

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО  
ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО «Уральский государственный экономический университет»

В.П. Часовских

# Разработка и поддержка КИС

Екатеринбург 2023

# Создание КИС

Создание КИС — это сложная и важная задача для любой компании «среднего» и «крупного» бизнеса. Здесь важно определить потребности предприятия по регистрации данных его информационных потоков и на основе этого выбрать решения на базе которых будет вестись учет и анализ.

ИС — это вся инфраструктура предприятия, задействованная в процессе управления всеми информационно-документальными потоками.

Главная задача такой системы — эффективное управление всеми ресурсами предприятия (материально-техническими, финансовыми, технологическими и интеллектуальными) для получения максимальной прибыли и удовлетворения материальных и профессиональных потребностей всех сотрудников предприятия. Без определения важных узловых элементов ИС не возможно переходить к этапу проектирования системы.

## Обязательные элементы ИС

- Информационная модель ИС. Включает все формы документов, структуру справочников и данных, и т.д;
- Регламент развития информационной модели и правила внесения в неё изменений;
- Кадровые ресурсы (департамент развития, привлекаемые консультанты), отвечающие за формирование и развитие информационной модели;
- Программное обеспечение, конфигурация которого соответствует требованиям информационной модели;
- Кадровые ресурсы, отвечающие за настройку и адаптацию ПО, и его соответствие утвержденной информационной модели;

- Регламент внесения изменений в настраиваемые структуры (специфические настройки, структуры баз данных и т.д.) и конфигурацию программного обеспечения и состав его функциональных модулей;
- Аппаратно-техническая база: компьютеры, периферия, каналы телекоммуникаций, системное ПО и СУБД;
- Эксплуатационно-технические кадровые ресурсы, включая персонал по обслуживанию аппаратно-технической базы;
- Правила использования программного обеспечения и пользовательские инструкции, регламент обучения и сертификацию пользователей.

Система управления любой компании включает три основные подсистемы:

- Планирование продаж и операций. Это общий план функционирования предприятия, устанавливающий объемы изготовления готовой продукции. Главным здесь является планирование спроса и оценка ресурсов, необходимых для удовлетворения спроса. Здесь же создается основной производственный план, определяющий, какие изделия, в каком количестве и в какие сроки нужно произвести;
- Детальное планирование необходимых ресурсов (материалов, производственных мощностей, трудовых ресурсов и т.д.). Составленный план определяет время и объем заказов для всех материалов и комплектующих, необходимых для реализации основного производственного плана;
- Управление исполнением планов в процессе производства и закупок (снабжения).



Все эти подсистемы реализуются на основе КИС.

Корпоративные информационные системы (КИС) — это интегрированные системы управления территориально распределенной корпорацией, основанные на углубленном анализе данных, широком использовании систем информационной поддержки принятия решений, электронных документообороте и делопроизводстве. КИС призваны объединить стратегию управления предприятием и передовые информационные технологии.

Корпоративная информационная система — это совокупность технических и программных средств предприятия, реализующих идеи и методы автоматизации.

Комплексная автоматизация бизнес процессов предприятия на базе современной аппаратной и программной поддержки может называться по-

разному. В настоящее время наряду с названием Корпоративные информационные системы (КИС) употребляются, например, следующие названия:

- Автоматизированные системы управления (АСУ);
- Интегрированные системы управления (ИСУ);
- Интегрированные информационные системы (ИИС);
- Информационные системы управления предприятием (ИСУП).

### Процесс создания единой корпоративной информационной системы

Создание единой (корпоративной) информационной системы (КИС) — процесс не простой, и кроме определения задач, которые должна решать такая система, важно определить, на базе какого решения она будет функционировать. В России большую долю рынка информационных систем занимают программные продукты на базе «1С».

КИС по своему составу — это совокупность различных программно-аппаратных платформ, универсальных и специализированных приложений различных разработчиков, интегрированных в единую информационно-однородную систему, которая наилучшим образом решает в некотором роде уникальную задачу каждого конкретного предприятия.

