

**Lesson 6- Accessing backend services using RESTful API**

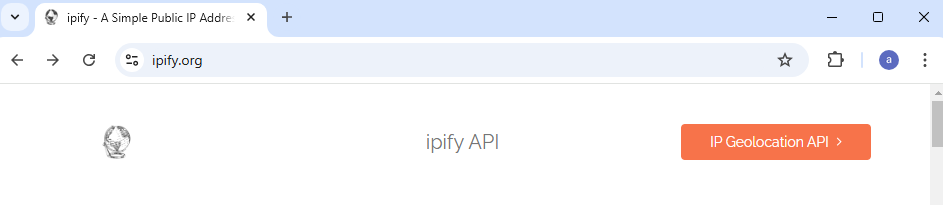
**Activity 11 –PostIT App – Location-Based Services**

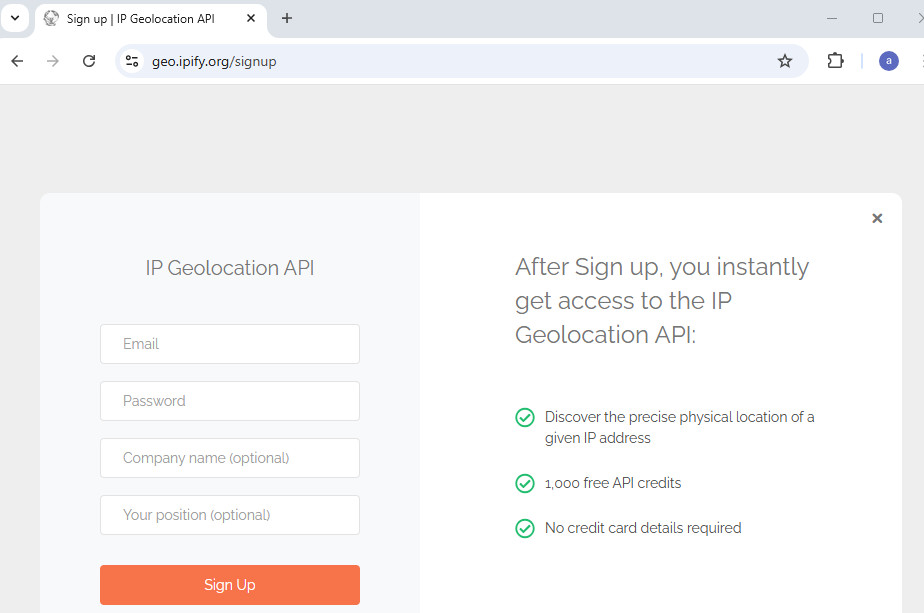
The objectives of this activity are to:

1. Implement location-based services.
2. Sign-up with ipify.org to get the API Key.
3. Write the code to determine the IP address, current region and country.

