**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ   
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Департамент программной инженерии

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Кандидат технических наук, приглашенный преподаватель департамента программной инженерии факультета компьютерных наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ О. В. Максименкова  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2020 г. | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Шилов  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_2020 г. |
| |  |  | | --- | --- | | ***Подп. и дата*** |  | | ***Инв. № дубл.*** |  | | ***Взам. Инв. №*** |  | | ***Подп. и дата*** |  | | ***Инв. № подл.*** | RU.17701729.04.0 1-01 81 01-1-ЛУ | | **Веб-приложение для волонтеров с элементами геймификации**  **Пояснительная записка**  **ЛИСТ УТВЕРЖДЕНИЯ**  **RU.17701729.04.01-01 81 01-1-ЛУ**  Исполнитель  студент группы БПИ 173  \_\_\_\_\_\_\_\_\_\_\_/Королев Д. П./  «\_\_\_»\_\_\_\_\_\_\_\_\_\_2020 г.  **2020** | | |  |

УТВЕРЖДЕНО

RU.17701729.04.01-01 81 01-1-ЛУ

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | ***Подп. и дата*** |  | | ***Инв. № дубл.*** |  | | ***Взам. Инв. №*** |  | | ***Подп. и дата*** |  | | ***Инв. № подл.*** | RU.17701729.04.0 1-01 81 01-1 | | **Веб-приложение для волонтеров с элементами геймификации**  Пояснительная записка  RU.17701729.04.01-01 81 01-1  **Листов 25**  **2020** |  |

**СОДЕРЖАНИЕ**

[1. Введение 4](#_Toc36949205)

[1.1. Наименование программы 4](#_Toc36949206)

[1.2. Документы, на основании которых ведется разработка 4](#_Toc36949207)

[2. Назначение и область применения 5](#_Toc36949208)

[2.1. Функциональное назначение 5](#_Toc36949209)

[2.2. Эксплуатационное назначение 5](#_Toc36949210)

[2.3. Краткая характеристика области применения 6](#_Toc36949211)

[3. Технические характеристики 7](#_Toc36949212)

[3.1. Постановка задачи на разработку программы 7](#_Toc36949213)

[3.2. Описание алгоритма и функционирования программы 8](#_Toc36949214)

[3.2.1 База данных приложения 8](#_Toc36949215)

[3.2.2 Устройство приложения и серверная логика 9](#_Toc36949216)

[3.2.3 Веб-верстка и страничные представления 10](#_Toc36949217)

[3.2.4 Хостинг. Публикация и развертывание приложения 11](#_Toc36949218)

[3.2.4 Локализация приложения 11](#_Toc36949219)

[3.3. Описание и обоснование выбора метода организации входных и выходных данных 12](#_Toc36949220)

[3.4. Описание и обоснование выбора состава технических и программных средств 12](#_Toc36949221)

[3.4.1. Описание выбора состава технических средств 12](#_Toc36949222)

[3.4.2. Обоснование выбора состава технических средств 13](#_Toc36949223)

[3.4.3. Описание выбора состава программных средств 13](#_Toc36949224)

[3.4.4. Обоснование выбора состава программных средств 13](#_Toc36949225)

[4. Ожидаемые технико-экономические показатели 14](#_Toc36949226)

[4.1. Предполагаемая востребованность 14](#_Toc36949227)

[4.2. Экономические преимущества разработки по сравнению с аналогами 14](#_Toc36949228)

[5. Источники, использованные при разработке 15](#_Toc36949229)

[6. Приложения 17](#_Toc36949230)

[6.1. Описание и функциональное назначение классов 17](#_Toc36949231)

[6.2. Описание и функциональное назначение полей, методов и свойств 17](#_Toc36949232)

# 1. Введение

## 1.1. Наименование программы

Наименование программы – «Веб-приложение для волонтеров с элементами геймификации».

## 1.2. Документы, на основании которых ведется разработка

Основанием для разработки является приказ декана факультета компьютерных наук Национального исследовательского университета «Высшая школа экономики» № 2.3-02/1012-02 от 10.12.18 «Об утверждении тем, руководителей курсовых работ студентов образовательной программы Программная инженерия факультета компьютерных наук».

Разработка программы проводится в рамках выполнения курсовой работы по теме «Веб-приложение для волонтеров с элементами геймификации»

# 2. Назначение и область применения

## 2.1. Функциональное назначение

Назначением программы является предоставление удобной для использования платформы для волонтеров с элементами геймификации. Приложение представляет из себя веб-сайт, к которому предоставляется доступ по сети Интернет.

