СОЗДАНИЕ СИСТЕМЫ ДЛЯ УЧЕТА ИНФОРМАЦИИ ОБ АБИТУРИЕНТАХ С ПОМОЩЬЮ VISUAL STUDIO И ПЛАТФОРМЫ ASP.NET CORE

CREATION OF A SYSTEM FOR ACCOUNTING INFORMATION ABOUT APPLICANTS USING VISUAL STUDIO AND ASP.NET CORE

УДК 004.42

Журавлев Александр Александрович, студент, Уральский федеральный университет им. Б. Н. Ельцина, ИРИТ-РтФ, г. Екатеринбург, ул. Мира, д. 32, e-mail: SanyaProgrammer2503@gmail.com

Zhuravlev A.A. SanyaProgrammer2503@gmail.com

Аннотация

В современное время информация о каком-либо событии изменяется очень быстро. Для того чтобы иметь возможность их зафиксировать можно разработать систему, позволяющую осуществить данное действие. Система позволяет выполнить следующее: программа содержит информацию об институтах, уровнях образования (бакалавриат, специалитет, магистратура, аспирантура), направлениях и абитуриентах; имеется возможность добавления, изменения и удаления информации об абитуриенте; предусмотрена возможность пагинации (переход с одной страницы на другую), а также фильтрации (отбор объектов по конкретному признаку). Приложение разработано с помощью среды разработки Visual Studio Community 2019 и платформы ASP.NET Core. Программа может быть полезна для людей, которые поступают в университет.

Annotation

In modern times, information about an event changes very quickly. In order to be able to fix them, it is possible to develop a system that allows carrying out this action. The system allows doing the following: the program contains information about faculties, education levels (bachelor's, specialty, master's, postgraduate studies), specializations and applicants; it is possible to add, update and delete information about the applicant; there is a possibility of pagination (transition from one page to another), as well as filtering (selection of objects by a specific criterion). The application is developed using the Visual Studio Community 2019 IDE and ASP.NET Core. The program can be useful for people who are going to university.

Ключевые слова: создание, система, учет информации, абитуриенты, Visual Studio, ASP.NET Core

Keywords: creation, system, information control, applicants, Visual Studio, ASP.NET Core

Введение

В современное время информация о каком-либо событии изменяется очень быстро. Для того чтобы иметь возможность их зафиксировать можно разработать систему, позволяющую осуществить данное действие.

Цель статьи – с помощью среды Visual Studio Community 2019 и платформы ASP.NET Core разработать систему для информации об абитуриентах, требования к которому представлены в разделе «Описание задачи».

В качестве **материала исследования** выбрана информация об абитуриентах.

В статье используется **эмпирический метод исследования**, поскольку основной источник результата – проектирование.

Описание задачи

Разработать приложение для учета информации об абитуриентах, удовлетворяющее следующим требованиям:

1) Программа содержит информацию об институтах, уровнях образования (бакалавриат, специалитет, магистратура, аспирантура), направлениях и абитуриентах.
2) Имеется возможность добавления, изменения и удаления информации об абитуриенте.
3) Предусмотрена возможность пагинации (переход с одной страницы на другую), а также фильтрации (отбор объектов по конкретному признаку).
4) Институт имеет следующие характеристики: Название, Количество студентов.
5) Характеристики уровня образования: Название, Количество студентов.
6) Направление подготовки имеет следующие атрибуты: Номер направления, Название, Количество студентов.
7) Атрибуты абитуриента: Фамилия, Имя, Отчество, Рейтинг.

Разработка системы

В качестве среды разработки приложения выбрана Visual Studio Community 2019 [1], платформы – ASP.NET Core [2, 3]. Приложение построено по принципу MVC (Model-View-Controller) [4, 5].

Для хранения информации используется Microsoft SQL Server [6].

База данных состоит из четырех таблиц: Институты, Уровни образования, Направления, Абитуриенты. Код каждой сущности (объекта, который хранится в базе данных) представлен ниже [7, 8].

