

Tech Task Symfony Developer

Нижче наведено PHP-скрипт, який отримує дані про погоду із зовнішнього API. Ваше завдання - рефакторинг цього коду в повноцінний Symfony-додаток, дотримуючись принципів об'єктно-орієнтованого програмування та кращих практик Symfony.

```
<?php

function getWeatherData($city) {
    $apiKey = 'abc123weatherapikey';
    $url =
    "https://api.weatherapi.com/v1/current.json?key={$apiKey}&q={$city}";

    $ch = curl_init();
    curl_setopt($ch, CURLOPT_URL, $url);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($ch, CURLOPT_TIMEOUT, 30);
    $response = curl_exec($ch);

    if(curl_errno($ch)) {
        $error = 'Curl error: ' . curl_error($ch);
        curl_close($ch);
        return ['error' => $error];
    }

    curl_close($ch);
    $data = json_decode($response, true);

    if(isset($data['error'])) {
        return ['error' => $data['error']['message']];
    }

    $result = [
        'city' => $data['location']['name'],
        'country' => $data['location']['country'],
        'temperature' => $data['current']['temp_c'],
        'condition' => $data['current']['condition']['text'],
        'humidity' => $data['current']['humidity'],
        'wind_speed' => $data['current']['wind_kph'],
    ];
}
```

```

        'last_updated' => $data['current']['last_updated'],
    ];

    file_put_contents('weather_log.txt', date('Y-m-d H:i:s') . " - Погода в
{$result['city']}: {$result['temperature']}°C, {$result['condition']}\n",
FILE_APPEND);

    return $result;
}

$weatherData = getWeatherData('London');
if(isset($weatherData['error'])) {
    echo "Ошибка: " . $weatherData['error'];
} else {
    echo "Текущая погода в {$weatherData['city']},
{$weatherData['country']}: \n";
    echo "Температура: {$weatherData['temperature']}°C\n";
    echo "Состояние: {$weatherData['condition']}\n";
    echo "Влажность: {$weatherData['humidity']}%\n";
    echo "Скорость ветра: {$weatherData['wind_speed']} км/ч\n";
    echo "Последнее обновление: {$weatherData['last_updated']}\n";
}

```

Вимоги:

1. Створити програму на Symfony (використовувати останню LTS версію)
2. Реалізувати сервіс Weather для обробки комунікації з API
3. Створити контролер для надання даних про погоду через ендпоінт
4. Впровадити правильне керування конфігурацією для API-ключа
5. Додати базову обробку помилок та логування
6. Створити простий Twig-шаблон для відображення інформації про погоду
7. Використовувати використання залежностей там, де це доречно

8. Написати хоча б один модульний тест для вашого сервісу
9. Критерії оцінки
Організація та структура коду
10. Правильне використання компонентів та сервісів Symfony
11. Обробка помилок та керування винятками
12. Управління конфігурацією
13. Поділ відповідальності
14. Якість коду та дотримання кращих практик
15. Документація та коментарі

Вимоги до задачі

Будь ласка, надайте:

1. Повний вихідний код
2. Інструкції з встановлення та налаштування
3. Коротке пояснення ваших рішень щодо реалізації

Успіхів!