

Уральский государственный экономический университет

Администрирование информационных систем

02.03.03 - Математическое обеспечение и администрирование информационных систем, направленность (профиль) - разработка и администрирование информационных систем

<http://vikchas.ru>

Лабораторная работа 5

Тема «Цели и задачи администрирования информационных систем лабораторных работ»

Часовских Виктор Петрович

доктор технических наук,
Профессор кафедры ШИиКМ
ФГБОУ ВО «Уральский государственный
экономический университет

Екатеринбург 2026

Приоритетные направления научно-технологического развития и перечень важнейших наукоемких технологий РФ

Указ Президента Российской Федерации от 18 июня 2024 г. № 529 "Об утверждении приоритетных направлений научно-технологического развития и перечня важнейших наукоемких технологий" определяет:

Приоритетные направления научно-технологического развития

1. Высокоэффективная и ресурсосберегающая энергетика.
2. Превентивная и персонализированная медицина, обеспечение здорового долголетия.
3. Высокопродуктивное и устойчивое к изменениям природной среды сельское хозяйство.
4. *Безопасность получения, хранения, передачи и обработки информации.*
5. Интеллектуальные транспортные и телекоммуникационные системы, включая автономные транспортные средства.
6. Укрепление социокультурной идентичности российского общества и повышение уровня его образования.
7. Адаптация к изменениям климата, сохранение и рациональное использование природных ресурсов.

Перечень важнейших наукоемких технологий

I. Критические технологии

1. Технологии создания высокоэффективных систем генерации, распределения и хранения энергии (в том числе атомной).
2. Технологии создания энергетических систем с замкнутым топливным циклом.
3. Биомедицинские и когнитивные технологии здорового и активного долголетия.
4. Технологии разработки лекарственных средств и платформ нового поколения (биотехнологических, высокотехнологичных и радиофармацевтических лекарственных препаратов).
5. Технологии персонализированного, лечебного и функционального питания для здоровьесбережения.
6. Технологии разработки медицинских изделий нового поколения, включая биогибридные, бионические технологии и нейротехнологии.
7. Технологии повышения продуктивности (в том числе с помощью селекции) сельскохозяйственных животных и их устойчивости к заболеваниям.

8. Технологии разработки ветеринарных лекарственных средств нового поколения, в том числе для профилактики и лечения инфекционных заболеваний у сельскохозяйственных животных.

9. Технологии получения устойчивых к изменениям природной среды новых сортов и гибридов растений.

10. Технологии создания биологических и химических средств для повышения урожайности сельскохозяйственных культур и их защиты от болезней и вредных организмов (природного или искусственного происхождения).

11. Технологии микроэлектроники и фотоники для систем хранения, обработки, передачи и защиты информации.

12. Технологии защищенных квантовых систем передачи данных.

13. *Технологии создания доверенного и защищенного системного и прикладного программного обеспечения, в том числе для управления социальными и экономически значимыми системами.*

14. Транспортные технологии для различных сфер применения (море, земля, воздух), в том числе беспилотные и автономные системы.

15. Технологии космического приборостроения для развития современных систем связи, навигации и дистанционного зондирования Земли.

16. Технологии системного анализа и прогноза социально-экономического развития и безопасности Российской Федерации в формирующемся миропорядке.

17. Современный инструментарий исследования и укрепления цивилизационных основ и традиционных духовно-нравственных ценностей российского общества, включая историко-культурное наследие и языки народов Российской Федерации.

18. Социально-психологические технологии формирования и развития общественных и межнациональных отношений.

19. Мониторинг и прогнозирование состояния окружающей среды и изменения климата (в том числе ключевых районов Мирового океана, морей России, Арктики и Антарктики), технологии предупреждения и снижения рисков чрезвычайных ситуаций природного и техногенного характера, негативных социально-экономических последствий.

20. Экологически чистые технологии эффективной добычи и глубокой переработки стратегических и дефицитных видов полезных ископаемых.

21. Технологии сохранения биологического разнообразия и борьбы с чужеродными (инвазивными) видами животных, растений и микроорганизмов.

II. Сквозные технологии

22. Технологии, основанные на методах синтетической биологии и генной инженерии.

23. Технологии создания новых материалов с заданными свойствами и эксплуатационными характеристиками.

24. Технологии производства малотоннажной химической продукции, включая особо чистые вещества, для фармацевтики, энергетики и микроэлектроники.

