# СОЗДАНИЕ СОЦИАЛЬНОЙ СЕТИ С ПОМОЩЬЮ СРЕДЫ РАЗРАБОТКИ VISUAL STUDIO И ПЛАТФОРМЫ ASP.NET CORE MVC

CREATING A SOCIAL NETWORK WITH THE VISUAL STUDIO DEVELOPMENT ENVIRONMENT AND ASP.NET CORE MVC PLATFORM

**УДК 004.42**

**Журавлев Александр Александрович,** студент, Уральский федеральный университет им. Б. Н. Ельцина, ИРИТ-РтФ, г. Екатеринбург
**Немтинов Александр Валерьевич,** кандидат технических наук, доцент, Уральский федеральный университет им. Б. Н. Ельцина, ИРИТ-РтФ, г. Екатеринбург

**Zhuravlev A.A.** SanyaProgrammer2503@gmail.com
**Nemtinov A. V.** a.v.nemtinov@urfu.ru

**Аннотация**

В данный момент социальные сети играют огромную роль в жизни почти каждого человека. Они позволяют людям общаться между собой, а также мгновенно получать важную информацию. Для разработки социальной сети, приведенной в данной статье, используется C#, среда Visual Studio и платформа ASP.NET Core MVC. Социальная сеть соответствует следующим требованиям:

1. Пользователь может разместить произвольную информацию о себе.
2. Остальные пользователи могут оценивать материал, проставляя оценки (лайки) и оставляя комментарии.
3. Предусматриваются списки общения для каждого пользователя.

4. Все сведения хранятся в базе данных.
5. Реализуются процедуры регистрации и авторизации.


## Annotation

Now social networks play a huge role in the life of almost everyone. They allow people to communicate with each other, as well as instantly receive important information. The social networking site described in this article uses C#, Visual Studio and ASP.NET Core MVC platform. The social network fits the following requirements:

1. The user can post any information about himself.
2. Other users can rate the material by putting ratings (likes) and leaving comments.
3. There are communication lists for each user.
4. All information is stored in the database.
5. Registration and authorization procedures are being implemented.



**Ключевые слова:** создание, социальная сеть, Visual Studio, ASP.NET Core MVC.

**Keywords:** creation, social network, Visual Studio, ASP.NET Core MVC.


## Введение

В современное время социальные сети играют огромную роль в жизни почти каждого человека. Они позволяют людям общаться между собой, а также мгновенно получать важную информацию.

Для создания социальной сети используется множество языков программирования такие, как Python, PHP, Java и C#. Для разработки социальной сети, приведенной в данной статье, используется C# и среда Visual Studio.

**Цель статьи** – разработать социальную сеть, описанную в разделе «Постановка задачи», с помощью среды разработки Visual Studio и платформы ASP.NET MVC.

**Материалом исследования** является разрабатываемая социальная сеть.

В статье используется **эмпирический метод исследования**, поскольку основной источник результатов - моделирование.

## Постановка задачи

Разработать систему, в которой каждый пользователь может разместить произвольную информацию о себе. При этом все остальные пользователи должны иметь возможность проставить оценку размещенным материалам (поставить лайк) и/или оставить комментарии к ним. Необходимо предусмотреть списки пользователей общения для каждого пользователя и механизм одобрения включения пользователя в список другого пользователя. Все сведения должны храниться в базе данных. Также приложение должно реализовывать процедуру регистрации и авторизации.

## Решение задачи

Для разработки социальной сети, описанной в разделе «Постановка задачи» используется среда разработки Visual Studio и платформа ASP.NET Core MVC.

Социальная сеть состоит из следующих моделей:

1) User – пользователь социальной сети. Код модели представлен ниже.

```
using System;
using System.Collections.Generic;
using Microsoft.AspNetCore.Identity;

namespace SocialNetworkingSystem.Models
{
    public class User : IdentityUser<int>
    {
        public string FirstName { get; set; } // Имя пользователя
        public string LastName { get; set; } // Фамилия пользователя
        public Gender? Gender { get; set; } // Пол пользователя
        public string ProfilePictureUrl { get; set; } // Ссылка на аватарку
```

```
        public DateTime? BirthDate { get; set; } // Дата рождения
        public string Country { get; set; } // Страна
        public string City { get; set; } // Город
        public DateTime CreatedOn { get; set; } = DateTime.UtcNow; // Дата создания
        профиля
        public ICollection<Page> Pages { get; set; } // Страницы пользователя
        public ICollection<Post> Posts { get; set; } // Посты пользователя
        public ICollection<Comment> Comments { get; set; } // Комментарии к постам
        public ICollection<Like> Likes { get; set; } // Лайки к постам

        public User()
        {
            Pages = new HashSet<Page>();
            Posts = new HashSet<Post>();
            Comments = new HashSet<Comment>();
            Likes = new HashSet<Like>();
        }
    }
}
```

