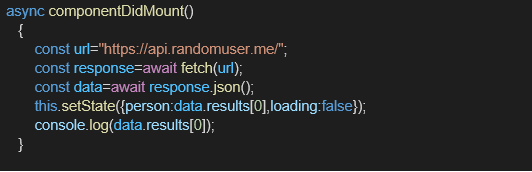
## **Notes**

Estimated time to complete this lab: **60 minutes.**

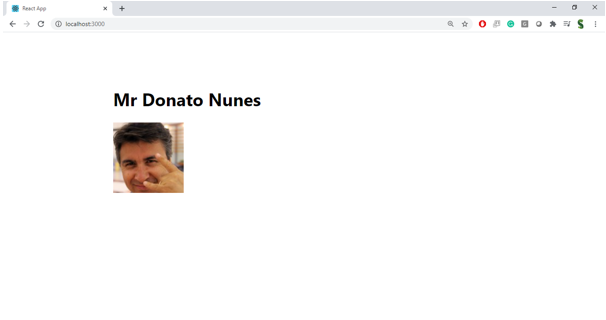
Create a React Application “fetchuserapp” which will retrieve the user details from <https://api.randomuser.me/> and display the title, firstname and image of a user.

Create a component named “Getuser” and in the asynchronous method “ComponentDidMount ()” invoke the URL using fetch method and the response can be displayed in the render method of the component.

Code Snippet in Getuser Component:



**Expected Output:**



**SOURCE CODE**

**File: Getuser.js**

import React, { Component } from 'react';

class Getuser extends Component {

// Set the initial state

state = {

person: {},

loading: true,

};

// Fetch data after the component mounts

async componentDidMount() {

const url = "<https://api.randomuser.me/>";

try {

const response = await fetch(url);

const data = await response.json();

// Update state with the fetched data

this.setState({ person: data.results[0], loading: false });

console.log(data.results[0]); // Log the data to the console for debugging

} catch (error) {

console.error("Error fetching data: ", error);

this.setState({ loading: false });

}

}

render() {

const { person, loading } = this.state;

// Render loading message if data is being fetched

if (loading) {

return <div>Loading...</div>;

}

return (

<div className="user-card">

<h2>User Details</h2>

<img src={person.picture?.large} alt="User" />

<h3>{person.name?.title} {person.name?.first}</h3>

</div>

);

}

}

export default Getuser;

**File: App.js**

import React from 'react';

import './App.css';

import Getuser from './Getuser';

function App() {

return (

<div className="App">

<h1>Fetch User App</h1>

<Getuser />

</div>

);

}

export default App;

**File: App.css**

.App {

text-align: center;

padding: 20px;

}

.user-card {

background-color: #f3f3f3;

padding: 20px;

border-radius: 8px;

display: inline-block;

margin-top: 20px;

width: 250px;

box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

}

.user-card img {

border-radius: 50%;

width: 100px;

height: 100px;

object-fit: cover;

}

h1 {

font-size: 2rem;

}

h3 {

margin-top: 10px;

}

**OUTPUT:**

