

УДК 681.518:004.652

*В.П. Часовских, Е.В. Кох*

Уральский государственный лесотехнический университет, Институт экономики и управления, г. Екатеринбург

**САЙТ ПРЕПОДАВАТЕЛЯ ВУЗА - БАЗА ДАННЫХ И ПЕРВАЯ СТРАНИЦА**

Определив модель и логику предметной области сайта преподавателя вуза (Часовских, Стаин, 2013, 2014 а,б) отобразим их в базу данных с помощью инфраструктуры Entity Framework. Подобное действие будет выполняться, автоматически не требуя понимания и технологии работы Entity Framework. Для базы данных будет применяться SQL сервер. Доступ к базе данных будет осуществляться с помощью Entity Framework (EF) – инфраструктуры ORM (Object Relational Mapping ) для платформы .NET (Столбовский, 2011; Эспозито, 2012; Фримен, Сандерсон, 2012; Фримен, 2015а,б). ORM позволяет работать с таблицами, полями и связями реляционной БД, как с объектами, свойствами и массивами, не рассматривая вопросы порядка выборки и сохранения модифицированных данных, вопросы переносимости, генерации уникальных первичных ключей, заполнения полей ссылок для моделирования связей. ORM представляет таблицы, строки и столбцы базы данных с помощью обычных объектов C# и позволяет использовать эффективное средство LINQ (Фримен и др., 2011), имеющее SQL – подобный синтаксис запросов к базе данных.

**Создание базы данных**

При разработке сайта мы используем Visual Studio, включающее удобное средство LocalDB (локальную базу данных SQL). LocalDB не требует администрирования и предназначена для разработки проектов Visual Studio.

Для создания базы данных необходимо подключиться к LocalDB. Для этого необходимо в Visual Studio открыть меню «Вид» и «Обозреватель серверов» (рис. 1).

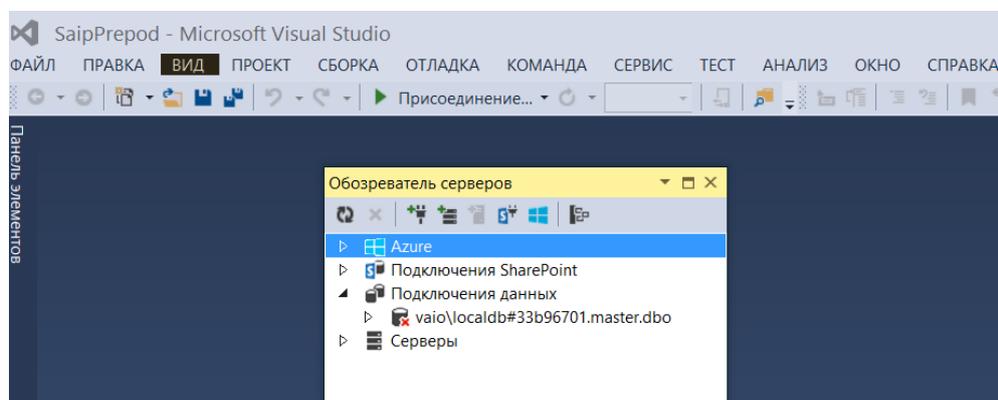
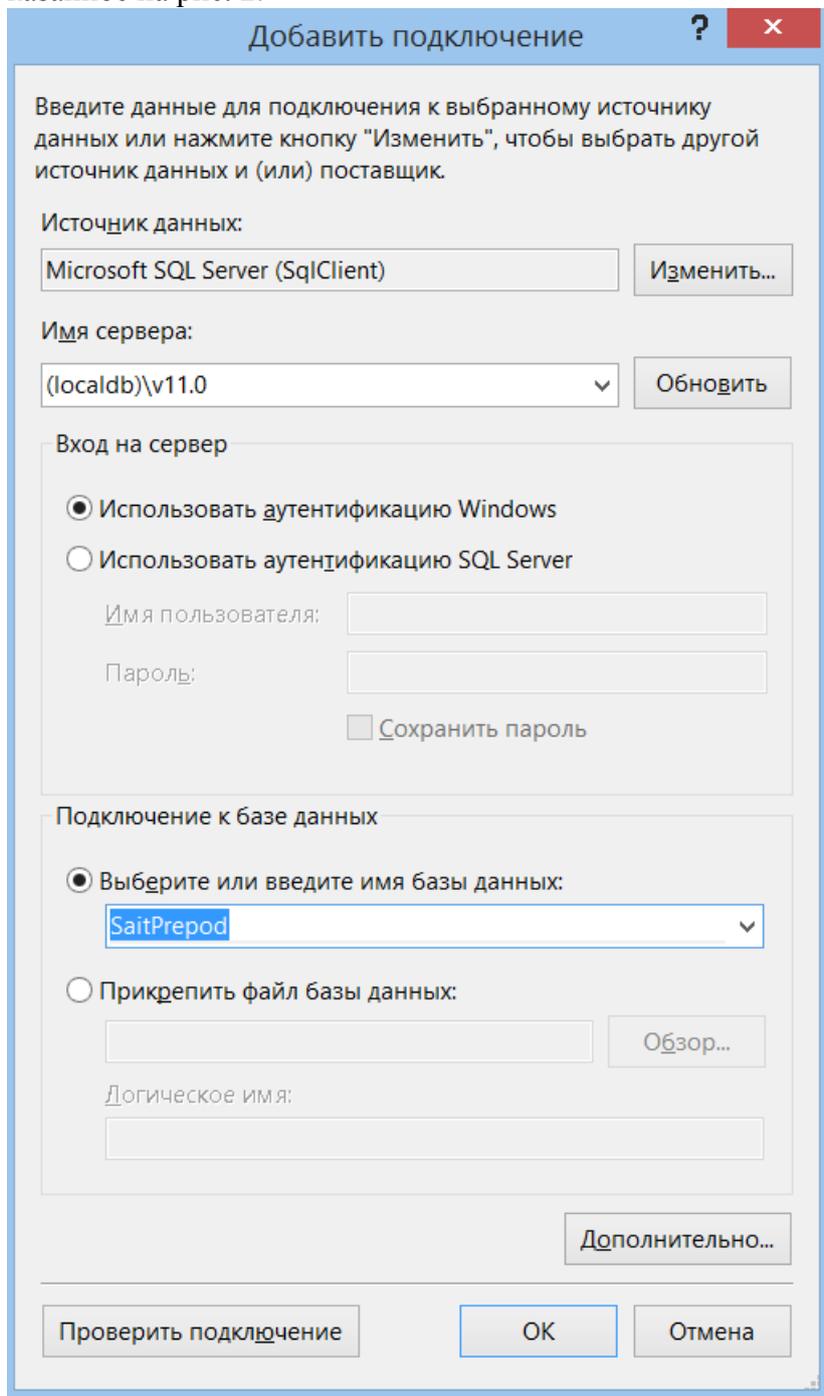