25. Технологии искусственного интеллекта в отраслях экономики, социальной сферы (включая сферу общественной безопасности) и в органах публичной власти.

26. Технологии создания отечественных средств производства и научного приборостроения.

27. Природоподобные технологии.

28. Биотехнологии в отраслях экономики.

Критическая технология - комплекс междотраслевых (междисциплинарных) технологических решений, которые создают предпосылки для дальнейшего развития различных тематических технологических направлений, имеют широкий потенциальный круг конкурентоспособных инновационных приложений в разных отраслях экономики и вносят в совокупности наибольший вклад в реализацию приоритетных направлений развития науки, **технологий** и техники.

Технологии критические – имеют системное значение для решения важных социально-экономических задач и обеспечения обороны и безопасности государства.

Сквозные технологии – перспективные технологии межотраслевого назначения, обеспечивающие создание инновационных продуктов и сервисов и оказывающие существенное влияние на развитие экономики, радикально меняя существующие рынки и (или) способствуя формированию новых рынков.

К сквозным цифровым технологиям относятся:

1. *Большие данные.*
2. *Искусственный интеллект.*
3. Система распределённого реестра (блокчейн).
4. Квантовые технологии.
5. Новые производственные технологии.
6. Промышленный интернет.
7. Робототехника.
8. Сенсорика.
9. Беспроводная связь.
10. *Виртуальная реальность.*

Информационные системы - общие сведения

Информационная система (ИС) ГОСТом РВ 51987 РФ определяется как «автоматизированная система, результатом функционирования которой является представление выходной информации для последующего использования».

В узком смысле информационной системой называют только подмножество компонентов ИС в **широком смысле**, включающее базы данных, СУБД и специализированные прикладные программы.

Федеральным законом РФ «Об информации, информационных технологиях и о защите информации» трактуется понятие об информационной системе, подразумевая совокупность информации, которая содержится в базах данных, и информационных технологий и технических средств, которые обеспечивают ее обработку.

Российские ученые в области информатики также дают определение понятию информационной системы.

Наиболее полное определение дается М.Р. Когаловским, который считает, что в понятие ИС кроме данных, программного и аппаратного обеспечений и людских ресурсов необходимо включить коммуникационное оборудование, лингвистические средства и информационные ресурсы, и все это в совокупности образует систему, которая будет обеспечивать «поддержку динамической информационной модели некоторой части реального мира для удовлетворения информационных потребностей пользователей»

Моё добавление — современные информационные системы в качестве данных погружены в BIG DATA.

13 июля 2023 года Президент РФ принял участие в пленарном заседании Форума будущих технологий «Вычисления и связь. Квантовый мир».

Центральное событие посвящено обсуждению приоритетных задач в развитии технологий вычисления и передачи данных, их потенциала для роста экономики и повышения качества жизни людей.

Указания нашего Президента:

«Наша принципиальная задача — перевести всю экономику, социальную сферу, органы власти, работу органов власти на качественно новые принципы работы, внедрить управление на новых данных — на основе **больших данных**.

Всё, что связано с данными, большими данными, принимает критически важное значение. Речь, по сути, идёт о **системообразующей инфраструктуре для нашего дальнейшего развития**, для будущего нашей экономики в целом. Необходимо в этом году подготовить новый национальный проект **экономики данных**.»

Администрирование информационной системы заключается в предоставлении пользователям соответствующих прав использования возможностей работы с системой (базой данных); обеспечении целостности данных, а также создании многопользовательских приложений.

Выделяют три основные категории пользователей информационной системы: разработчики, администраторы, специалисты по BIG DATA и собственно пользователи.

Администрирование информационной системы осуществляется лицом, управляющим этой системой – администратором. Если информационная система большая, эти обязанности могут выполнять несколько человек (группа администраторов).

Чтобы лучше понять администрирование информационной системы необходимо выполнить лабораторные работы.

Структуру работ определяют две составляющие: первая - изучение реальной информационно системы с целью определения и классификации составляющих администрирования; вторая - подготовка новой информационной системы со всеми выявленными составляющими и новыми компонентами цифровой экономики - большие данные и машинное обучение.

Основной средой лабораторных работ является Visual Studio 2019 (большие данные и машинное обучение представлено в Visual Studio 2019) и более высокие версии.

Будем изучать информационную систему SportsStore (автор разработки Адам Фриман - опытный специалист в области информационных технологий) и будет следовать классическому подходу, который повсеместно используется в онлайн-магазинах.