2) Page – пользовательская страница.

```
using System.Collections.Generic;

namespace SocialNetworkingSystem.Models
{
    public class Page
    {
        public int Id { get; set; } // Id страницы
        public ICollection<Post> Posts { get; set; } // список постов

        public Page()
        {
            Posts = new HashSet<Post>();
        }
    }
}
```

3) Post – пост пользователя.

```
using System;
using System.Collections.Generic;

namespace SocialNetworkingSystem.Models
{
    public class Post
    {
        public int Id { get; set; } // Id поста
        public string Title { get; set; } // Название поста
        public string Content { get; set; } // Содержание поста
        public DateTime CreatedOn { get; set; } = DateTime.UtcNow; // Время создания
        поста
        public int UserId { get; set; } // Id пользователя поста
        public User User { get; set; } // Пользователя поста
        public ICollection<Like> Likes { get; set; } // Лайки к посту
        public ICollection<Comment> Comments { get; set; } // Комментарии к посту
        public bool IsDeleted { get; set; } // Переменная для проверки на удаление
        поста

        public Post()
        {
```

```
            Likes = new HashSet<Like>();
            Comments = new HashSet<Comment>();
        }
    }
}
```

4) Comment – комментарий к посту.

```
using System;
using System.ComponentModel.DataAnnotations.Schema;

namespace SocialNetworkingSystem.Models
{
    public class Comment
    {
        public int Id { get; set; } // Id комментария
        public int AuthorId { get; set; } // Id автора комментария
        public User Author { get; set; } // Автор комментария
        public string CommentText { get; set; } // Текст комментария
        [ForeignKey("Post")]
        public int PostId { get; set; } // Id поста
        public Post Post { get; set; } // Пост
        public DateTime PostedOn { get; set; } = DateTime.UtcNow; // Дата комментария
    }
}
```

5) Like – оценка (лайк) к посту.

```
using System;

namespace SocialNetworkingSystem.Models
{
    public class Like
    {
        public int UserId { get; set; } // Id Лайка
        public User User { get; set; } // Пользователь, поставивший лайк
        public int PostId { get; set; } // Id поста
        public Post Post { get; set; } // Пост
        public DateTime LikedOn { get; set; } = DateTime.UtcNow; // Дата проставления
        лайка
    }
}
```

6) UserFollower – подписчик текущего пользователя.

```
using System;

namespace SocialNetworkingSystem.Models
{
    public class UserFollower
    {
        public int UserId { get; set; } // Id пользователя
        public User User { get; set; } // Пользователя
        public int FollowerId { get; set; } //Id подписчика
        public User Follower { get; set; } // Подписчик
        public DateTime FollowedOn { get; set; } = DateTime.UtcNow; // Дата подписки
    }
}
```

Для каждой модели были реализованы соответствующий контроллер и представление.

Для миграции БД используется команда add-migration в консоли диспетчера пакетов (рисунок 1).

```
PM> add-migration Sociality
```

Рисунок 1 – Создание миграции базы данных

Процедуры регистрации и авторизации пользователя в приложении представлены на рисунках 2 и 3.