То есть, КИС — человеко-машинная система и инструмент поддержки интеллектуальной деятельности человека, которая под его воздействием должна:

- Накапливать определенный опыт и формализованные знания;
- Постоянно совершенствоваться и развиваться;
- Быстро адаптироваться к изменяющимся условиям внешней среды и новым потребностям предприятия.

Комплексная автоматизация предприятия подразумевает перевод в плоскость компьютерных технологий всех основных деловых процессов организации. И использование специальных программных средств, обеспечивающих информационную поддержку бизнес-процессов, в качестве основы КИС представляется наиболее оправданным и эффективным.

Современные системы управления деловыми процессами позволяют интегрировать вокруг себя различное программное обеспечение, формируя единую информационную систему. Тем самым решаются проблемы координации деятельности сотрудников и подразделений, обеспечения их необходимой информацией и контроля исполнительской дисциплины, а руководство получает своевременный доступ к достоверным данным о ходе производственного процесса и имеет средства для оперативного принятия и воплощения в жизнь своих решений. И, что самое главное, полученный автоматизированный комплекс представляет собой гибкую открытую структуру, которую можно перестраивать на лету и дополнять новыми модулями или внешним программным обеспечением.

Под корпоративной информационной системой будем понимать информационную систему организации, отвечающую следующему минимальному перечню требований:

- Функциональная полнота системы;
- Надежная система защиты информации;
- Наличие инструментальных средств адаптации и сопровождения системы;
- Обеспечение обмена данными между разработанными информационными системами и др. программными продуктами, функционирующими в организации;
- Возможность консолидации информации;
- Наличие специальных средств анализа состояния системы в процессе эксплуатации.

## Функциональная полнота системы

- Выполнение международных стандартов управленческого учета MRP II, ERP, CSRP;
- Автоматизация в рамках системы решения задач планирования, бюджетирования, прогнозирования, оперативного (управленческого) учета, бухгалтерского учета, статистического учета и финансово-экономического анализа;
- Формирование и ведение учета одновременно по российским и международным стандартам;
- Количество однократно учитываемых параметров деятельности организации от 200 до 1000, количество формируемых таблиц баз данных — от 800 до 3000.

## Система защиты информации



- Парольная система разграничения доступа к данным и реализуемым функциям управления;
- Многоуровневая система защиты данных (средства авторизации вводимой и корректируемой информации, регистрация времени ввода и модификации данных).

## **Инструментальные средства адаптации и сопровождения системы**

- Изменение структуры и функций бизнес-процессов;
- Изменение информационного пространства;
- Изменение интерфейсов ввода, просмотра и корректировки информации;
- Изменение организационного и функционального наполнения рабочего места пользователя;
- Генератор произвольных отчетов;
- Генератор сложных хозяйственных операций;
- Генератор стандартных форм.

## **Возможность консолидации информации**

- На уровне организации — объединение информации филиалов, холдингов, дочерних компаний и т.д.;
- На уровне отдельных задач — планирования, учета, контроля и т.д.;
- На уровне временных периодов — для выполнения анализа финансово-экономических показателей за период, превышающий отчетный.

## **Специальные средства анализа состояния системы в процессе эксплуатации**

- Анализ архитектуры баз данных;

- Анализ алгоритмов;
- Анализ статистики количества обработанной информации;
- Журнал выполненных операций;
- Список работающих станций серверов;
- Анализ внутрисистемной почты.

Наиболее развитые корпоративные ИС (КИС) предназначены для автоматизации всех функций управления корпорацией: от научно-технической и маркетинговой подготовки ее деятельности до реализации ее продукции и услуг. В настоящее время КИС имеют в основном экономическую и производственную направленность.

Корпоративная информационная система (КИС) является подсистемой и основой «Системы управления предприятием и бизнеса».