В рамках приложения каждому пользователю доступен личный кабинет. С помощью регистрации в приложении система может распознавать и идентифицировать пользователей, которые регистрируются и принимают участие в волонтерском мероприятии.

Для того, чтобы пользователь мог наблюдать за своим прогрессом, а также организаторы (менеджеры) мероприятий могли следить и привлекать пользователей, в системе поддержана геймификация, которая достигается за счет внутри-сервисной валюты, имеющей ценность только внутри приложения.

В качестве меры для утверждения принятия волонтером участия в мероприятии решено применять уникальный QR-код [15] или токен [1], который генерируется для каждого пользователя (см. п. 3.2 текущего документа).

## 2.2. Эксплуатационное назначение

Программа предназначена для пользователей, которые хотят принять участие в различных общественных мероприятиях (для волонтеров), или для пользователей, которые модерируют или организуют эти мероприятия

## 2.3. Краткая характеристика области применения

Данное приложение может быть полезно следующим пользователями:

1. Волонтер, ищет ресурс, где располагается информация о проводимых мероприятиях некоторыми организациями (теми, которые зарегистрированы в том же приложении).
2. Человек, который не имеет достаточного уровня мотивации для того, чтобы принимать участие в волонтерском мероприятии. Такого человека может заинтересовать бонусная система, поддерживаемая системой.
3. Представитель организации, которая проводит волонтерские мероприятия. Такой пользователь имеет расширенные полномочия, относительно обычного пользователя в системе, и может создавать волонтерские мероприятия от лица организации.

# 3. Технические характеристики

## 3.1. Постановка задачи на разработку программы

Функции, указанные ниже, должны быть реализованы в программе:

1. Возможность просматривать и переходить с одной страницы приложения на другую посредством нажатия на соответствующие кнопки в меню, которое располагается на каждой из страниц веб-сайта.
2. Регистрация и аутентификация пользователя в приложении.
3. У пользователя должна быть возможность просматривать информацию об организациях, зарегистрированных в приложении. У менеджера – просматривать, создавать и редактировать информацию об организациях.
4. У пользователя должна быть возможность просматривать информацию о мероприятиях, проводить регистрацию на принятие участия и так же ее отменять. У менеджера должны быть возможность создавать мероприятия, редактировать и просматривать их информацию.
5. У менеджера должна быть возможность активировать и деактивировать принятие волонтером участия в приложении путем сканирования QR-кода или с помощью ввода токена в соответствующее поле. QR-код и токен представляются менеджеру пользователем с своего устройства.
6. У администратора должна быть возможность просматривать зарегистрированных пользователей приложения, удалять их, а также изменять их роли.

## 3.2. Описание алгоритма и функционирования программы

Веб-сервис представляет из себя ASP.NET-серверное приложение [2], которое имеет статический IP-адрес, к которому пользователь обращается для отображения страниц к его браузере.