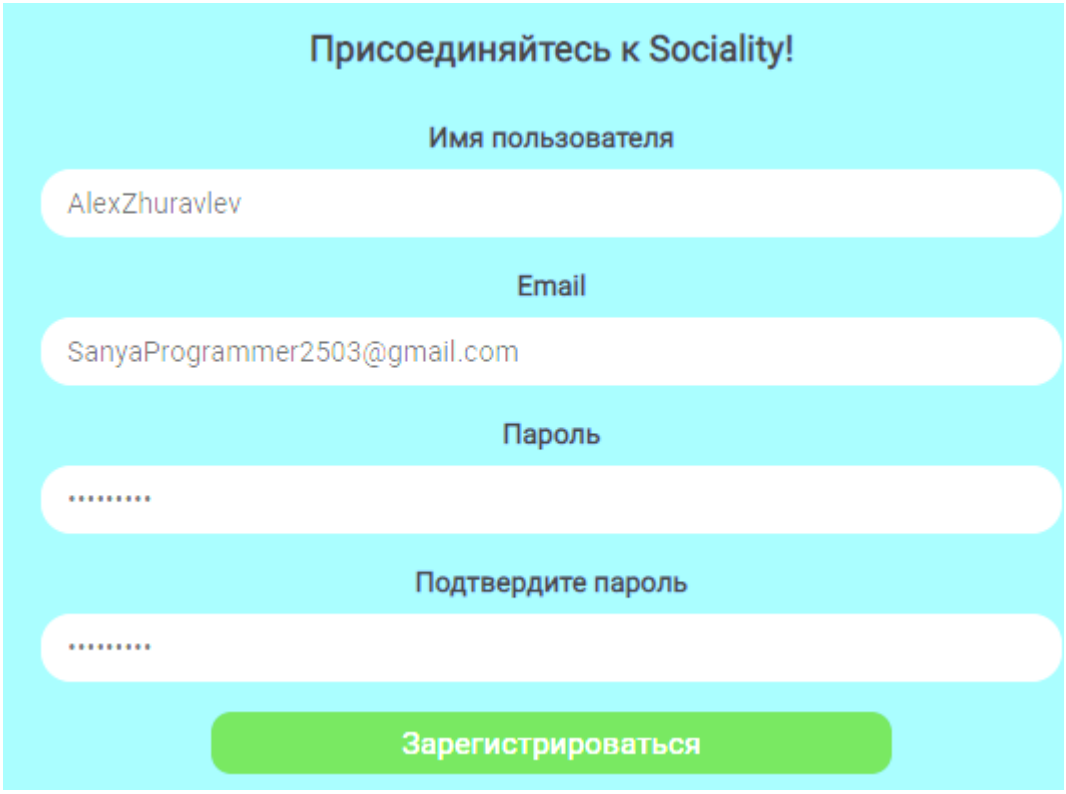



Рисунок 2 – Регистрация пользователя

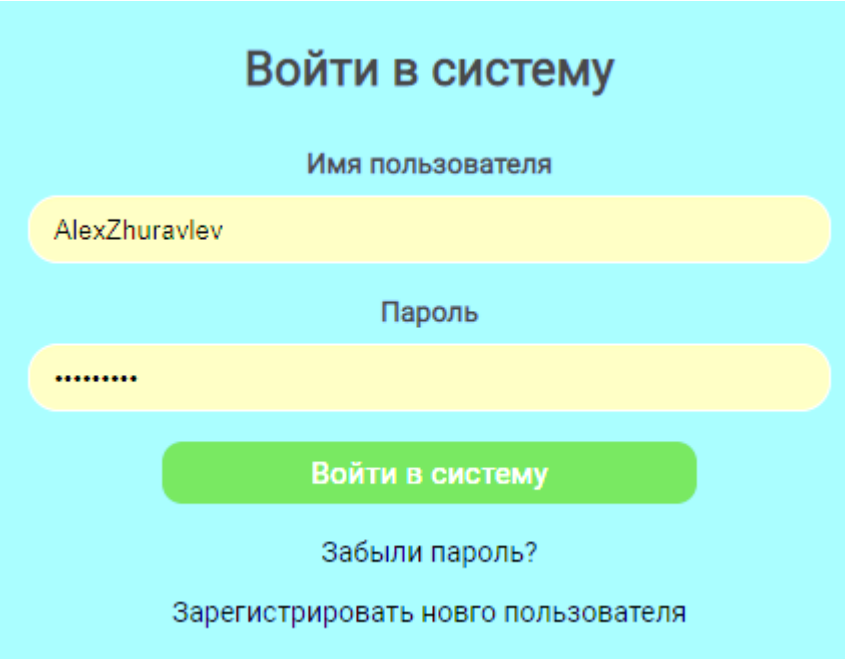



Рисунок 3 – Вход пользователя в систему

Страница с вводом пользователем информации о себе представлена на рисунке 4.