1) Faculty – институт.

```
namespace DataLayer.Entities.Faculties
{
    public class Faculty
    {
```

```
        public int Id { get; set; } // Идентификационный номер
        public string Name { get; set; } // Название
        public int StudentsCount { get; set; } // Количество студентов
        public List<EducationType> EducationTypes { get; set; } // Список
       уровней образования
    }
}
```

2) EducationType – уровень образования.

```
namespace DataLayer.Entities.EducationTypes
{
    public class EducationType
    {
        public int Id { get; set; } // Идентификационный номер
        public string Name { get; set; } // Название
        public int StudentsCount { get; set; } // Количество студентов
        public Faculty Faculty { get; set; } // Внешний ключ к Faculty
        public int FacultyId { get; set; } // Id института
        public List<Specialization> Specializations { get; set; } // Список
        направлений
    }
}
```

3) Specialization – направление.

```
namespace DataLayer.Entities.Specializations
{
    public class Specialization
    {
        public int Id { get; set; } // Идентификационный номер
        public string Number { get; set; } // Номер направления
        public string Name { get; set; } // Название
        public int StudentsCount { get; set; } // Количество студентов
        public EducationType EducationType { get; set; } // Внешний ключ к
       EducationType
        public int EducationTypeId { get; set; } // Id к уровню образования
        public List<Applicant> Applicants { get; set; } // Список абитуриентов
    }
}
```

4) Applicant – абитуриент.

```
namespace DataLayer.Entities.Applicants
{
    public class Applicant
    {
        public int Id { get; set; } // Идентификационный номер
        public string LastName { get; set; } // Фамилия
        public string FirstName { get; set; } // Имя
        public string SurName { get; set; } // Отчество
        public int Rating { get; set; } // Рейтинг
        public Specialization Specialization { get; set; } // Внешний ключ к
        Specialization
        public int SpecializationId { get; set; } // Id направления
    }
}
```

Работа приложения представлена на рисунках 1-5.