- КИС основывается на обработке объективных данных, характеризующих состояние производственных и бизнес-процессов;
- КИС разрабатывается либо как элемент общей системы управления предприятием, либо как начальная фаза создания системы управления проектами.

Стоимость разработки такой информационной системы — от **200 000 руб.** и включает в себя детальную проработку приведенных ниже этапов.

## **Этапы работ при разработке КИС**

- Разработка концепции КИС;
- Техническое задание;
- Технический проект;
- Рабочая документация;
- Ввод в действие.

Корпоративные информационные системы должны обладать следующими свойствами:

- Соответствие тем функциональным запросам, которые возникают в бизнесе клиента;
- Возможность интеграции различных модулей системы;
- Возможность реализации распределенных вычислений, позволяющих одновременно решать одну проблему на разных рабочих местах;
- Возможность автоматического управления сложным бизнесом, осуществляемым из общего административного центра;
- Надежность системы, ее защищенность от случайных сбоев, потери информации или проникновения нежелательных лиц к системе управления компании.

Последовательность работ, связанных с определением целесообразности создания и промышленной эксплуатации КИС, оформлена в виде процесса (создания или изготовления), который имеет иерархическое описание и состоит из стадий, каждая из которых включает в себя этапы, а они, в свою очередь, — виды работ.

В качестве фрагмента рассмотрим подробнее содержание процесса создания и внедрения КИС, который включает в себя следующие стадии, этапы и некоторые виды работ.

## Стадия 1.1 «Формирование требований и 1.2 «Разработка концепции»

Основная цель этапов и работ этих стадий заключается в формировании обоснованного с позиций заказчика предложения о создании КИС с определенными основными функциями и техническими характеристиками.

Основными выходными документами этой стадии являются: отчеты и технико-экономическое обоснование целесообразности создания КИС с выбранными функциями и их характеристиками; заявка на создание корпоративных информационных систем и исходные технические требования к КИС в объеме, соответствующем ГОСТу.

## Стадия 2.1 «Техническое задание» и 2.2 «Эскизное проектирование»

Основными целями стадии являются: подтверждение целесообразности и детальное обследование возможности создания эффективной ИС с функциями и техническими характеристиками, сформулированными в виде исходных технических требований к системе; планирование совокупности всех НИР, ОКР (опытно-конструкторские работы), проектных и монтажно-наладочных работ, сроков их выполнения и организаций исполнителей; подготовка всех материалов, необходимых для проведения проектных работ.

Выходными документами стадии являются: техническое задание на создание ИС, научно-технический отчет, содержащий результаты проведенных предпроектных исследований; эскизный проект ИС.

Техническое задание— документ, определяющий цели, требования и основные исходные данные, необходимые для разработки автоматизированной системы управления.

В рамках разработки технического задания решаются следующие задачи:

- Устанавливается общая цель создания информационной системы, определяется состав подсистем и функциональных задач;

- Разрабатываются и обосновываются требования, предъявляемые к подсистемам, математическому и программному обеспечению, комплексу технических средств (включая средства связи и передачи данных);
- Устанавливаются общие требования к проектируемой системе;
- Определяется перечень задач по созданию системы и исполнителей;
- Определяются этапы создания системы и сроки их выполнения;
- Проводится предварительный расчет затрат на создание системы и определяется уровень экономической эффективности ее внедрения.

## Стадия 2.3 «Технический проект»

Целями работ, выполняемых на этой стадии, являются разработка корпоративной информационной системы: ее основных технических решений и окончательное определение ее сметной стоимости.

Работы этой стадии завершаются разработкой: общесистемных решений, необходимых и достаточных для выпуска эксплуатационной документации на систему в целом; проектно-сметной документации, входящей в состав раздела «Автоматизация» технического проекта строительства; проектов заявок на разработку новых технических средств; документации специального математического и технического обеспечений, включая техническое задание на программирование. Основные результаты работ стадии оформляются в виде технического проекта.