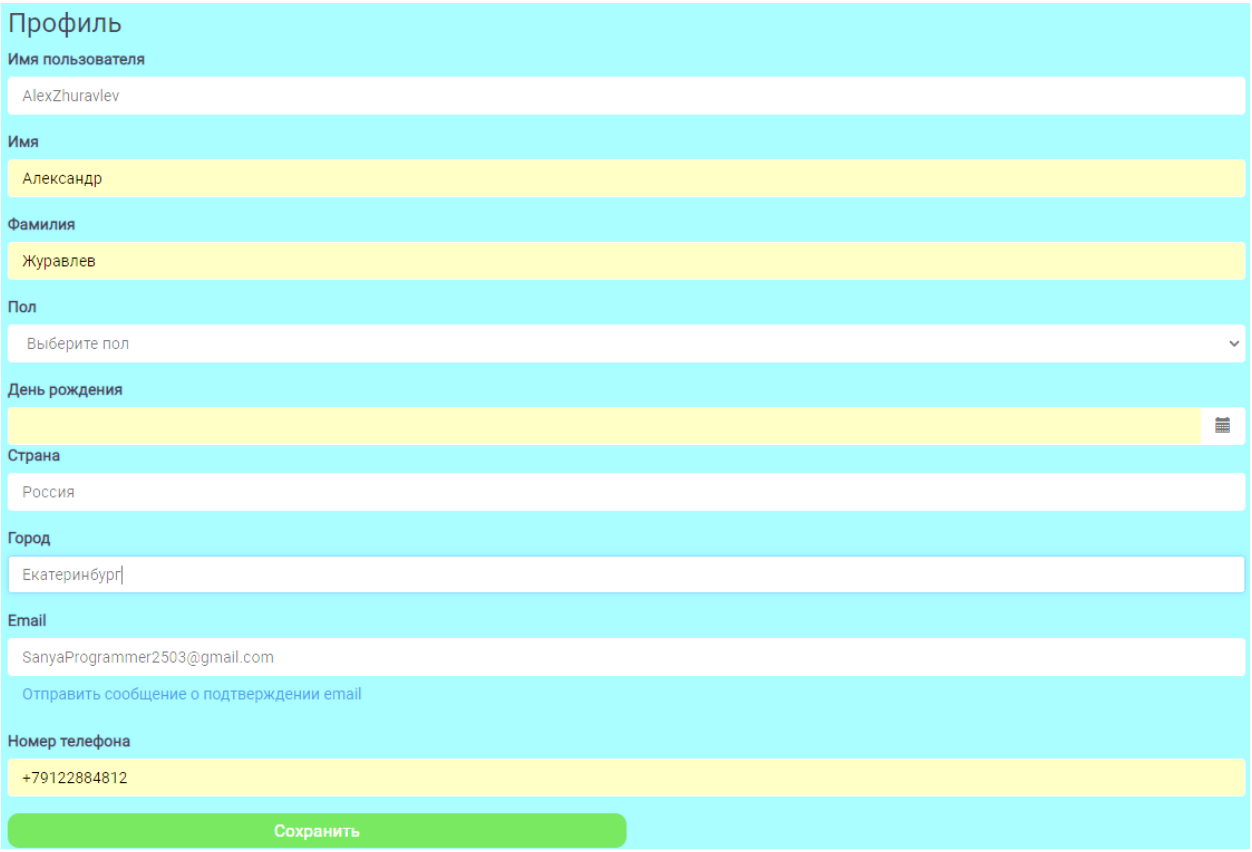



Рисунок 4 – Заполнение информации пользователем

Интерфейс страницы пользователя представлен на рисунке 5.