Мы изучим и создадим онлайн-каталог товаров, который пользователи могут просматривать по категориям и страницам, корзину для покупок, куда пользователи могут добавлять и удалять товары, и форму оплаты, где пользователи могут вводить сведения, связанные с доставкой.

Изучим административную область, которая включает в себя средства создания, чтения, обновления и удаления (create, read, update, delete — CRUD) для управления каталогом товаров, и защитим ее так, чтобы изменения могли вносить только зарегистрированные администраторы.

Изучим все основные составляющие базы данных. Добавления новых составляющих определим по результатам первой составляющей.

Каждая работа начинается с регистрации на первой странице сайта <http://vikchas.ru> и заканчивается кратким отчетом.

Отчет в формате Word Microsoft загружаемый на сайт <http://vikchas.ru> . Формируемый сегмент информационной системы загружается в формате rar на сайт <http://vikchas.ru> .

В качестве чата для общения используется сайт <http://vikchas.ru> . Чат доступен круглосуточно.

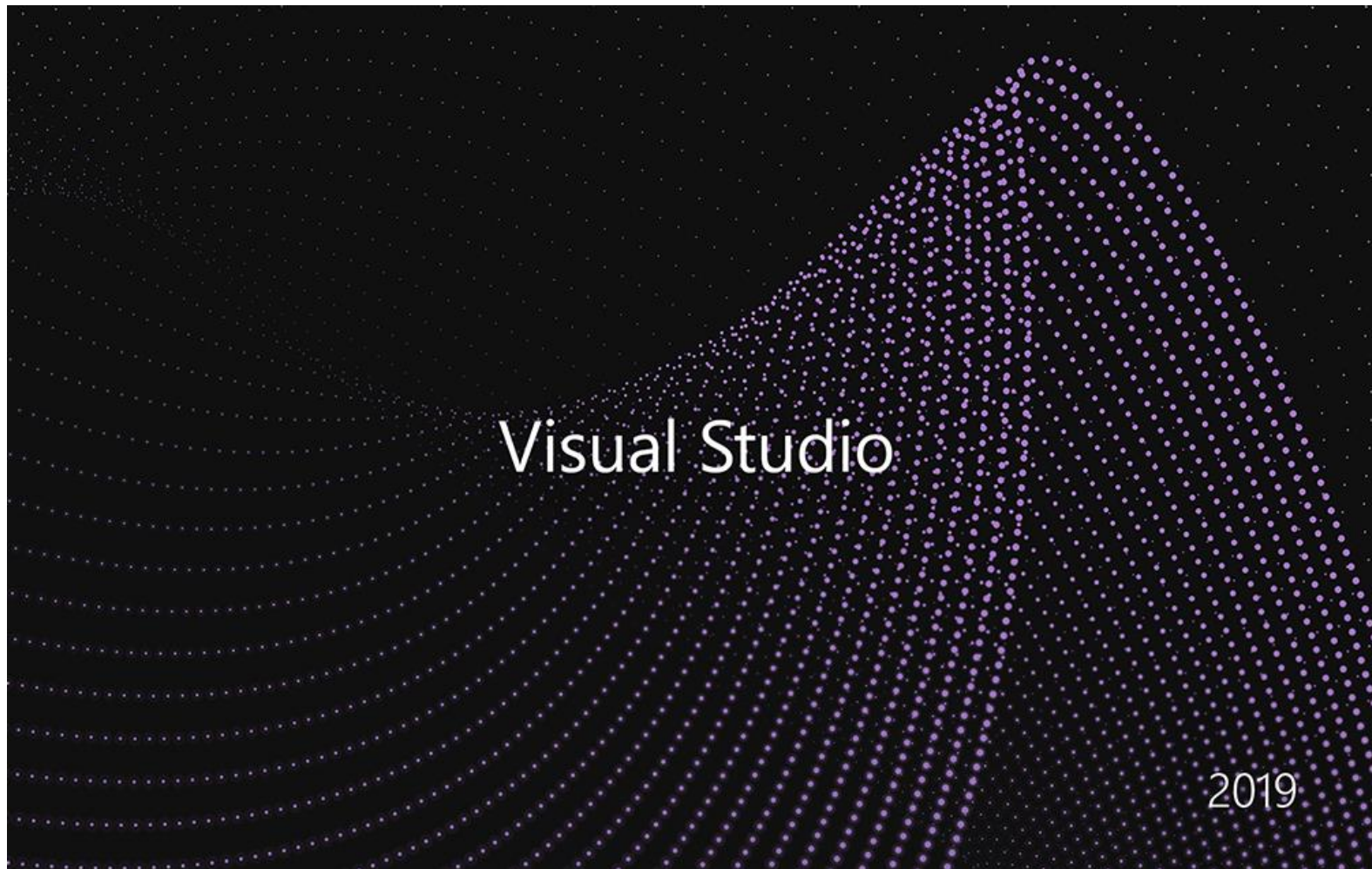
В каждом отчете должна присутствовать таблица из двух столбцов — в первом указываются составляющие администратора, во-втором составляющие пользователя.