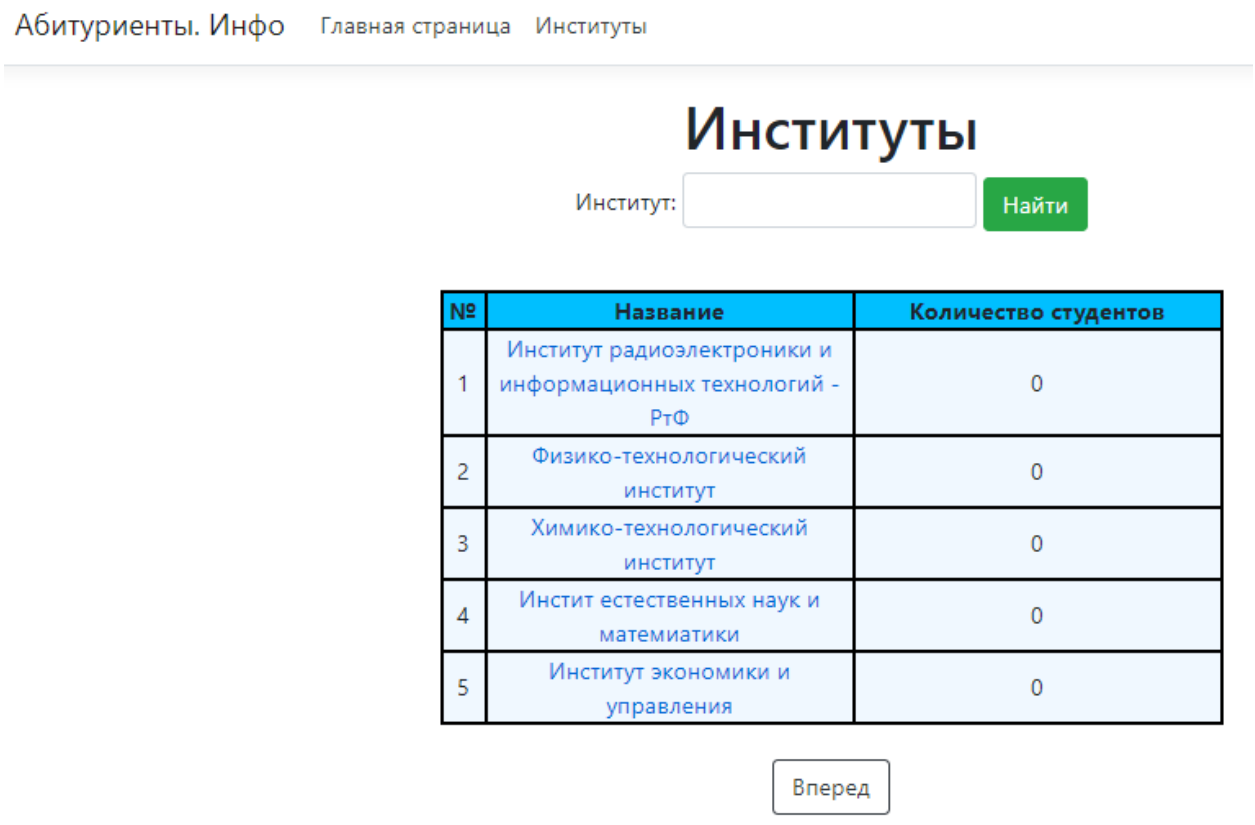

Рисунок 1 – Страница с институтами

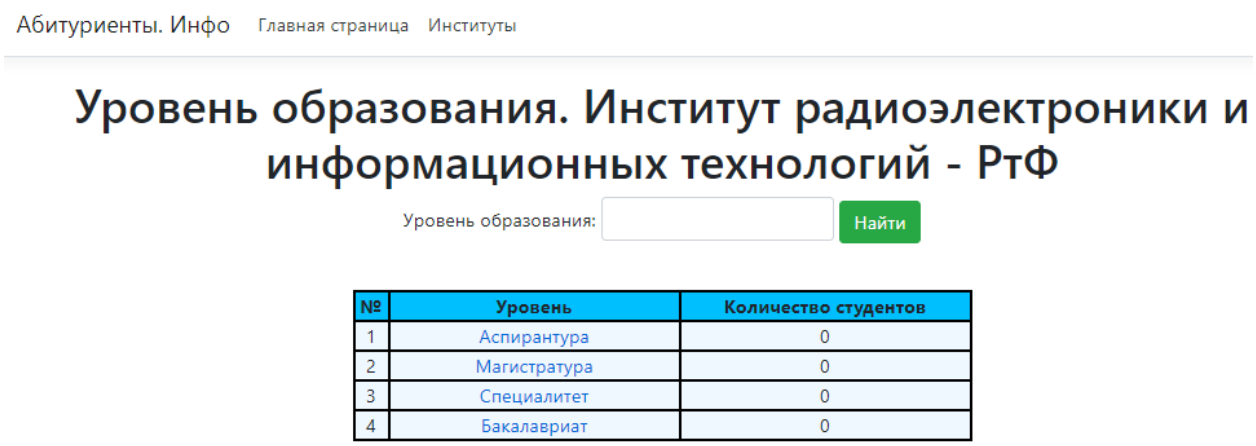

Рисунок 2 – Страница с уровнями образования

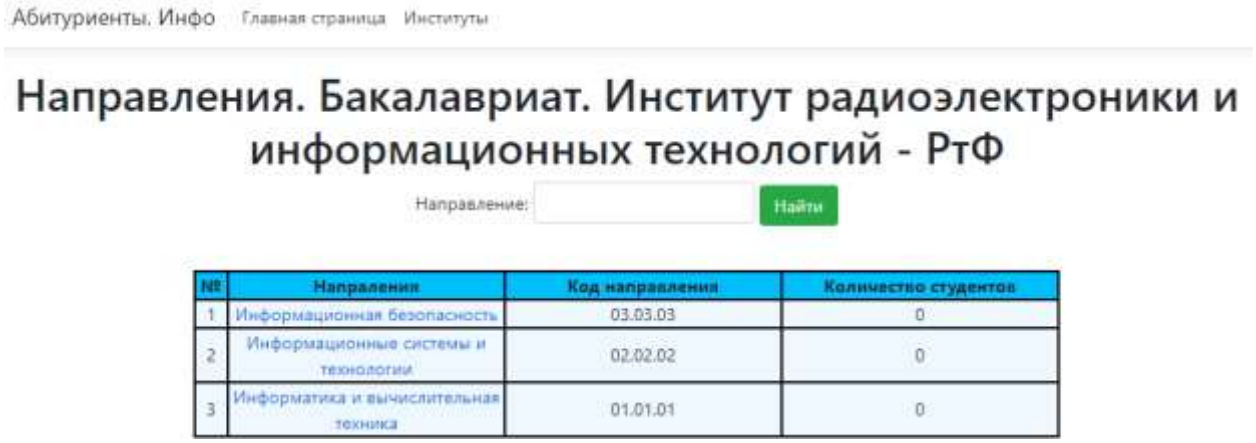

Рисунок 3 – Страница с направлениями подготовки

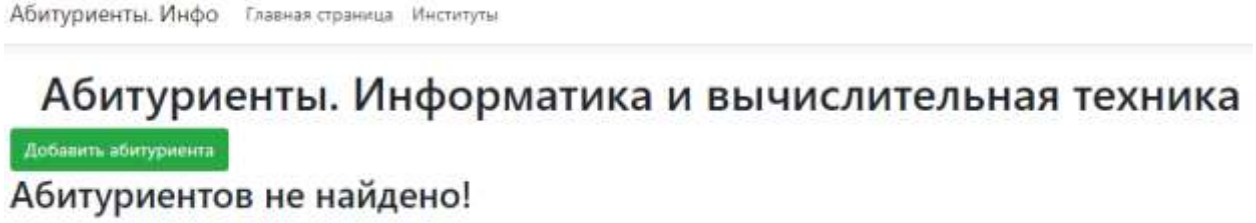

Рисунок 4 – Страница с абитуриентами до добавления

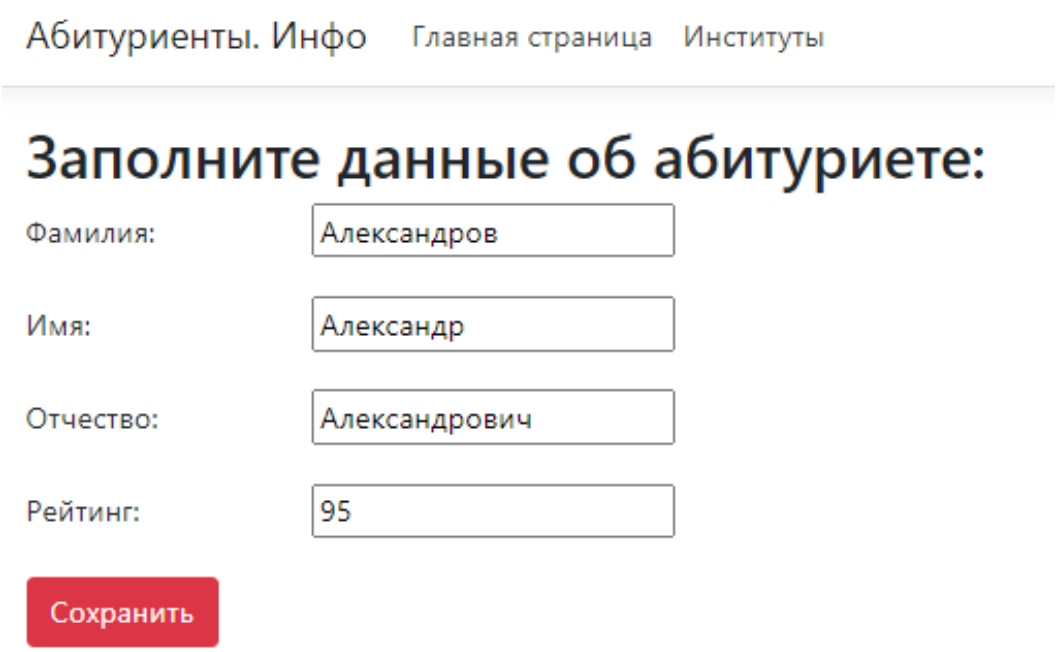

Рисунок 5 – Форма для заполнения информации об абитуриенте

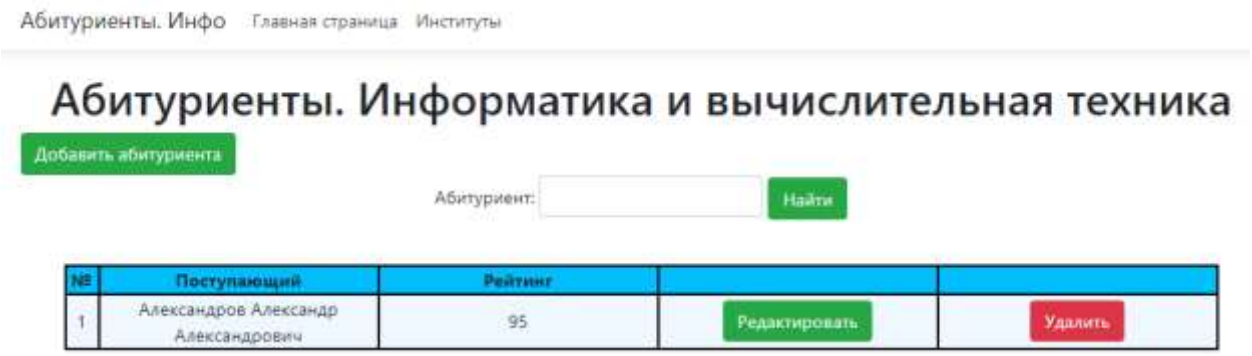

Рисунок 6 – Страница с абитуриентами после добавления

Для перехода к уровням образования необходимо выбрать соответствующий институт, к направлениям подготовки – уровень образования, к абитуриентам – направление подготовки. Во вкладке с абитуриентами пользователь имеет возможность добавлять, редактировать и удалять данные. Система имеет возможность пагинации и фильтрации.

Программа может быть полезна для людей, которые поступают в университет.

Заключение

В данной статье описана разработка приложения для учета информации об абитуриентах с помощью среды Visual Studio Community 2019 и платформы

