**Тестовое задание на позицию «Backend developer C#»**

Необходимо создать сервис для HR по процессу отбора кандидатов.  
  
**Минимальные сущности(расширяйте структуру по своему усмотрению):**

***Рабочая группа:***

Имя

Список участников

***Пользователь:***

Роль(Админ, HR)

ФИО

Логин

Пароль

Рабочая группа

***Данные кандидата:***

Имя

Фамилия

Отчество

Email

Телефон

Список социальных сетей

Страна

Дата рождения

***Кандидат:***

Данные кандидата

Дата последнего обновления

Рабочий график(Офис, Гибрид, Удаленка)

Рабочая группа

***Социальная сеть:***

Имя пользователя

Тип

Дата добавления

***Сотрудник:***

Данные кандидата

Дата найма

***Проверка:***

Кто провел

Дата

Данные кандидата

Список ивентов

***Ивент проверки:***

Тип(Список кандидатов/Сотрудники)

Результат(список Id кандидатов/сотрудников)  
  
**Требования к БД:**

1. БД MySQL/PostgreSQL

2. EF Core code first

3. Предусмотреть развертывание БД с готовыми тестовыми данными посредством миграций(аккаунты администратора, и 2 HR в разных рабочих группах, несколько кандидатов)

**Требования к серверному приложению**

1. Реализовать Web-Api Asp.Net Core
2. С# .NET 8
3. Реализация HealthChecks
4. Реализация Swagger с описанием методов
5. Работа с JWT токенами авторизации

**Пользователи, авторизация и роли:**1. Методы аутентификации и обновления токена. Метод аутентификации должен возвращать структуру с access и refresh токенами для использования в авторизационном атрибуте в остальных контроллерах.

1. Вся работа с кандидатам ограничена рабочей группой(другие рабочие группы не видят чужих кандидатов и не могут с ними взаимодействовать)  
   3. Роли:  
   1) Админ может создавать HR, создавать рабочие группы и редактировать их состав  
   2) HR может работать с кандидатами внутри своей рабочей группы

**Требования к функционалу:**

1. Метод создания пользователя с ролью HR

2. Метод создания рабочей группы

3. Метод редактирования списка участников рабочей группы(при перемещении HR в другую рабочую группу предусмотреть миграцию его кандидатов)

4. Метод создания кандидата

5. Метод редактирования кандидата(редактируются все поля кроме даты последнего обновления)

6. Метод получения списка кандидатов с пагинацией и следующими необязательными фильтрами:

1) Search: Поиск по ФИО/Имейлу/Имени пользователя любой из социальных сетей

2) Сортировка по Дате последнего обновления

3) Фильтр по рабочему графику (мультивыбор)

4) Фильтр "Только мои", true - отображать только кандидатов, добавленных мною

7. Метод перевода кандидата в сотрудника

1) Создать сотрудника -> привязать данные кандидата -> Удалить кандидата

8. Метод проверки кандидата по ФИО. На выход должен быть отдан id проверки, без ожидания завершения поиска, проверки запускать параллельно друг другу, поиски в проверках проводятся без учета рабочих групп.

1) Запустить поиск по списку кандидатов

2) Запустить поиск по списку сотрудников

9. Метод получения списка результатов по id проверки(на выход отдать ФИО по кому искали и список найденных id)

**Результаты:**

Готовый проект опубликовать на Github и приложить ссылку. Срок выполнения 2 недели с момента получения.