Рис.1. Обозреватель серверов.

Далее щелкните по  и откроется диалоговое окно «Добавить подключение», показанное на рис. 2.



Добавить подключение

Введите данные для подключения к выбранному источнику данных или нажмите кнопку "Изменить", чтобы выбрать другой источник данных и (или) поставщик.

Источник данных:  
Microsoft SQL Server (SqlClient)

Имя сервера:  
(localdb)\v11.0

Вход на сервер

Использовать аутентификацию Windows  
 Использовать аутентификацию SQL Server

Имя пользователя:   
Пароль:   
 Сохранить пароль

Подключение к базе данных

Выберите или введите имя базы данных:  
SaitPrepod

Прикрепить файл базы данных:  
   
Логическое имя:

Рис.2. Добавить подключение

Необходимо указать (localdb)\v11.0, имя базы данных SaitPrepod и нажать ОК. В окне «Обозреватель сервера» появится созданная база данных (рис. 3).

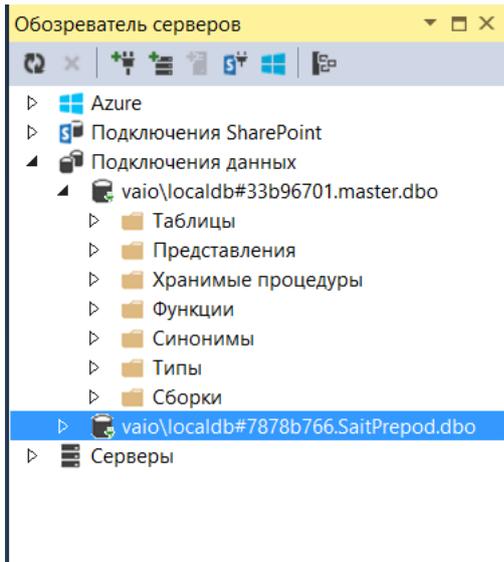


Рис. 3. Созданная база данных

Созданную базу данных нужно закрепить за создаваемым сайтом. Необходимо в файле Web.config добавить в секцию <connectionStrings> доступ к базе данных.

```
<connectionStrings>
```

```

<add name="EFDbContext" connectionString="Data Source=(localdb)\v11.0;Initial Catalog=SaitPrepodChas201501;Integrated Security=True" providerName="System.Data.SqlClient" />

```

```
</connectionStrings>
```

Далее необходимо добавить в наши проекты удобные пакеты и сборки. Необходимо правой кнопкой мышки щелкнуть по имени проекта SaitPrepod.Domain, получим форму (рис. 4). Выбираем «Добавить» и «Ссылка...» (рис.5).