Технический проект — уникальный процесс, состоящий из совокупности скоординированной и управляемой деятельности с начальной и конечной датами, предпринятый для достижения цели, соответствующей конкретным требованиям, включающий ограничения сроков, стоимости и ресурсов.

К важнейшим характеристикам проекта относятся технико-экономические показатели: объем работ; сроки выполнения; себестоимость; экономическая эффективность, обеспечиваемая реализацией проекта; социальная и

общественная значимость проекта. Любой проект характеризуется классом, типом и масштабом.

Основное назначение технического проекта — это выработка и документальное оформление окончательных проектных решений по изделию или системе.

#### Стадия 2.4 «Рабочая документация»

Целью работ, выполняемых на этой стадии, является выпуск рабочей документации на создаваемую систему. Стадия завершается выпуском рабочего проекта ИС, состоящего из проектной документации, необходимой и достаточной для приобретения, монтажа и наладки комплекса технических средств системы, и документации программного и организационного обеспечений, необходимых и достаточных для наладки и эксплуатации системы.

Рабочая документация — документация, которая разрабатывается в целях реализации в процессе строительства архитектурных, технических и технологических решений.

Состав и содержание рабочей документации должны определяться заказчиком в зависимости от степени детализации решений, содержащихся в проектной документации, и указывается в задании на проектирование.

#### Стадия 2.5 «Внедрение»

Цель стадии и главный результат работ — передача действующей системы в промышленную эксплуатацию, а также получение объективных и систематизированных данных о качестве созданной системы, текущем состоянии и реальном эффекте функционирования системы на основании опыта ее промышленной эксплуатации.

Анализ функционирования выполняется также в ходе промышленной эксплуатации. Для этого определяются показатели эксплуатационной надежности для системы в целом и отдельных реализуемых ею функций, показатели технико-экономической эффективности системы, функционально-алгоритмическая полнота (развитость) системы и социально-психологическая подготовка персонала системы.

# Администрирование корпоративных информационных систем

<http://vikchas.ru>

## Лабораторные работы

Тема «Платформа создания корпоративных информационных систем» (материал

**Часовских Виктор Петрович**

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

Екатеринбург 2023

# 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** корпоративной 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 корпоративной информационной системы: разработчики, администраторы, специалисты по BIG DATA и собственно пользователи.

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

Чтобы лучше понять администрирование корпоративной информационной системы необходимо выполнить лабораторные работы.

Структуру работ определяют две составляющие: первая - изучение реальной корпоративной информационной 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 корпоративную 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 результату первой составляющей.