Для работ с системой Visual Studio 2019 можно пользоваться методическими указаниями из библиотеки сайта <http://vikchas.ru> .

Для каждой работы есть методические указания в библиотеке сайта <http://vikchas.ru>

Создание нового проекта ASP.NET MVC Visual Studio 2019

Запускаем Visual Studio 2019



Получаем следующий экран монитора

Visual Studio 2019

Открыть последние

▲ Сегодня

-  **WebApplication1.sln** 30.08.2023 11:52
C:\Users\u2007\source\repos\WebApplication1
-  **WorkingWithVisualStudio.sln** 30.08.2023 11:18
C:\Users\u2007\source\repos\WorkingWithVisualStudio
-  **WorkingWithVisualStudio.sln** 30.08.2023 10:40
H:\...\06 - Working with Visual Studio\WorkingWithVisual...
-  **WorkingWi thVisualStudio.sln** 30.08.2023 9:20
C:\Users\u2007\source\repos\WorkingWi thVisualStudio
-  **Testing.sln** 30.08.2023 8:51
H:\...\pro-asp.net-core-3-master\06 - Testing\Testing

Начало работы



Клонирование репозитория

Получить код из интернет-репозитория, например, GitHub или Azure DevOps



Открыть проект или решение

Открыть локальный проект Visual Studio или SLN-файл



Открыть локальную папку

Перейти и изменить код в любой папке



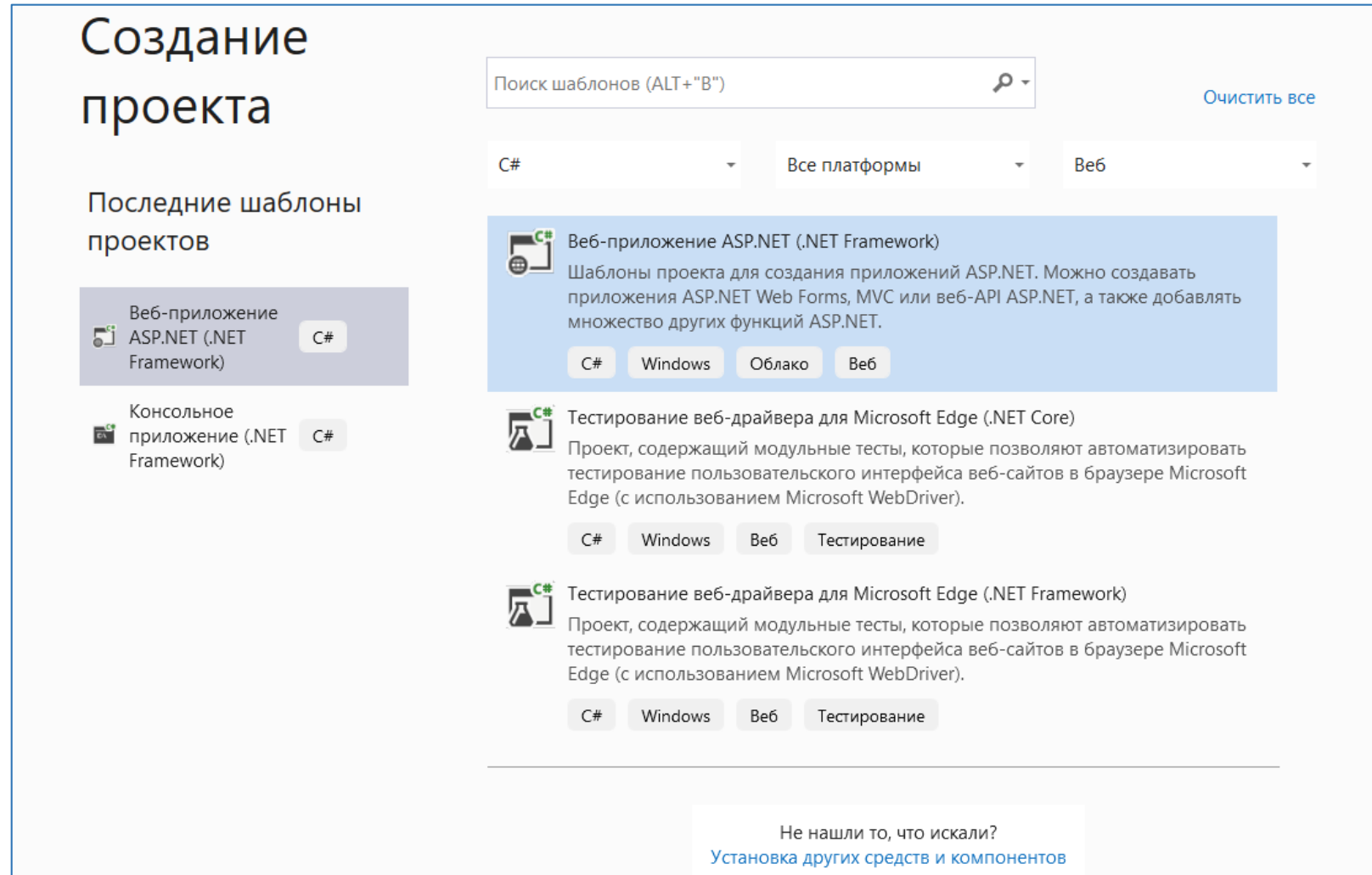
Создание проекта

Выберите шаблон проекта с формированием шаблонов кода, чтобы начать работу

[Продолжить без кода →](#)

В полученной визуализации видим две части, левая показывает список проектов, в которых мы недавно работали. На правой панели вы можете увидеть встроенные опции для создания, клонирования, открытия проекта и решения.

Нажимаем «Создать новый проект», получаем:



Появилась визуализация, на которой мы должны выбрать, какой тип приложения хотим создать. Находим «**ASP.NET Web Application (.NET Framework)**» и нажимаем кнопку «Далее».

Появится новая визуализация, на которой указываем имя вашего проекта и место, где хотим зарезервировать файлы приложения. Остальные настройки останутся такими же, как и на изображении выше, в последнем нажатии на кнопку «Создать». Появится:

Создать веб-приложение ASP.NET

**Пустой**

Пустой шаблон проекта для создания приложений ASP.NET. В этом шаблоне отсутствует какое-либо содержимое.

**Веб-формы**

Шаблон проекта для создания приложений ASP.NET Web Forms. С помощью ASP.NET Web Forms можно создавать динамические веб-сайты, используя привычную модель с перетаскиванием элементов и управлением событиями. Область конструирования, а также сотни элементов управления и компонентов, позволяют быстро создавать сложные и высокоэффективные сайты с пользовательским интерфейсом, которые обеспечивают доступ к данным.

**MVC**

Шаблон проекта для создания приложений ASP.NET MVC. ASP.NET MVC дает возможность создавать приложения с помощью архитектуры Model-View-Controller. ASP.NET MVC включает в себя множество функций, обеспечивающих быструю разработку, управляемую тестами, что дает возможность создавать приложения, использующие самые последние стандарты.

**Веб-API**

Шаблон проекта для создания служб HTTP REST для широкого диапазона клиентов, включая браузеры и мобильные устройства.

Аутентификация

без проверки подлинности
[Изменение](#)

Добавить папки и основные ссылки

☐ Веб-формы

☒ MVC

☐ Веб-API

Дополнительно

☐ Настроить для HTTPS

☐ Поддержка Docker
(требуется [Docker Desktop](#))

☐ Также создать проект для модульных тестов

WebApplication2.Tests

Выбираем «Пустой» и MVC в основных ссылках, а затем нажать кнопку «Создать». Клетка «HTTPS» должна быть пустой.

Создать веб-приложение ASP.NET



Пустой

Пустой шаблон проекта для создания приложений ASP.NET. В этом шаблоне отсутствует какое-либо содержимое.



Веб-формы

Шаблон проекта для создания приложений ASP.NET Web Forms. С помощью ASP.NET Web Forms можно создавать динамические веб-сайты, используя привычную модель с перетаскиванием элементов и управлением событиями. Область конструирования, а также сотни элементов управления и компонентов, позволяют быстро создавать сложные и высокоэффективные сайты с пользовательским интерфейсом, которые обеспечивают доступ к данным.