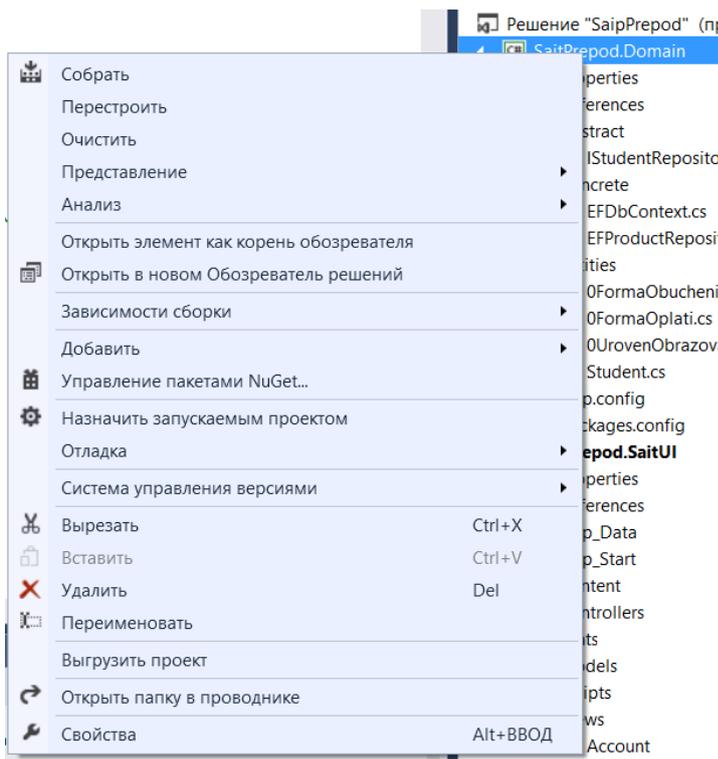


Рис. 4. Команда «добавить».

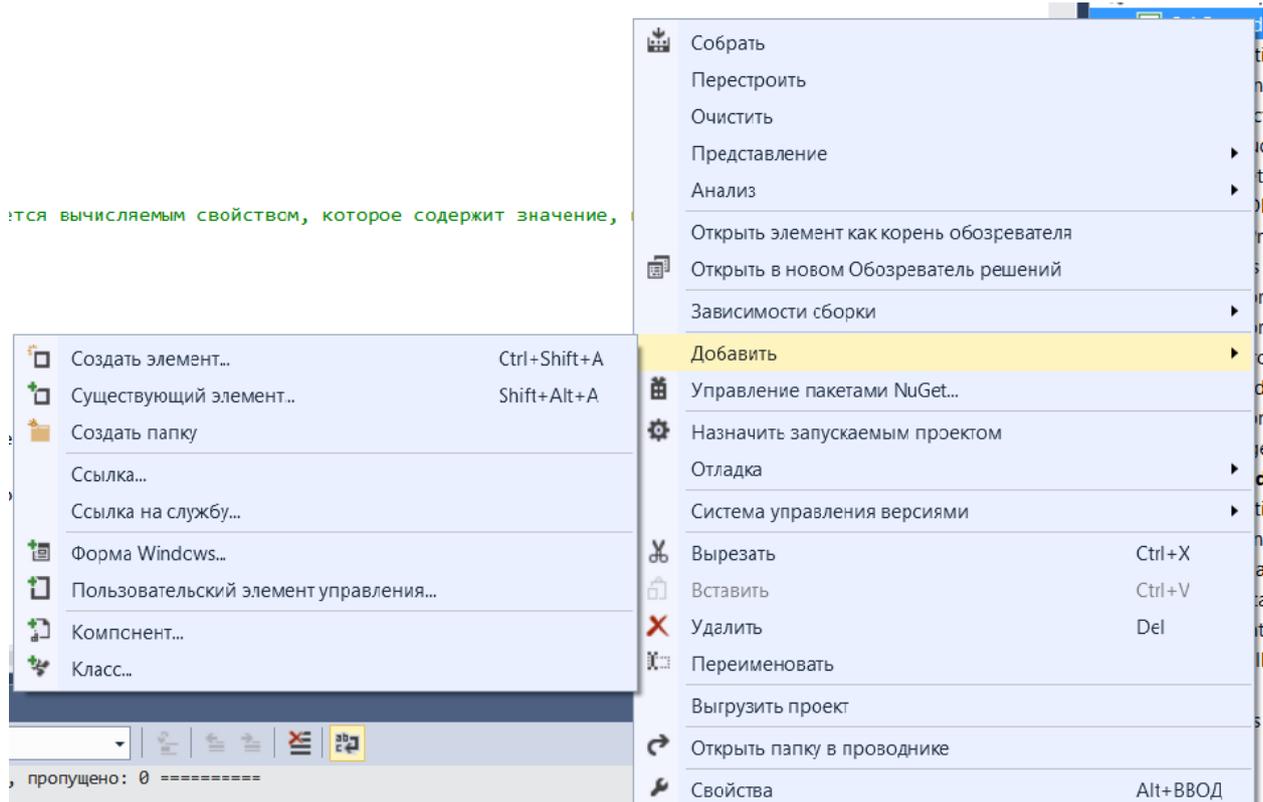


Рис. 5. Команда «Ссылка».

В появившейся форме указываем сборку System.ComponentModel.DataAnnotations и нажимаем кнопку ОК (рис. 6).

