**ЛАБОРАТОРНА РОБОТА № 2**

**Express, HBS. Проект “WeatherApp”**

Хід роботи

Лістинг коду:

index.js:

const express = require('express');

const hbs = require('hbs');

const axios = require('axios');

const env = require('dotenv').config();

const app = express();

const port = 3000;

app.set('view engine', 'hbs');

hbs.registerPartials(\_\_dirname + '/views/partial');

const weather\_api\_key = process.env.WEATHER\_API\_KEY;

const geolocation\_api\_key = process.env.GEOLOCATION\_API\_KEY;

async function getUserLocation() {

try {

const response = await axios.get(`https://api.maptiler.com/geolocation/ip.json?key=${geolocation\_api\_key}`);

const locationData = response.data;

console.log(locationData.city);

return locationData.city;

} catch (error) {

console.error(error);

throw new Error('Error retrieving user location');

}

}

app.get('/', (req, res) => {

res.render('index');

}

);

app.get('/weather', async (req, res) => {

try {

const city = req.query.city || await getUserLocation();

const response = await axios.get(`https://api.openweathermap.org/data/2.5/weather?q=${city}&units=metric&appid=${weather\_api\_key}`);

const weatherData = response.data;

const sunrise = new Date(weatherData.sys.sunrise \* 1000);

const sunriseHours = sunrise.getHours();

const sunriseMinutes = sunrise.getMinutes();

const sunset = new Date(weatherData.sys.sunset \* 1000);

const sunsetHours = sunset.getHours();

const sunsetMinutes = sunset.getMinutes();

weatherData.sunrise = `${sunriseHours}:${sunriseMinutes}`;

weatherData.sunset = `${sunsetHours}:${sunsetMinutes}`;

res.render('weather', { weather: weatherData });

} catch (error) {

console.error(error);

res.status(500).send('Error retrieving weather data');

}

});

app.listen(port, () => {

console.log(`Example app listening at http://localhost:${port}`)

}

);

index.hbs:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">

<title>Weather App</title>

</head>

<body>

{{> header}}

<div class="container d-flex align-items-center flex-column">

<h1>Welcome to Weather App</h1>

<a href="/weather"><button class="btn btn-primary">Check weather</button></a>

</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz" crossorigin="anonymous"></script>

</body>

</html>

weather.hbs:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Weather in {{weather.name}}</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">

<link rel="icon" href="http://openweathermap.org/img/w/{{weather.weather.[0].icon}}.png">

</head>

<body>

{{> header}}

<div class="container">

<h1>Weather Data</h1>

<form action="/weather" method="GET">

<div class="input-group mb-3">

<input type="text" class="form-control" placeholder="Enter city name" value="{{weather.name}}" name="city">

<button class="btn btn-primary" type="submit">Search</button>

</div>

</form>

<div class="row">

<div class="col-md-6">

<div class="card">

<div class="card-body">

<h2>{{weather.name}}</h2>

<div class="weather-info">

<div class="weather-icon">

<img src="http://openweathermap.org/img/w/{{weather.weather.[0].icon}}.png" alt="">

</div>

<div class="weather-details">

<p>Temperature: {{weather.main.temp}}℃</p>

<p>Humidity: {{weather.main.humidity}}%</p>

<p>Wind Speed: {{weather.wind.speed}}m/s</p>

<p>Conditions: {{weather.weather.[0].description}}</p>

</div>

</div>

</div>

</div>

</div>

<div class="col-md-6">

<div class="card">

<div class="card-body">

<h2>Full Weather Info</h2>

<p>Temperature Min: {{weather.main.temp\_min}}℃</p>

<p>Temperature Max: {{weather.main.temp\_max}}℃</p>

<p>Pressure: {{weather.main.pressure}}hPa</p>

<p>Visibility: {{weather.visibility}}m</p>

<p>Sunrise: {{weather.sunrise}}</p>

<p>Sunset: {{weather.sunset}}</p>

</div>

</div>

</div>

</div>

</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz" crossorigin="anonymous"></script>

</body>

</html>

header.hbs:

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container">

<a class="navbar-brand" href="/">Weather App</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNav">

<ul class="navbar-nav ml-auto">

<li class="nav-item">

<a class="nav-link" href="/">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="/weather">Weather</a>

</li>

</ul>

</div>

</div>

</nav>

Результат виконання програми:

Головна сторінка:

A screenshot of a computer

Description automatically generated

Рис.1. Головна сторінка програми

Сторінка погоди:

Погода підтягується за допомогою Maptiler API, яке дозволяє визначити місцеположення користувача по IP-адресі

Також використовується OpenWeatherMap API, для визначення погоди для будь якого міста, або населеного пункту

A screenshot of a computer

Description automatically generated

Рис.2. Сторінка погоди

Таким чином якщо просто відкрити сторінку з погодою, автоматично буде визначена погода вашого міста, без вказування цього в посиланні на сторінку:



Рис.3. Посилання на сторінку погоди

Але якщо вписати місто в поле для тексту і натиснути “Search”, посилання буде замінена на потрібне місто.

A screenshot of a computer

Description automatically generated

Рис.4. Погода для окремого міста

A close-up of a sign

Description automatically generated

Рис.5. Посилання на погоду окремого міста

Отримати API-ключі для тестування можна тут:

OpenWeatherMap - <https://openweathermap.org/api>

Maptiler API - <https://cloud.maptiler.com/auth/widget?next=https://cloud.maptiler.com/account/keys/>