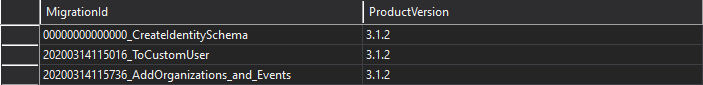
### 3.2.1 База данных приложения

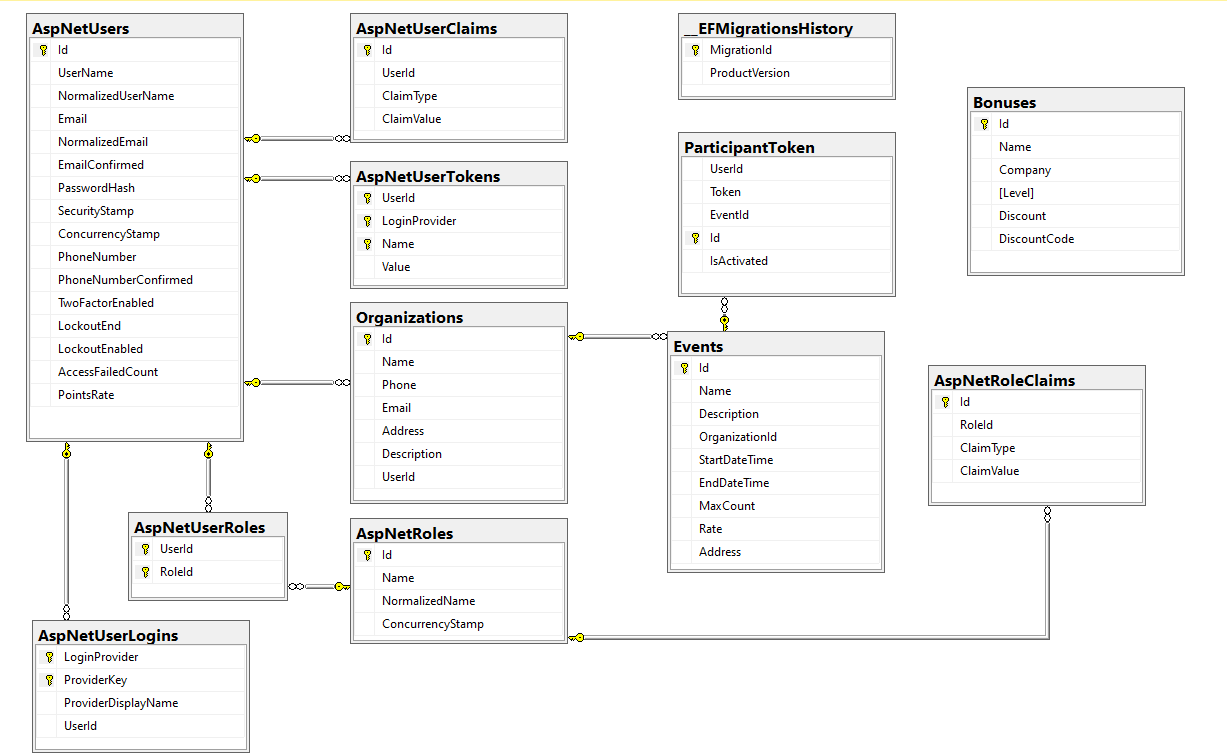
Приложение оперирует личными данными пользователей, которые хранятся в базе данных. В данном приложении была выбрана база данных MS SQL [3]. В качестве библиотеки для написания логики запросов к базе данных была выбрана Entity Framework Core [4]. Все обращения к базе данных асинхронны для оптимизации.

Для проектирования архитектуры базы данных принята методология Model-First [5], позволяющая генерировать таблицы базы данных из моделей и атрибутов свойств у моделей, описанных на языке C#. Ниже я приведу итоговую схему базы данных (рис 2). База спроектирована на основе реляционной теории.

Для управления учетными записями и поддержки регистрации и аутентификации пользователей в приложение было интегрировано решение ASP.NET Identity [6], таблицы которой можно наблюдать в схеме базы данных.

Проектирование и изменение базы данных поддерживалось миграциями. Таблица с историями миграций включена в базу данных. Это решение позволяет вернуться к прежней схеме базы данных по имени миграции (рис 1).

  
*рис 1. – Фрагмент таблицы с миграциями базы данных*

  
*рис 2. – Схема базы данных приложения*

### 3.2.2 Устройство приложения и серверная логика

Для каждой части приложения создан собственный проект. Все части приложения поддерживают версию .NET Core 3.1. Приложение (имеет название ‘Volunteers’ в среде разработки Visual Studio 2019) разделено на 4 проекта:

* Volunteers.Common – библиотека для определения моделей и основных классов, которыми будет оперировать приложение
* Volunteers.Core – библиотека для определения бизнес-логики приложения
* Volunteers.DataAccessLibrary – библиотека, предоставляющая набор методов для оперирования базой данных.
* Volunteers.Web – ASP.NET приложение-проект веб-сайта.

Все запросы к приложению происходят по методологии REST API [7]. Для каждого пользовательского запроса создается собственный контроллер (класс), обрабатывающий запрос и возвращающий ответ. Для создания контроллера и передачи ему требуемых объектов (сервисов) используется Dependency Injection [8] - процесс предоставления внешней зависимости. В качестве библиотеки для данной возможности используется библиотека Microsoft.Extensions.DependencyInjection. Все запросы, отправляемые пользователем, поддерживают асинхронные операции. Таким образом достигается оптимизация работы приложения.

Для генерации токена участия пользователя в мероприятии используется генерация статически уникального 128-битного идентификатора GUID [16]. Токен в таком виде предоставляется пользователю, а QR-код генерируется на основе этого значения токена.