ASP.NET Core. Программа содержит информацию об институтах, уровнях образования (бакалавриат, специалитет, магистратура, аспирантура), направлениях и абитуриентах. Имеется возможность добавления, изменения и удаления информации об абитуриенте. Также предусмотрена возможность пагинации (переход с одной страницы на другую), а также фильтрации (отбор объектов по конкретному признаку). Приложение может быть полезно для людей, которые поступают в университет.

Литература

1. Документация по Visual Studio [Электронный ресурс] // URL: https://docs.microsoft.com/ru-ru/visualstudio/windows/?view=vs-2019
2. Документация по ASP.NET [Электронный ресурс] // URL: https://docs.microsoft.com/ru-ru/aspnet/core/?view=aspnetcore-5.0
3. Руководство по ASP.NET Core 5 [Электронный ресурс] // URL: https://metanit.com/sharp/aspnet5/
4. Начало работы с MVC ASP.NET Core [Электронный ресурс] // URL: https://docs.microsoft.com/ru-ru/aspnet/core/tutorials/first-mvc-app/start-mvc?view=aspnet core-5.0&tabs=visual-studio
5. Общие сведения ASP.NET Core MVC [Электронный ресурс] // URL: https://docs.microsoft.com/ru-ru/aspnet/core/mvc/overview?view=aspnet core-5.0
6. Документация по Microsoft SQL [Электронный ресурс] // URL: https://docs.microsoft.com/ru-ru/sql/?view=sql-server-ver15