MVC

Шаблон проекта для создания приложений ASP.NET MVC. ASP.NET MVC дает возможность создавать приложения с помощью архитектуры Model-View-Controller. ASP.NET MVC включает в себя множество функций, обеспечивающих быструю разработку, управляемую тестами, что дает возможность создавать приложения, использующие самые последние стандарты.



Веб-API

Шаблон проекта для создания служб HTTP REST для широкого диапазона клиентов, включая браузеры и мобильные устройства.

Аутентификация

без проверки подлинности

[Изменение](#)

Добавить папки и основные ссылки

- ☐ Веб-формы
- ☒ MVC
- ☐ Веб-API

Дополнительно

- ☐ Настроить для HTTPS
- ☐ Поддержка Docker
(требуется [Docker Desktop](#))
- ☐ Также создать проект для модульных тестов

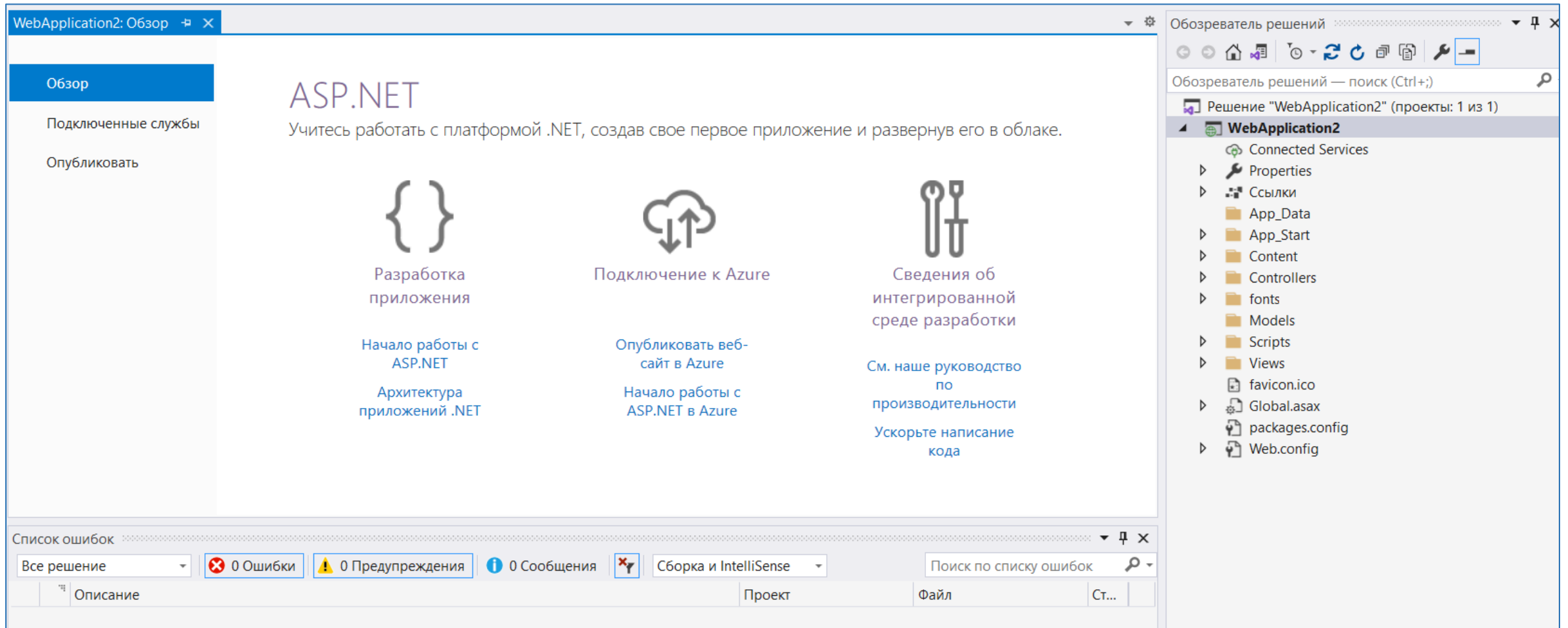
WebApplication2.Tests

Выбрав «Пустой» и **MVC** в **основных ссылках**, а затем нажимаем кнопку «Создать».

Если выберем **ПУСТОЙ**, будет создан новый проект без каких-либо данных по умолчанию.