### 3.2.3 Веб-верстка и страничные представления

Проект Volunteers.Web представляет из себя ASP.NET Razor Pages-приложение, которое устроено следующим образом: для каждой веб-страницы создан собственный класс на языке C#, который описывает методы REST API, и соответствующее графическое представление. Для определения интерфейса выбран язык разметки HTML, формальный язык стилей CSS, а также Razor [9] – синтаксис разметки для генерации HTML-представления вместе с C#-объектами.

Так как разрабатываемый продукт является веб-приложением, то для поддержки мобильных устройств выбрана библиотека Bootstrap 4, позволяющая отображать и менять графическое представление страниц в зависимости от используемого устройства.

### 3.2.4 Хостинг. Публикация и развертывание приложения

Для хостинга приложения используется платформа Azure [11], которая предоставляет бесплатное использование студентам факультета Компьютерных Наук ВШЭ. Эта платформа предоставляет виртуальную машину, на которой запущено приложение. Сервер расположен на севере Швейцарии.

Для публикации и развертывании приложения произведена интеграция с Github [12] и Azure DevOps [13]. Github – репозиторий для хранения и версионирования кода приложения. Azure DevOps – платформа для поддержки проекта, а также настройки публикации и развертке приложения. При внесении любого изменения в основной ветке репозитория на Github, Azure DevOps выкачивает последнюю версию кода приложения, собирает проект и выкладывает его на сервере.

Для того, чтобы приложение было надежным, в приложение также добавлен проект с Unit-тестами [14] приложения. При сборке приложения в Azure DevOps при непрохождении всех тестов выдается ошибка, и приложение не выкладывается на сервер. Таким образом поддерживается надежность и бесперебойность работы приложения.

### 3.2.4 Локализация приложения

Для поддержки локализации для каждой .html-страницы сделан соответствующий ресурс с тем же наименованием, что и страница, а также в наименовании находится кодировка культуры, которую этот ресурсный файл описывает. В модель каждой страницы добавляется сервис, позволяющий по наименованию ключа обратиться к ресурсному файлу и получить оттуда строковое значение, которое нужно отобразить. В путь запроса к странице встраивается культура. Например, шаблоном пути до стартовой страницы с наличием описания языка: https://volunteers.com/{culture}/Index.

## 3.3. Описание и обоснование выбора метода организации входных и выходных данных

Формат входных данных ограничен браузером, через которых происходит взаимодействие с веб-сайтом, а также полями ввода и кнопками на страницах приложения.

Формат выходных данных – отображаемые сообщения и значения полей на страницах веб-сайта.

## 3.4. Описание и обоснование выбора состава технических и программных средств

### 3.4.1. Описание выбора состава технических средств

Минимальные и рекомендуемые свойства компьютера:

1. Процессор Intel Core i3 с частотой 2000 МГц или более быстрый (рекомендуется не менее 3000 МГц) с подходящей материнской платой.
2. Не менее 1 ГБ оперативной памяти (рекомендуется не менее 4 ГБ).
3. Не менее 1 ГБ свободного места на жестком диске.
4. Периферия для персонального компьютера, а также клавиатура.
5. Видеокарта и монитор, способные воспроизводить графическое отображение работы программы с разрешением не менее 800х600 точек.
6. Современный браузер с версией, которая выходила не позднее 2015 года.

### 3.4.2. Обоснование выбора состава технических средств

Выбор состава технических средств обусловлен использованием ОС Microsoft Windows X и новее, а также комфортными условиями для быстродействия работы программы.

### 3.4.3. Описание выбора состава программных средств

Для запуска и работы программы требуется компьютер, со следующим предустановленным программным обеспечением:

1. Современный браузер с версией, которая выходила не позднее 2015 года.

### 3.4.4. Обоснование выбора состава программных средств

Год выпуска браузера обусловлен рекомендациями используемых средств в разработке.

# 4. Ожидаемые технико-экономические показатели

## 4.1. Предполагаемая востребованность

Программа может применяться для коммерческого и некоммерческого использования в рамках принятия участия в волонтерских мероприятиях и их организации.

## 4.2. Экономические преимущества разработки по сравнению с аналогами

В сети Интернет не было найдено аналогичной программы, которая предоставляет графический интерфейс для волонтеров и представителей организаций, которые создают мероприятия.

У данного приложения есть следующие преимущества:

1. Эксплуатация программы не требует затрат денежных средств (распространяется бесплатно).
2. Программа не подразумевает обслуживания.