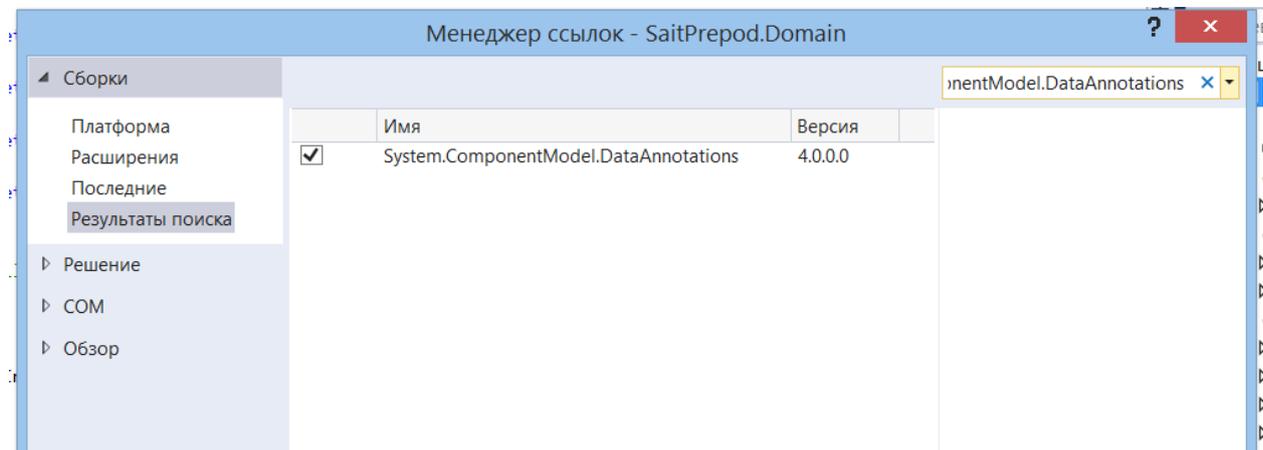


Рис. 6. Установка сборки.

Установка полезных пакетов осуществляется выполнением следующих команд. В меню «Сервис» выбираем «Консоль диспетчеров пакетов» (рис. 7). Далее, в поле команд диспетчера пакетов (рис. 8) вводим и выполняем следующие команды:

```

Install-Package Ninject -version 3.0.1.10 -projectname SaitPrepod.SaitUI
Install-Package Ninject.Web.Common -version 3.0.0.7 -projectname SaitPrepod.SaitUI
Install-Package Ninject.MVC3 -Version 3.0.0.6 -projectname SaitPrepod.SaitUI
Install-Package Moq -version 4.1.1309.1617 -projectname SaitPrepod.SaitUI
Install-Package Microsoft.AspNet.Mvc -version 5.0.0 -projectname SaitPrepod.Domain
Install-Package EntityFramework -projectname SaitPrepod.Domain
Install-Package EntityFramework -projectname SaitPrepod.SaitUI
    
```

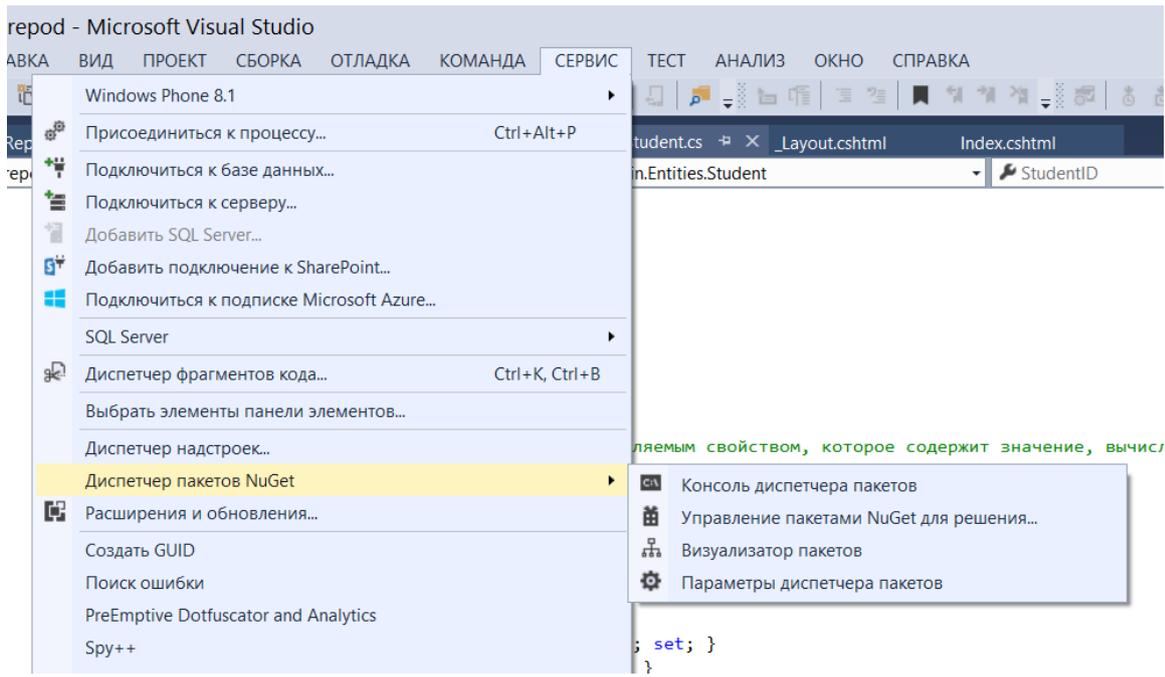


Рис.7. Выбор «Диспетчера пакетов NuGet».