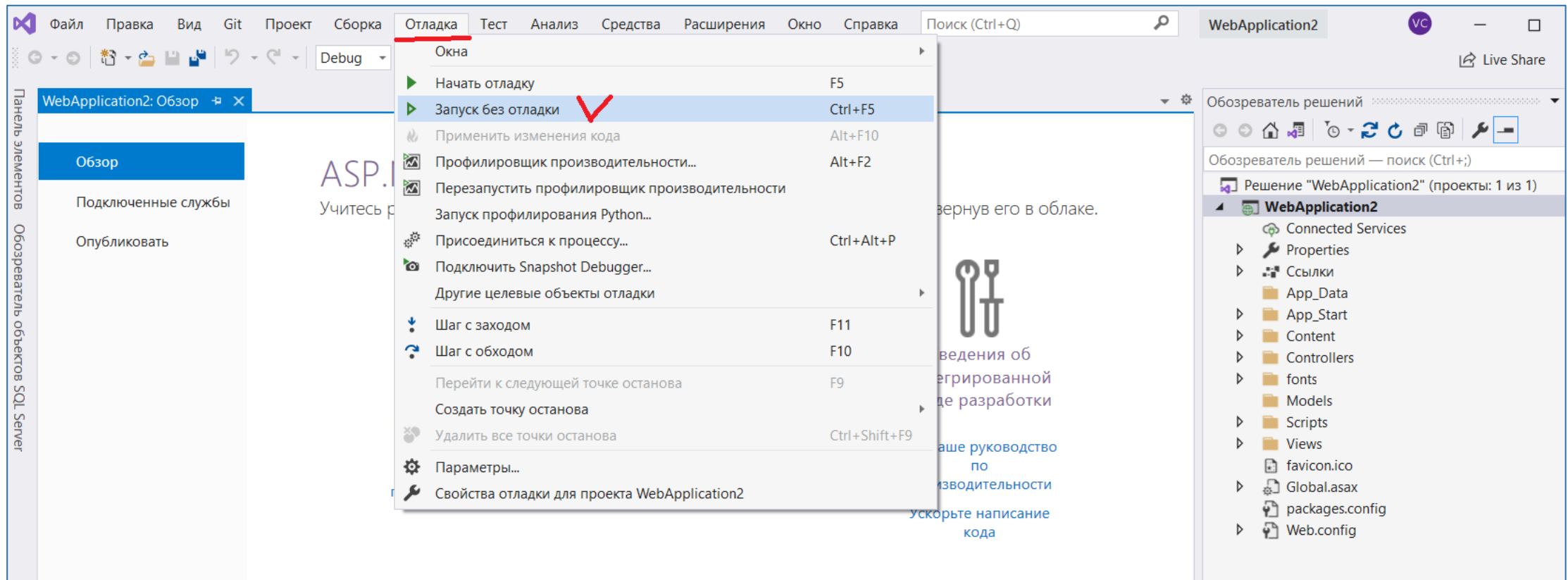
Если выберем **MVC**, проект будет создан с пустой папкой **Model, View, Controller**.

После этого начинается процесс создания проекта, который занимает от 1 до 3 минут в зависимости от характеристик ЭВМ. Появиться:



Видим **созданное веб-приложение ASP.NET MVC**. На правой боковой панели видим папки и файлы. Это основы вашего проекта, технически мы назовем эту правую боковую панель **Обозревателем решений**.

Обращаем внимание на название вашего проекта. Все файлы и папки находятся под этим проектом, можно добавить несколько проектов в одно и то же решение (**такой подход практикуется для разработки универсального решения**). Выполняем наш проект:



И получаем начало работы с нашей информационной системой

ASP.NET

ASP.NET is a free web framework for building great Web sites and Web applications using HTML, CSS and JavaScript.

[Learn more »](#)

Getting started

ASP.NET MVC gives you a powerful, patterns-based way to build dynamic websites that enables a clean separation of concerns and gives you full control over markup for enjoyable, agile development.

[Learn more »](#)

Get more libraries

NuGet is a free Visual Studio extension that makes it easy to add, remove, and update libraries and tools in Visual Studio projects.

[Learn more »](#)

Web Hosting

You can easily find a web hosting company that offers the right mix of features and price for your applications.

[Learn more »](#)

Общие сведения о файлах и папках

Ссылки содержат список библиотек, которые вы используете в своем приложении. Например, если вы используете **Entity Framework**, вы также найдете имя в ссылках.