7. Албахари, Джозеф C# 3.0. Справочник / Джозеф Албахари, Бен Албахари. - М.: БХВ-Петербург, 2015. - 499 с.
8. Абрамян, Михаил Visual C# на примерах / Михаил Абрамян. - М.: БХВ-Петербург, 2016. - 436 с.

Literature

1. Visual Studio documentation [Electronic resource] // URL: https://docs.microsoft.com/ru-ru/visualstudio/windows/?view=vs-2019
2. ASP.NET documentation [Electronic resource] // URL: https://docs.microsoft.com/ru-ru/aspnet/core/?view=aspnetcore-5.0
3. ASP.NET Core 5 documentation [Electronic resource] // URL: https://metanit.com/sharp/aspnet5/
4. Getting started with MVC ASP.NET Core [Electronic resource] // URL: https://docs.microsoft.com/ru-ru/aspnet/core/tutorials/first-mvc-app/start-mvc?view=aspnet core-5.0&tabs=visual-studio
5. General information about ASP.NET Core MVC [Electronic resource] // URL: https://docs.microsoft.com/ru-ru/aspnet/core/mvc/overview?view=aspnet core-5.0
6. Microsoft SQL documentation [Electronic resource] // URL: https://docs.microsoft.com/ru-ru/sql/?view=sql-server-ver15
7. Albahari, Joseph C # 3.0. Handbook / Joseph Albahari, Ben Albahari. - M .: BHV-Petersburg, 2015 .-- 499 p.
8. Abrahamyan, Mikhail Visual C # by examples / Mikhail Abrahamyan. - M .: BHV-Petersburg, 2016 .-- 436 p.