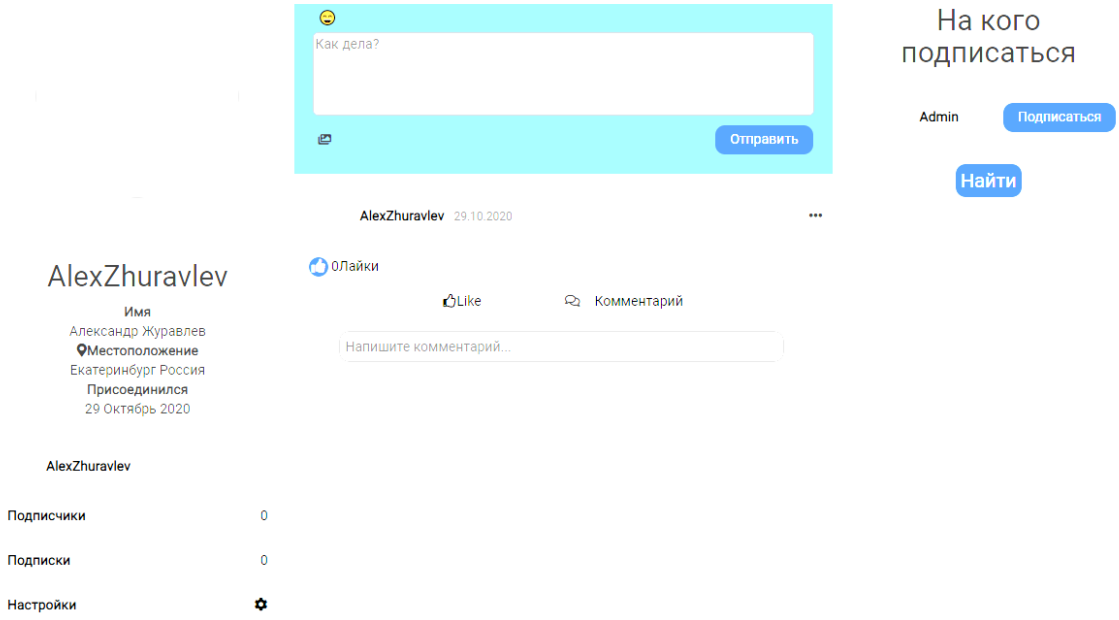



Рисунок 5 – Интерфейс страницы пользователя

На рисунке 5 представлен интерфейс страницы пользователя. На нем представлено поле, в котором можно написать текст поста. Чтобы пост появился в социальной сети, необходимо нажать кнопку «Отправить». Данный пост могут оценивать (кнопка «Like») и оставлять комментарий (кнопка «Комментарий»). Кроме этого, на левой панели представлена информация о пользователе: его имя, местоположение, количество подписчиков, подписки, и пр.

## Заключение

В данной статье с помощью среды разработки Visual Studio описана модель социальной сети, которая соответствует следующим требованиям:

1. Пользователь может разместить произвольную информацию о себе.
2. Остальные пользователи могут оценивать материал, проставляя оценки (лайки) и оставляя комментарии.
3. Предусматриваются списки общения для каждого пользователя.
4. Все сведения хранятся в базе данных.

5. Реализуются процедуры регистрации и авторизации.

## Литература


1. Руководство по ASP.NET MVC 5 [Электронный ресурс]. URL: https://metanit.com/ sharp/mvc5/
2. ASP.NET MVC 3 [Электронный ресурс]. URL: https://docs. microsoft.com/ru-ru/aspnet/mvc/mvc3
3. Бхаргава А.. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих. - СПб.: Питер, 2017. - 288 С.
4. Дасгупта С. Алгоритмы / С. Дасгупта, Х. Пападимитриу, У. Вазирани; Пер. с англ. под ред. А. Шеня. — М.: МЦНМО, 2014. — 320 С.
5. Седжвик, Роберт, Уэйн, Кевин. Алгоритмы на Java, 4-е изд.: Пер. с англ. – М. : ООО «И.Д. Вильямс», 2013. – 848 С.
6. Лафоре Р.. Структуры данных и алгоритмы в Java. Классика Computers Science. 2-е изд. — СПб.: Питер, 2013. — 704 С.


## Literature


1. Guide to ASP.NET MVC 5 [Electronic resource]. URL: https:// metanit.com/sharp/mvc5/
2. ASP. NET MVC 3 [Electronic resource]. URL: https://docs.microsoft. com/ru-ru/aspnet/mvc/mvc3
3. Bhargava A.. Grokah algorithms. Illustrated guide for programmers and curious. - St. Petersburg: Peter, 2017 . - 288 P.
4. Dasgupta S. Algorithms / S. Dasgupta, H. Papadimitriou, U. Wazirani; Translate from English under the editorship of A. Shen. - M: 2014. - 320 P.
5. Sedgwick, Robert, Wayne, Kevin. Algorithms in Java, 4th ed.: Translate from English - M.: LLC «I.D. Williams», 2013. - 848 P.
6. Lafore R.. Data structures and algorithms in Java. Classic Computers Science. 2nd ed. – St. Petersburg: Peter, 2013. – 704 P.