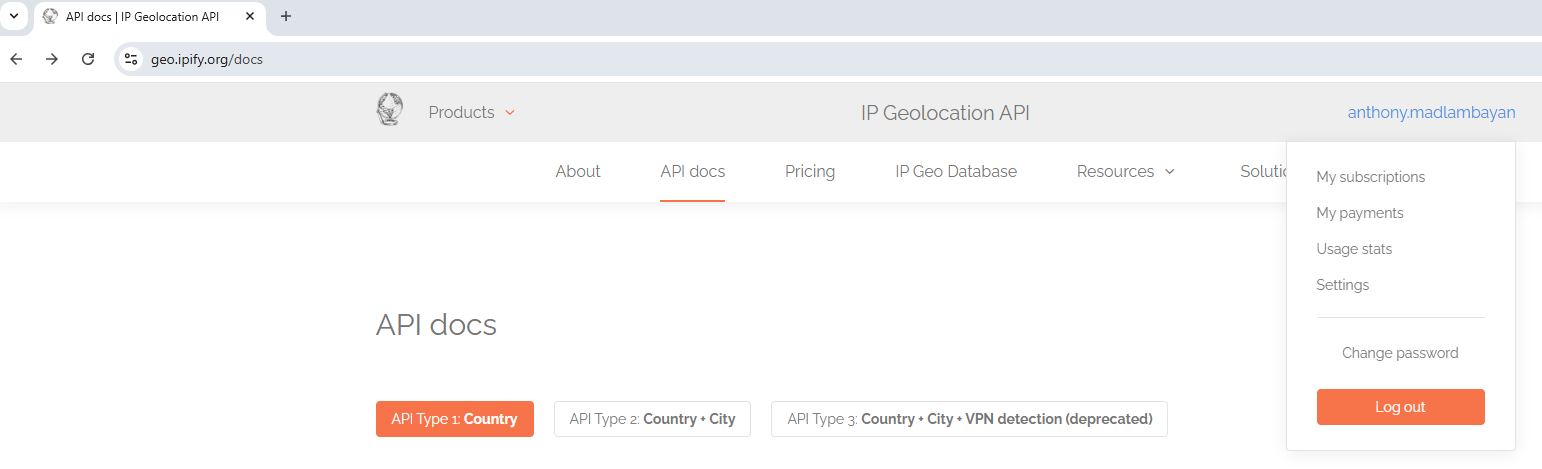
**SIGN UP WITH ipify**

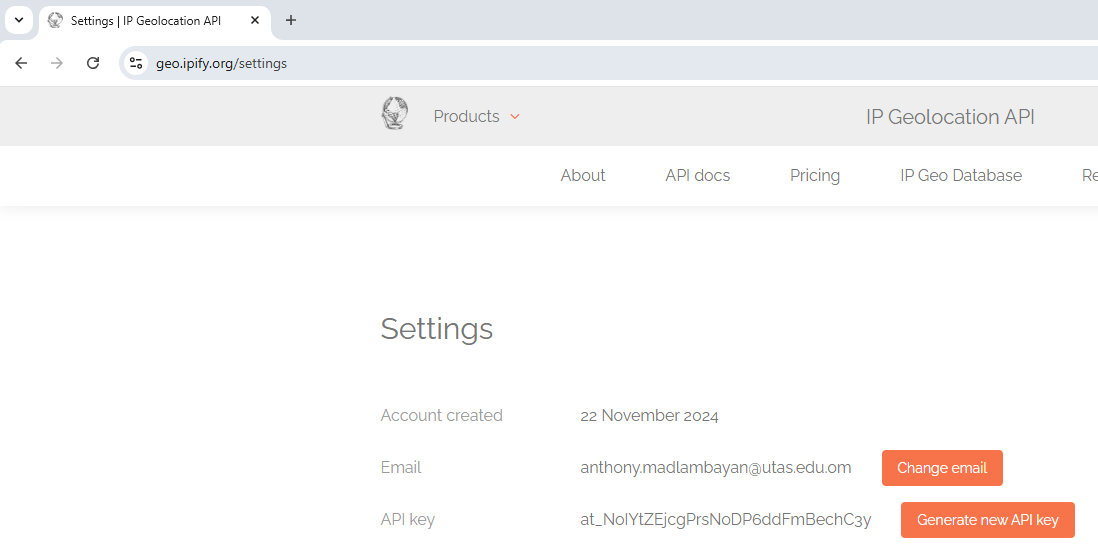
1. To implement this location-based services, you need to have API Key from ipify.org, an organization that has IP Geolocation API to get the geolocation data by IP.
2. Visit the site <https://www.ipify.org/> and click on IP Geolocation API link and signup.





1. After signing-up, you have to login and go to account settings to get your API Key.





**IMPLEMENTING THE LOCATION-BASED SERVICES**

1. In the client, create a new component **Location.js.**

const Location = () =>{

    return(

        <div>

        </div>

    );

}

export default Location;

1. Import the following:

import React, { useState, useEffect } from "react";

import axios from "axios";

1. Create state variables to hold the IP Address and geolocation data.

function Location() {

  const [ip, setIp] = useState(null); //State to hold the IP address

  const [country, setCountry] = useState(null); //State to hold geolocation

  const [region, setRegion] = useState(null); // State to hold geolocation

1. Content of **.env** file which Location API\_KEY from the <https://www.ipify.org/>

REACT\_APP\_API\_URL=http://localhost:3001

REACT\_APP\_IP\_URL= https://api.ipify.org?format=json

REACT\_APP\_LOCAION\_API\_KEY=https://geo.ipify.org/api/v2/country?apiKey=at\_oOvyKK588yW2Uq9OToo6sYvoPLVcD

1. Codes below will fetch the geolocation data based on the IP. Make sure to provide or place your API Key appropriately.

const getGeoLocationData = async () => {

    try {

      const response = await axios.get(`${process.env.REACT\_APP\_LOCAION\_API\_KEY}`);

      setIp(response.data.ip);

      setCountry(response.data.location.country); // Set country

      setRegion(response.data.location.region); // Set region

    } catch (error) {

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

    }

  };

1. The codes below will fetch the IP Address and geolocation data when the component is loaded. In the return part will display fetched the current IP address, country and region.

  useEffect(() => {

      getGeoLocationData();

  }, []);

 return (

    <div>

      <h6>{ip}</h6>

      <h6>{country}</h6>

      <h6>{region}</h6>

    </div>

  );

}

1. In the **User.js** **component**, import **Location.js** **component** and render it as shown in the code below (in red font color).

import { useSelector } from "react-redux";

import userimg from "../Images/user.png";

import Location from "./Location";

const User = () => {

 const { user } = useSelector((state) => state.users);

  return (

    <div>

      <img src={userimg} className="userImage" alt="" />

      <h6>{user?.name}</h6>

      <h6>{user?.email}</h6>

      <Location />

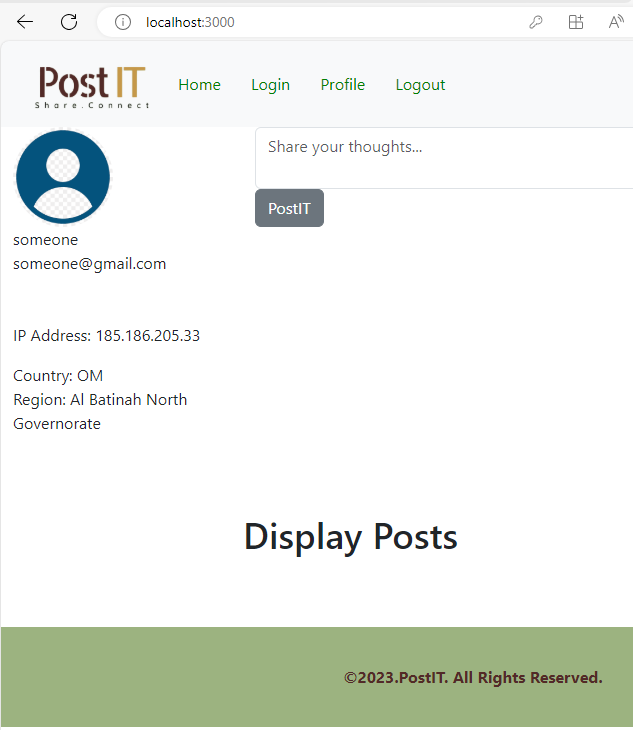
    </div>

  );

};

export default User;

1. Test your code and it should display the output below (click Home link).



*Note: The program is tested here in UTAS-Shinas that it is why it is displaying Al Batinah North Governorate as the region.*