# 5. Источники, использованные при разработке

1. Токен [Электронный ресурс] / Режим доступа: https://ru.wikipedia.org/wiki/Токен, свободный (дата обращения 28.03.20)

2. ASP.NET [Электронный ресурс] / Режим доступа: https://dotnet.microsoft.com/apps/aspnet, свободный (дата обращения 28.03.20)

3. MS SQL. [Электронный ресурс] / Режим доступа: https://www.microsoft.com/ru-ru/sql-server/sql-server-2019, свободный (дата обращения 28.03.20)

4. Entity Framework Core. [Электронный ресурс] / Режим доступа: https://docs.microsoft.com/ru-ru/ef/core/, свободный (дата обращения 28.03.20)

5. Model-First. [Электронный ресурс] / Режим доступа: https://docs.microsoft.com/ru-ru/ef/ef6/modeling/designer/workflows/model-first, свободный (дата обращения 28.03.20)

6. Asp.Net Identity. [Электронный ресурс] / Режим доступа: https://docs.microsoft.com/en-us/aspnet/core/security/authentication/identity?view=aspnetcore-3.1&tabs=visual-studio, свободный (дата обращения 28.03.20)

7. REST API. [Электронный ресурс] / Режим доступа: https://ru.wikipedia.org/wiki/REST, свободный (дата обращения 28.03.20)

8. Dependency Injection. [Электронный ресурс] / Режим доступа: https://docs.microsoft.com/ru-ru/aspnet/core/fundamentals/dependency-injection?view=aspnetcore-3.1, свободный (дата обращения 28.03.20)

9. Razor. [Электронный ресурс] / Режим доступа: https://docs.microsoft.com/ru-ru/aspnet/core/mvc/views/razor?view=aspnetcore-3.1, свободный (дата обращения 28.03.20)

10. Bootstrap 4. [Электронный ресурс] / Режим доступа: https://bootstrap-4.ru/, свободный (дата обращения 28.03.20)

11.Azure. [Электронный ресурс] / Режим доступа: https://azure.microsoft.com/ru-ru/, свободный (дата обращения 28.03.20)

12. Github. [Электронный ресурс] / Режим доступа: https://ru.wikipedia.org/wiki/GitHub, свободный (дата обращения 28.03.20)

13. Azure DevOps. [Электронный ресурс] / Режим доступа: https://azure.microsoft.com/ru-ru/services/devops/, свободный (дата обращения 28.03.20)

14. Unit-тесты. [Электронный ресурс] / Режим доступа: https://ru.wikipedia.org/wiki/Модульное\_тестирование, свободный (дата обращения 28.03.20)

15. QR-код. [Электронный ресурс] / Режим доступа: https://ru.wikipedia.org/wiki/QR-код, свободный (дата обращения 28.03.20)

16. GUID. [Электронный ресурс] / Режим доступа: https://ru.wikipedia.org/wiki/GUID, свободный (дата обращения 28.03.20)

# 6. Приложения

## 6.1. Описание и функциональное назначение классов

Таблица 1. Название классов и их функциональное назначение (описание).

|  |  |
| --- | --- |
| Класс | Назначение |
| CodeGenerator | Класс для генерации различных значений. |
| LevelCounter | Класс для расчетов значений уровня пользователя в зависимости от значений внутри-сервисной валюты. |
| DbSeeder | Класс для первоначального внесения объектов в базу данных. |
| DbBaseService | Абстрактный класс, определяющий основные операции взаимодействия с базой данных. |
| BonusService | Класс-сервис для операций с базой данных, связанных с бонусами. |
| AppDbContext | Класс-контекст связи с базой данных |
| Organization | Класс-определение организации |
| ParticipantToken | Класс-определение токена участия в мероприятии |
| Event | Класс-определение мероприятия |
| Bonus | Класс-определение бонуса |

## 6.2. Описание и функциональное назначение полей, методов и свойств

Таблица 2. Описание и функциональное назначение полей класса CodeGenerator

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| \_qrCodeGenerator | QRCodeGenerator | private readonly | Генератор QR-кодов |
| \_randomGenerator | Random | private readonly | Генератор случайных чисел |

Таблица 3. Описание и функциональное назначение методов класса CodeGenerator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификаторы | Тип возвращаемого значения | Аргументы | Назначение |
| GetQrCodeBitmap | public | Bitmap | String token | Возвращает сформированный QR-код |
| GenerateBonusCode | public | int | - | Возвращает сгенерированный код из 4 целых цифр – как промокод. |

Таблица 4. Описание и функциональное назначение полей класса LevelCounter

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| Alpha | double | private const | Константа используемая для вычисления уровня |