RouteConfig используется для определения маршрутизации приложения. Вы также можете получить представление из названия, по умолчанию определен один маршрут, но вы можете добавить свои собственные маршруты по своему усмотрению.

Папка **контроллера** используется для размещения **контроллера** приложения, которые используются для управления потоком данных и приложением.

Папка **модели** используется для размещения модели (классов модели, модели представления, объекта передачи данных) приложения, которые используются для передачи данных из базы данных в представление.

View папка используется для размещения представления (html-страниц) приложения, которые используются для визуализации пользовательского интерфейса.

Global.asax, который содержит события более высокого уровня приложения, такие как начальное событие (**Application_Start**).

Packages.config используется для обслуживания пакетов по ссылкам на проекты. Это позволит диспетчеру пакетов NuGet легко восстанавливать зависимость от отдельных проектов при переносе решения на другие серверы.

Файл **Web.config** содержит основную конфигурацию всего приложения. Зависимости приложений и их версии настраиваются здесь автоматически. Даже если вы можете использовать некоторые пользовательские ключи для получения их данных во всем приложении. Каким-то образом вы можете видеть, что в основном разработчики использовали для размещения своей строки подключения к базе данных в этом файле.

Папка **App_Data**

В эту папку помещаются закрытые данные, такие как XML-файлы или базы данных, если используется SQL Server Express, SQLite или другие хранилища на основе файлов

Папка **App_Start**

Эта папка содержит ряд основных настроек конфигурации для проекта, в том числе определение маршрутов и фильтров, а также пакетов содержимого. Маршруты и фильтры рассмотрим позже

Файл **bin**, сюда помещается скомпилированная сборка приложения MVC вместе со всеми ссылаемыми сборками, находящимися не в GAC

Папка **Content** , сюда помещается статическое содержимое, такое как CSS-файлы и изображения. Это является необязательным соглашением. Статическое содержимое можно хранить в любом подходящем месте.

Папка **Controllers**

Сюда помещаются классы контроллеров. Это является соглашением. Классы контроллеров могут размещаться где угодно, поскольку они компилируются в ту же самую сборку.

Папка **Models** Сюда помещаются классы моделей представлений и моделей предметной области, хотя все кроме простейших приложений выигрывают от определения модели предметной области в отдельном проекте.

Это является соглашением. Классы моделей могут быть определены где угодно в текущем проекте или вообще вынесены в отдельный проект

Папка **Scripts** Эта папка предназначена для хранения библиотек JavaScript, используемых в приложении. По умолчанию Visual Studio добавляет библиотеки jQuery и несколько других популярных JavaScript-библиотек. Это является соглашением. Файлы сценариев могут находиться в любом месте, т.к. в действительности они представляют собой просто другой тип статического содержимого.

Папка **Views** В этой папке хранятся представления и частичные представления, обычно сгруппированные вместе в папках с именами контроллеров, с которыми они связаны.

Папка **Views/Shared** В этой папке хранятся компоновки и представления, не являющиеся специфичными для какого-либо контроллера.

Папка **Views/Web.config** Это конфигурационный файл. В нем содержится конфигурационная информация, которая обеспечивает обработку представлений с помощью ASP.NET и предотвращает их обслуживание веб-сервером IIS, а также пространства имен, по умолчанию импортируемые в представления.

Файл в папке **Views/Web.config** предотвращает обслуживание веб-сервером IIS содержимого этих папок. Представления должны визуализироваться через методы действий.

Файл **Global.asax** Это глобальный класс приложения ASP.NET. В его файле отделенного кода (**Global.asax.cs**) регистрируется конфигурация маршрутов, а также предоставляется любой код, который должен выполняться при запуске или завершении приложения либо в случае возникновения необработанного исключения.

Web.config Конфигурационный файл для приложения.

В полученном проекте ИС изучить назначение всех папок и файлов. Произвести изменение в проекте - заменим тексты на английском языке в тексты на русском языке.

Отчет по лабораторной работе

В отчете по лабораторной работе необходимо привести:

- 1.Скриншоты создания ИС в среде VS-2019 с кратким описанием действий и скриншоты замены текстов с английского языка на русский.
- 2.Раскрыть какие профессиональные компетенции необходимы администратору ИС в пространстве Big Data.
- 3.Раскрыть возможности искусственного интеллекта в администрирование данных ИС.

Благодарю за внимание!