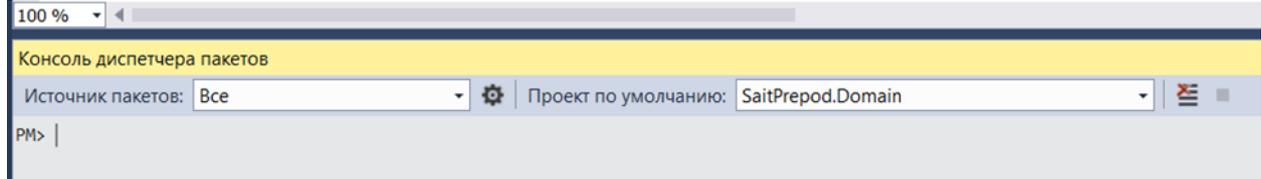


Рис. 8. Диспетчер пакетов.

**Подготовка описаний таблиц базы данных и доступа к ним**

Далее, в качестве примера рассмотрим описание сущности «Студент» и трех справочников.

В проекте SaitPrepod.Domain создадим папку Entities. Щелчком правой кнопкой мыши по названию папки и выберем «создать класс Student» (рис. 9).

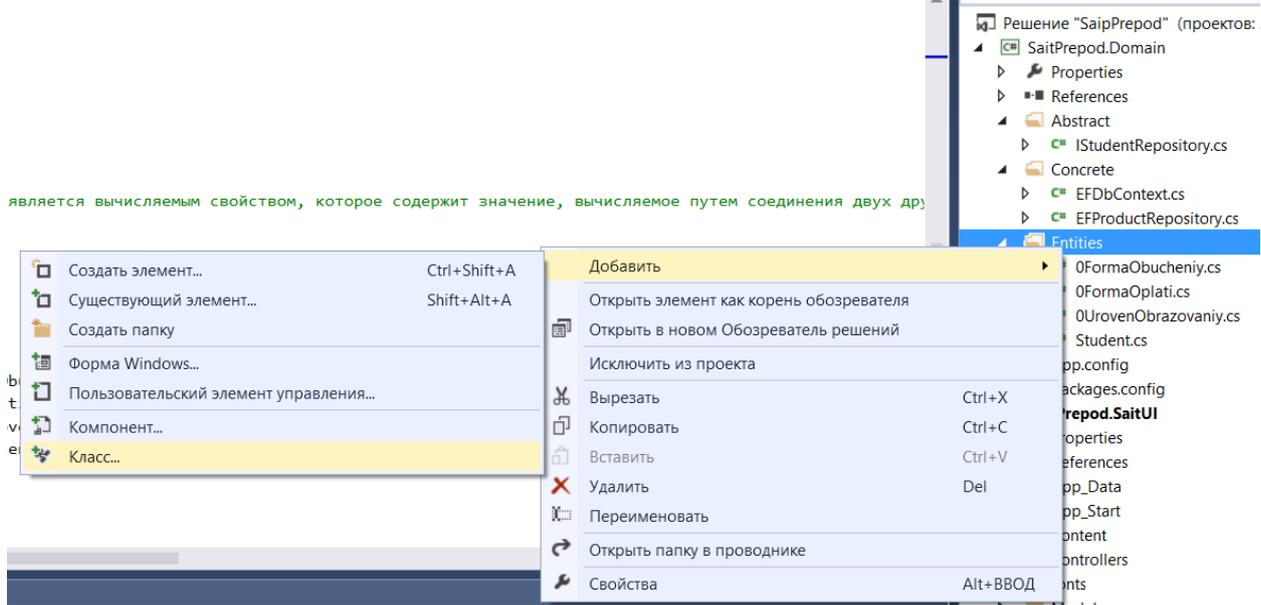


Рис. 9. Создание класса.

Заменяем код класса Student следующим листингом (рис. 10).

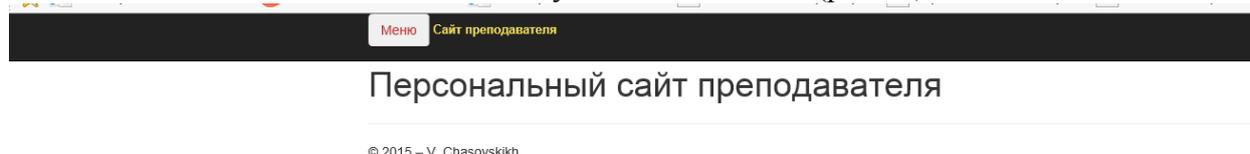


Рис. 10. Начальная страница сайта.

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

namespace SaitPrepod.Domain.Entities
{
    public class Student
    {
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        [Display(Name = "Номер студенческого билета")]
        public int StudentID { get; set; }
        [MaxLength(50, ErrorMessage = "В фамилии можно указать до 50 символов.")]
        [Display(Name = "Фамилия")]
        public string Familiy { get; set; }
        [Display(Name = "Имя")]
        public string Imy { get; set; }
        [Display(Name = "Отчество")]
        public string Otchestvo { get; set; }
        [Display(Name = "Форма обучения")]
        public int _0FormaObucheniyaID { get; set; }
        [Display(Name = "Уровень образования")]
        public int _0UrovenObrazovaniyaID { get; set; }
        [Display(Name = "Форма оплаты")]
        public int _0FormaOplatiID { get; set; }
        [Display(Name = "Группа")]
        public string Gruppy { get; set; }
        [Display(Name = "Дата зачисления")]
        [DisplayFormat(DataFormatString = "{0:d}", ApplyFormatInEditMode = true)]
        public DateTime DataZachisleniy { get; set; }

        public virtual _0FormaObucheniya _0FormaObucheniya { get; set; }
        public virtual _0FormaOplati _0FormaOplati { get; set; }
        public virtual _0UrovenObrazovaniya _0UrovenObrazovaniya { get; set; }
        public virtual ICollection<Student> Students { get; set; }
    }
}
```

Аналогично, в папке Entities проекта SaitPrepod.Domain создает три класса с именами соответственно 0FormaObucheniya, 0FormaOplati, 0UrovenObrazovaniya и сгенерированный код заменяем соответственно следующими листингами:

#### Листинг 0FormaObucheniya

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.ComponentModel.DataAnnotations;
```

```

using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace SaitPrepod.Domain.Entities
{
    public class _0FormaObucheniya
    {
        public int _0FormaObucheniyaID { get; set; }
        [DisplayName("Формы обучения")]
        public string FormaObucheniya { get; set; }
        public virtual ICollection<_0FormaObucheniya> _0FormaObucheniya { get; set; }
    }
}

```

#### Листинг 0FormaOplati

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace SaitPrepod.Domain.Entities
{
    public class _0FormaOplati
    {
        [DisplayName("Ключ")]
        public int _0FormaOplatiID { get; set; }
        [DisplayName("Формы оплаты")]
        public string FormaOplati { get; set; }
        public virtual ICollection<_0FormaOplati> _0FormaOplatis { get; set; }
    }
}

```

#### Листинг 0UrovenObrazovaniya

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace SaitPrepod.Domain.Entities
{
    public class _0UrovenObrazovaniya
    {
        [DisplayName("Ключ")]
        public int _0UrovenObrazovaniyaID { get; set; }
        [DisplayName("Уровень образования")]
        public string UrovenObrazovaniya { get; set; }
        public virtual ICollection<_0UrovenObrazovaniya> _0UrovenObrazovaniya { get; set; }
    }
}

```

Создаем в проекте SaitPrepod.Domain папку Concrete, в ней два класса с именами EFDbContext и EFProductRepository. Сгенерированные коды заменяем следующим:

#### Листинг EFDbContext

```

using SaitPrepod.Domain.Entities;

```

```
using System.Data.Entity;
namespace SaitPrepod.Domain.Concrete {
    public class EFDbContext : DbContext
    {
        public DbSet<Student> Students { get; set; }
    }
}
```

### Листинг EFProductRepository

```
using SaitPrepod.Domain.Abstract;
using SaitPrepod.Domain.Entities;
using System.Collections.Generic;
namespace SaitPrepod.Domain.Concrete
{
    public class EFStudentRepository : IStudentRepository
    {
        private EFDbContext context = new EFDbContext();
        public IEnumerable<Student> Students
        {
            get { return context.Students; }
        }
    }
}
```

Создаем в проекте SaitPrepod.Domain папку с именем Abstract, а в ней класс IStudentRepository. Заменяем сгенерированный код следующим:

### Листинг IStudentRepository

```
using System.Collections.Generic;
using SaitPrepod.Domain.Entities;
namespace SaitPrepod.Domain.Abstract
{
    public interface IStudentRepository
    {
        IEnumerable<Student> Students { get; }
    }
}
```

Назначение всех созданных библиотек и классов в них рассмотрим позже, после запуска сайта.

### Подготовка начальной страницы сайта

Ранее мы назначили запускаемым проектом SaitPrepod.SaitUI. В соответствии с принятыми соглашениями при запуске проекта выполняется то, что указано в классе RouteConfig библиотеки App\_Start.

### Листинг RouteConfig

```
namespace SaitPrepod.SaitUI
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");
            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }
            );
        }
    }
}
```

```
}
}
```

Запускается Index из библиотеки Home.

Заменяем код Index сгенерированный при создании проекта следующим:

#### Листинг Index в папке Home

```
@{
    ViewBag.Title = "Сайт преподавателя";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
```

```
<br />
<h1>Персональный сайт преподавателя</h1>
```

Изменим содержимое модуля \_Layout.cshtml папки Shared из библиотеки(папки) Views.

#### Листинг \_Layout.cshtml

```
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>@ViewBag.Title - Официальный сайт преподавателя</title>
    <meta name="description" content="Сайт полностью отвечающий требованиям нового закона об образовании" />
    <meta name="keywords" content="Менеджмент, УГЛУ, наука, институт экономики и управления, студенту, преподаватели, рабочие программы, патенты, УГЛУ" />
    <meta name="author" content="Часовских В.П. Chasovskikh V." />
    <link href="~/Content/bootstrap.css" rel="stylesheet" />
    <link href="~/Content/Site.css" rel="stylesheet" />
</head>
<body class="panel-body panel-danger">
    <div class="navbar navbar-inverse navbar-fixed-top">
        <div class="navbar-inner">
            <div class="container">
                <button type="button" class="btn btn-navbar" data-toggle="collapse" data-target=".nav-collapse">
                    <span class="icon-bar"></span>
                    <span class="icon-bar" style="color:#bd2f2f">Меню</span>
                    <span class="icon-bar"></span>
                </button>
                @Html.ActionLink("Сайт профессора В.Часовских", "Index", "Slaider1Stranizi", new { area = "" }, new { @class = "brand", @style = "color:#f1dc5d;font-weight:bolder;font-size:0.9em;" })
                <div class="nav-collapse collapse">
                    <ul class="nav">
                        <li>@Html.ActionLink("Сайт", "Index", "Slaider1Stranizi", new { @style = "color:#f1dc5d;font-weight:bolder;" }, @class = "label-primary" )</li>
                        <li>@Html.ActionLink("Консультация (вопрос-ответ)", "Index", "VoprosOtvets", new { area = "" }, new { @style = "color:#f1dc5d;font-weight:bolder;" }, @class = "label-primary" )</li>
                        <li>@Html.ActionLink("Научные исследования", "Index", "NauchIssledovs", new { area = "" }, new { @style = "color:#f1dc5d;font-weight:bolder;" }, @class = "label-primary" )</li>
```

```

        <li>@Html.ActionLink("Студенты", "Index", "Stu-
dents", new { area = "" }, new { @style = "color:#f1dc5d;font-
weight:bold"; }, @class = "label-primary" )</li>
        <li>@Html.ActionLink("Библиотека", "Index", "Bib-
lioteka", new { area = "" }, new { @style = "color:#f1dc5d;font-
weight:bold"; }, @class = "label-primary" )</li>
        <li>@Html.ActionLink("Обучение", "Index",
"Obuchenie", new { area = "" }, new { @style = "color:#f1dc5d;font-
weight:bold"; }, @class = "label-primary" )</li>
        <li>@Html.ActionLink("Расписание", "Index",
"Raspisanies", new { area = "" }, new { @style = "color:#f1dc5d;font-
weight:bold"; }, @class = "label-primary" )</li>
        <li>@Html.ActionLink("Об авторе", "Index",
"VikChas", new { area = "" }, new { @style = "color:#f1dc5d;font-
weight:bold"; }, @class = "label-primary" )</li>
        <li>@Html.ActionLink("Контакты", "Contact",
"Home", new { area = "" }, new { @style = "color:#f1dc5d;font-
weight:bold"; }, @class = "label-primary" )</li>
    </ul>
    @Html.Partial("_LoginPartial")

    </div>
</div>
</div>
</div>
<div class="container body-content">
    @RenderBody()
    <hr />
    <footer>
        <div id="footerimy">&copy; @DateTime.Now.Year - V. Chasov-
skikh</div>
    </footer>
</div>
<script src="~/Scripts/jquery-1.10.2.js"></script>
<script src="~/Scripts/bootstrap.js"></script>
<script src="~/Scripts/respond.js"></script>
    @RenderSection("scripts", required: false)
</body>
</html>

```

### Запуск сайта преподавателя

Убеждаемся, что запускаемым проектом назначен **SaitPrepod.SaitUI** (выделен утолщенной линией).

Для запуска нажимаем вместе клавиши Ctrl и F5. На экране монитора увидим начальную страницу нашего сайта (см. рис. 10).

Если щёлкнем по кнопке **Меню**, то меню разворачивается и доступны ссылки на другие страницы сайта (рис. 11).

При первом запуске сайта в базе данных созданы все таблицы и их связи из папки Entities. Проверяем, вызываем «Обозреватель серверов» и раскрываем таблицы (рис. 12).

В заключение средствами Visual Studio строим таблицы базы данных и связи между ними (рис. 13).



Рис.11. Меню сайта.

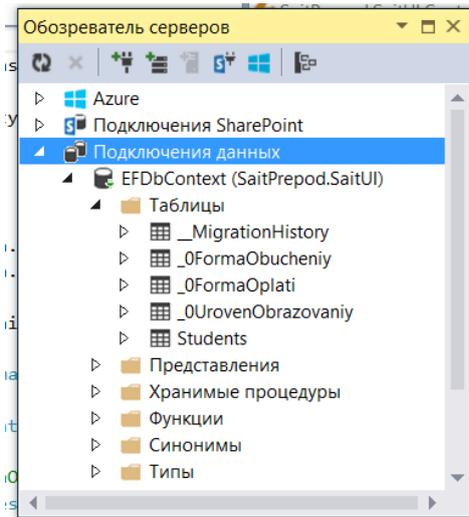


Рис.12. Таблицы базы данных.