Таблица 5. Описание и функциональное назначение методов класса LevelCounter

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификаторы | Тип возвращаемого значения | Аргументы | Назначение |
| GetUserLevel | public | int | Decimal points | Возвращает уровень пользователя в зависимости от переданного количества очков валюты |
| GetDiscountByLevel | public | decimal | Decimal maxDiscount, int userLevel, int maxLevel | Возвращает скидку, которую пользователь может получить в зависимости от его баллов |

Таблица 6. Описание и функциональное назначение полей класса DbSeeder

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| SeedUsersList | IEnumerable<User> | internal | Пользователи, которые будут добавлены в базу данных при инициализации приложения |
| SeedRolesList | IEnumerable<Role> | internal | Роли, которые будут добавлены в базу данных при инициализации приложения |
| SeedOrganizationList | IEnumerable<Organization> | internal | Организации, которые будут добавлены в базу данных при инициализации приложения |

Таблица 7. Описание и функциональное назначение методов класса DbSeeder

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификаторы | Тип возвращаемого значения | Аргументы | Назначение |
| SeedUsers | Public async | Task | - | Заполняет базу данных начальными значениями |
| ApplyRolesToUsers | Public async | Task | - | Заполняет базу данных начальными значениями |
| SeedRoles | Public async | Task | - | Заполняет базу данных начальными значениями |
| SeedOrganizations | Public async | Task | - | Заполняет базу данных начальными значениями |

Таблица 8. Описание и функциональное назначение полей класса DbBaseService

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| DbContext | IdentityDbContext<User> | Protected readonly | Контекст связи с базой данных |

Таблица 9. Описание и функциональное назначение методов класса Atribut

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификаторы | Тип возвращаемого значения | Аргументы | Назначение |
| Add | Public virtual async | Task | TType Item | Поддерживает CRUD-операцию |
| AddRange | Public virtual async | Task | List<TType> | Поддерживает CRUD-операцию |
| FindById | Public virtual async | Task<TType> | Tkey key | Поддерживает CRUD-операцию |
| Delete | Public virtual async | Task | TKey key | Поддерживает CRUD-операцию |
| Update | Public virtual async | Task | TType item | Поддерживает CRUD-операцию |
| UpdateContext | Public virtual async | Task |  | Обновляет контекст базы данных |

Таблица 10. Описание и функциональное назначение полей класса BonusService

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| \_codeGenerator | CodeGenerator | Private readonly | Генератор кодов |

Таблица 11. Описание и функциональное назначение полей класса AppDbContext

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| Organizations | DbSet<Organization> | public | Таблица со значениями организаций |
| Events | DbSet<Event> | public | Таблица со значениями событий |
| Bonuses | DbSet<Bonus> | public | Таблица со значениями бонусов |

Таблица 12. Описание и функциональное назначение полей класса Organization

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| Id | int | [Key] public | Идентификатор |
| Name | string | [Required] Public string | Наименование |
| Phone | String | [Required]  [DataType(DataType.PhoneNumber)]  Public | Телефон |
| Email | String | [Required]  [DataType(DataType.EmailAdress)]  Public | Почта |
| Address | String | Public | Адрес |
| Description | string | public | Описание |

Таблица 13. Описание и функциональное назначение полей класса ParticipantToken

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| Id | int | [Key] public | Идентификатор |
| UserId | String | Public | Ключ пользователя |
| EventId | int | public | Ключ мероприятия |
| IsActivated | bool | public | Активирован ли токен |

Таблица 14. Описание и функциональное назначение полей класса Event

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| Id | int | [Key] public | Идентификатор |
| Name | string | [Required] Public string | Наименование |
| Phone | String | [Required]  [DataType(DataType.PhoneNumber)]  Public | Телефон |
| Email | String | [Required]  [DataType(DataType.EmailAdress)]  Public | Почта |
| Address | String | Public | Адрес |
| Description | string | public | Описание |

Таблица 15. Описание и функциональное назначение полей класса Bonus

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Модификаторы | Назначение |
| Id | int | [Key] public | Идентификатор |
| Name | string | [Required] Public string | Наименование |
| Company | String | Public | Имя компании |
| Discount | decimal | Public | Значение скидки |
| DiscountCode | int | Public | Промокод скидки |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Изм. | Номера листов (страниц) | | | | Всего листов (страниц) в документе | № документа | Входящий № сопроводительного документа и дата | Подпись | Дата |
| измененных | замененных | новых | аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |