**Front-end (ReactJS)**

--App.js file -

const [searchTerm, setSearchTerm] = useState('');

const [searchResults, setSearchResults] = useState([]);

--create an input field and a button

<input type="text" value={searchTerm} onChange={e => setSearchTerm(e.target.value)} />

<button onClick={() => searchBing(searchTerm)}>Search</button>

--searchBing function making GET request to the Bing search API

const searchBing = async (searchTerm) => {

const response = await axios.get(`https://api.bing.microsoft.com/v7.0/search?q=${searchTerm}`, {

headers: {

'Ocp-Apim-Subscription-Key': 'YOUR\_BING\_SEARCH\_API\_KEY'

}

});

setSearchResults(response.data.webPages.value);

}

--mapping over the searchResults and displaying each result as a list

<ul>

{searchResults.map((result, index) => (

<li key={index}>

<a href={result.url}>{result.name}</a>

<p>{result.snippet}</p>

</li>

))}

</ul>

import React, { useState } from 'react';

import axios from 'axios';

const App = () => {

const [searchTerm, setSearchTerm] = useState('');

const [searchResults, setSearchResults] = useState([]);

const searchBing = async (searchTerm) => {

const response = await axios.get(`https://api.bing.microsoft.com/v7.0/search?q=${searchTerm}`, {

headers: {

'Ocp-Apim-Subscription-Key': 'YOUR\_BING\_SEARCH\_API\_KEY'

}

});

setSearchResults(response.data.webPages.value);

}

return (

<div>

<input type="text" value={searchTerm} onChange={e => setSearchTerm(e.target.value)} />

<button onClick={() => searchBing(searchTerm)}>Search</button>

<ul>

{searchResults.map((result, index) => (

<li key={index}>

<a href={result.url}>{result.name}</a>

<p>{result.snippet}</p>

</li>

))}

</ul>

**</div>**

);

}

export default App;

**Backend (.Net)**

private List<Location> GetLocations()

{

return new List<Location>

{

new Location { Name = "Pharmacy", Available = IsAvailable(DateTime.Now, 9, 21) },

new Location { Name = "Bakery", Available = IsAvailable(DateTime.Now, 6, 18) },

new Location { Name = "Barber Shop", Available = IsAvailable(DateTime.Now, 10, 19) },

new Location { Name = "Supermarket", Available = IsAvailable(DateTime.Now, 8, 22) },

new Location { Name = "Candy Store", Available = IsAvailable(DateTime.Now, 11, 20) },

new Location { Name = "Cinema Complex", Available = IsAvailable(DateTime.Now, 11, 23) },

new Location { Name = "Gym", Available = IsAvailable(DateTime.Now, 6, 23) },

new Location { Name = "Coffee Shop", Available = IsAvailable(DateTime.Now, 7, 22) },

new Location { Name = "Library", Available = IsAvailable(DateTime.Now, 10, 19) },

new Location { Name = "Post Office", Available = IsAvailable(DateTime.Now, 9, 17) }

};

}

private bool IsAvailable(DateTime date, int startHour, int endHour)

{

return date.Hour >= startHour && date.Hour <= endHour;

}

public class Location

{

public string Name { get; set; }

public bool Available { get; set; }

}