

## **Задача: Система за абонамент за прогнози за времето с имейл известия**

**Описание:** Създайте система, която позволява на потребителите да се абонират за прогнози за времето за определени локации. Потребителите трябва да могат да получават имейл известие с дневна прогноза за времето за съответната локация.

Системата трябва да включва интеграция с външен API за прогноза на времето и да предоставя механизъм за абониране, както и изпращане на известия за актуалната прогноза.

---

### **Изисквания:**

#### **1. Абонаменти:**

- Потребителите трябва да могат да се абонират за дневни прогнози за времето за различни локации, като предоставят имейл адрес.
- Всеки абонамент трябва да е свързан с една локация (например град или координати).
- Всеки абонамент трябва да включва активен или неактивен статус.

#### **2. Интеграция с API:**

- Използвайте външен API за извличане на прогнози за времето (например OpenWeatherMap или [WeatherStack](#)).
- Получавайте и използвайте прогнози за времето за съответната локация (основни данни като температура, вятър, валежи и т.н.).

#### **3. Известия:**

- Системата трябва да изпраща имейл с дневни известия на потребителите с прогнозата за времето на избраната от тях локация.
- Имейлите трябва да съдържат информация за текущото време, температурата, възможните валежи и други ключови данни.

#### **4. Периодични задачи:**

- Трябва да се създадат периодични задачи, които ще изпращат имейли с прогнозите за времето на потребителите.
- Тези задачи трябва да се стартират на дневна база и да изпращат прогноза за съответния ден.

#### **5. Управление на абонаменти:**

- Потребителите трябва да могат да се абонират за прогнози или да се отписват от тях.
  - Когато потребителят получи имейл с прогнозата, трябва да има линк за деактивиране на абонамента (отписване).
  - Абонаментите трябва да могат да бъдат активирани чрез API
- 

### **Забележка:**

Оставям на вас да решите как най-добре да структурирате архитектурата на приложението, как да организирате базата данни, да управлявате зависимостите и да имплементирате логиката за обработка на API заявките. Важното е да се създаде ефективен, мащабируем и лесен за поддръжка код, който отговаря на горните изисквания.

Задачата може да бъде реализирана главно с бекенд логика и REST API