

# Тестовое задание – Events Web-application

Вступительная задача для стажировки по направлению .NET

## CONTENTS

1.	ОПИСАНИЕ .....	3
1.1.	ФУНКЦИОНАЛ WEB-ПРИЛОЖЕНИЯ .....	3
1.2.	ИНФОРМАЦИЯ О СОБЫТИИ.....	4
1.3.	ИНФОРМАЦИЯ ОБ УЧАСТНИКЕ СОБЫТИЯ .....	4
1.4.	РАЗРАБОТКА КЛИЕНТСКОЙ ЧАСТИ ПРИЛОЖЕНИЯ .....	4
1.5.	ТРЕБОВАНИЯ К WEB API .....	5
1.6.	НЕОБХОДИМЫЕ К ИСПОЛЬЗОВАНИЮ ТЕХНОЛОГИИ .....	5
1.7.	АРХИТЕКТУРА.....	5

## 1. Описание

Разработка web-приложения для работы с событиями, выполняется на .Net Core, с использованием EF Core. Для разработки клиентской части приложения следует использовать Angular или React.

Должна прилагаться инструкция по запуску проекта.

### 1.1. Функционал Web API

Работа с событиями:

1. Получение списка всех событий;
2. Получение определенного события по его Id;
3. Получение события по его названию;
4. Добавление нового события;
5. Изменение информации о существующем событии;
6. Удаление события;
7. Получение списка событий по определенным критериям (по дате, месту проведения, категории события)
8. Возможность добавления изображений к событиям и их хранение.

Работа с участниками:

1. Регистрация участия пользователя в событии;
2. Получение списка участников события;
3. Получение определенного участника по его Id;
4. Отмена участия пользователя в событии;
5. \*Отправка уведомлений участникам события о изменениях в событии (например, об изменении даты или места проведения) (будет плюсом);

## 1.2. Информация о событии

1. Название;
2. Описание;
3. Дата и время проведения;
4. Место проведения;
5. Категория события;
6. Максимальное количество участников;
7. Список участников;
8. Изображение.

## 1.3. Информация об участнике события

1. Имя;
2. Фамилия;
3. Дата рождения;
3. Дата регистрации на событие;
4. Email.

## 1.4. Разработка клиентской части приложения

1. Реализация страницы аутентификации/регистрации пользователей;
2. Реализация страницы отображения списка событий. Если все места на событие уже заняты, отображать информацию о том, что свободных мест нет;
3. Реализация страницы с подробной информацией о событии;
4. Реализация страницы регистрации на событие;
5. Реализация страницы для просмотра событий, участником которых пользователь является;
6. Реализация пагинации, поиска по названию/дате события, фильтрации по категории/месту проведения;
7. Для пользователей, обладающих правами администратора, должен быть предусмотрен раздел для управления списком событий, где они могут создавать, редактировать и удалять события.

### 1.5. Требования к Web API

1. Реализация policy-based авторизации с использованием refresh и jwt access токенов;
2. Внедрение паттерна репозиторий и Unit of Work;
3. Разработка middleware для глобальной обработки исключений;
4. Реализация пагинации;
5. Обеспечение покрытия unit-тестами всех юзкейсов одной модели и двух методов ее репозитория (один на получение и один на сохранение данных). Модель выбирается на усмотрение кандидата (для всех моделей должны быть написаны use cases). Разрешено использовать InMemoryDatabase;
6. \*Внедрение кеширования изображений (будет плюсом).

### 1.6. Необходимые к использованию технологии

1. .Net 5.0+;
2. Entity Framework Core;
3. MS SQL / PostgreSQL or any other;
4. AutoMapper / Mapster or any other;
5. FluentValidation;
6. Authentication via jwt access & refresh token (ex.: IdentityServer4);
7. Swagger;
8. EF Fluent API;
9. Angular/React;
10. xUnit/nUnit.

### 1.7. Архитектура

Приложение должно быть разработано на чистой архитектуре.

**Перед выполнением задачи- сообщите HR сколько потребуется времени на всю реализацию (в часах)!!!**