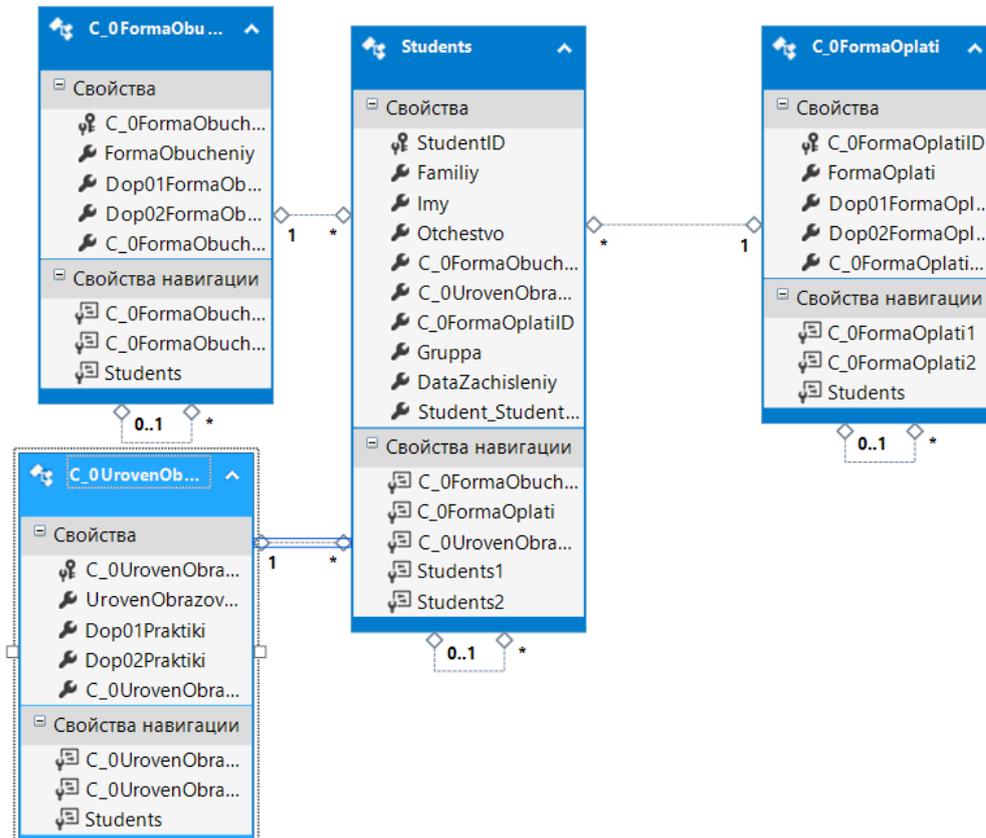


Рис.13. База данных сайта.

## Список использованной литературы

*Столбовский Д.Н.* Основы разработки Web-приложений на ASP.NET: Учебное пособие. М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2011. 304 с. (Основы информационных технологий). ([librabook.com.ua/121/Osnovu...na-ASP-NET-8951.html](http://librabook.com.ua/121/Osnovu...na-ASP-NET-8951.html)).

*Фримен А.* MVC 5 с примерами на C# для профессионалов. 5е изд.: Пер. с англ. М.: ООО «И.Д. Вильямс», 2015а. 736 с.

*Фримен А.* jQuery для профессионалов. 5-е изд.: Пер. с англ. М.: ООО «И.Д. Вильямс», 2015б. 1040 с. ([williamspublishing.com/cgi-bin/list.cgi?id=1070862](http://williamspublishing.com/cgi-bin/list.cgi?id=1070862)).

*Фримен А., Ратти Д.С., Джозеф С.* LINQ: язык интегрированных запросов в C# 2010 для профессионалов / Пер. с англ. М.: ООО «И.Д. Вильямс», 2011. 656 с. ([darkos.club/viewtopic.php?t=174144](http://darkos.club/viewtopic.php?t=174144)).

*Фримен А., Сандерсон С.* ASP.NET MVC 3 Framework с примерами на C# для профессионалов. 3-е изд.: Пер. с англ. М.: ООО «И.Д. Вильямс», 2012. 672 с. ([dom-eknig.ru/aspnet-mvc-3-framework...primerami-na...](http://dom-eknig.ru/aspnet-mvc-3-framework...primerami-na...)).

*Часовских В.П., Стаин Д.А.* Структура, содержание и среда разработки веб-сайта вуза // Эко – Потенциал: журнал мультидисциплинарных научных публикаций. 2013. № 3-4. С. 160-173 (<http://elar.usfeu.ru/handle/123456789/2739>).

*Часовских В.П., Стаин Д.А.* Представление и сравнительный анализ модели образовательного процесса университета в форме реляционных таблиц SQL и их мультипликативное отображение // Техника и технология: новые перспективы развития, научный журнал «Естественные и технические науки». Москва. 2014а. С. 101-106.

*Часовских В.П., Стаин Д.А.* Модель образовательного процесса и сайт вуза 2.0 // Эко–Потенциал: журнал мультидисциплинарных научных публикаций. Екатеринбург. 2014б. № 2(6). С. 113-119 (<http://elar.usfeu.ru/bitstream/123456789/3362/1/Chasovskikh.pdf>).

*Эспозито Д.* Программирование на основе Microsoft ASP.NET MVC. 2-е изд. / Пер. с англ. М.: Издательство «Русская редакция»; СПб.: БХВ-Петербург, 2012. 464 с. («Русская редакция») ([mdk-arbat.ru/bookcard?book\\_id=755050](http://mdk-arbat.ru/bookcard?book_id=755050)).

**Рецензент статьи:** кандидат технических наук,  
профессор кафедры менеджмента и ВЭДП  
Уральского государственного лесотехнического университета М.П. Воронов.