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

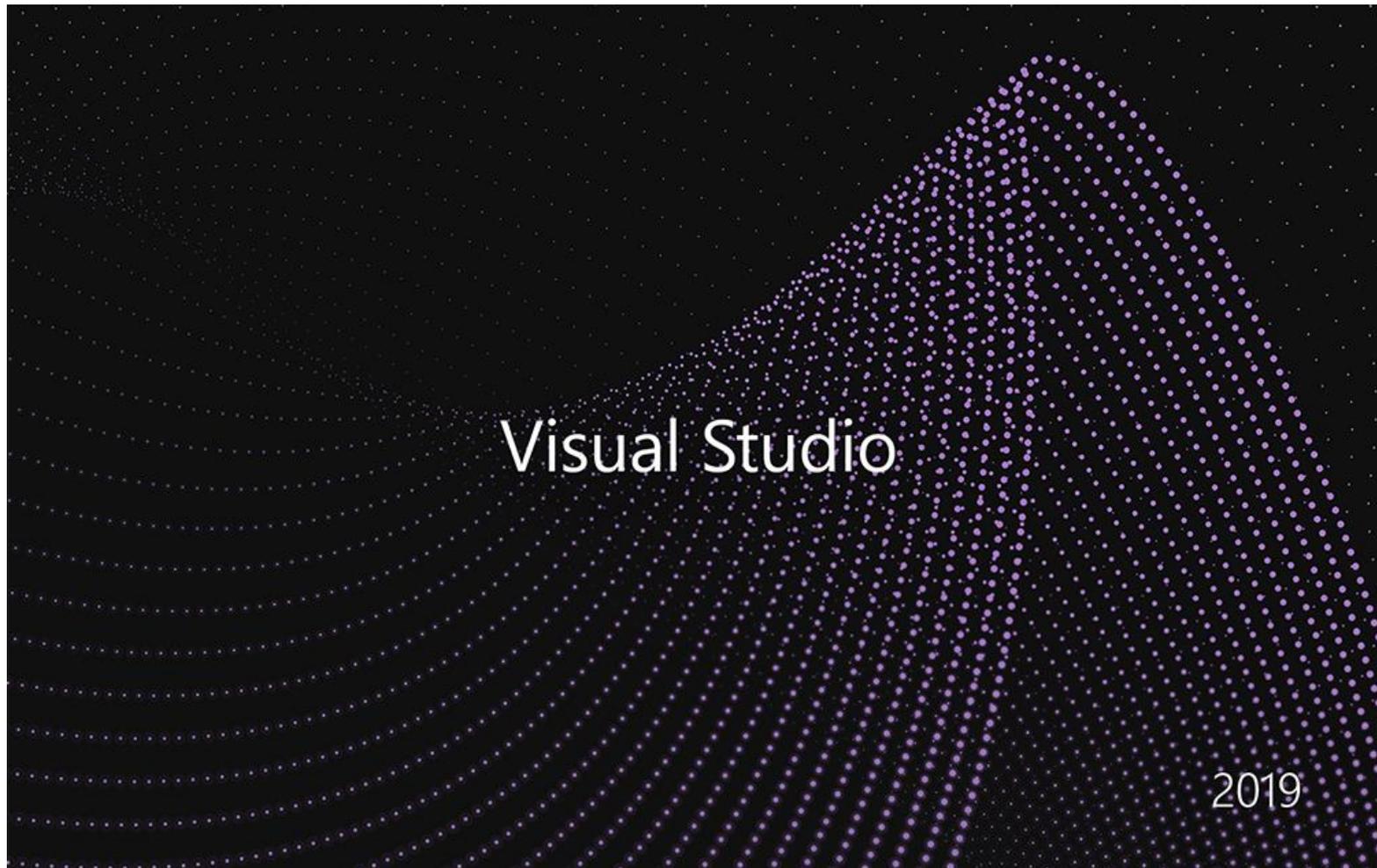
Отчет в формате Word Microsoft загружаемый на сайт <http://vikchas.ru> . Формируемый сегмент информационной системы загружается в формате rar на сайт <http://vikchas.ru> .

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.

# 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...
-  **WorkingWithVisualStudio.sln** 30.08.2023 9:20  
C:\Users\u2007\source\repos\WorkingWithVisualStudio
-  **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:

**Создание проекта**

Поиск шаблонов (ALT+"B") Очистить все

С# Все платформы Веб

**Последние шаблоны проектов**

- Веб-приложение ASP.NET (.NET Framework) C#
- Консольное приложение (.NET Framework) C#

**Веб-приложение ASP.NET (.NET Framework)**  
Шаблоны проекта для создания приложений ASP.NET. Можно создавать приложения ASP.NET Web Forms, MVC или веб-API ASP.NET, а также добавлять множество других функций ASP.NET.  
C# Windows Облако Веб

**Тестирование веб-драйвера для Microsoft Edge (.NET Core)**  
Проект, содержащий модульные тесты, которые позволяют автоматизировать тестирование пользовательского интерфейса веб-сайтов в браузере Microsoft Edge (с использованием Microsoft WebDriver).  
C# Windows Веб Тестирование

**Тестирование веб-драйвера для Microsoft Edge (.NET Framework)**  
Проект, содержащий модульные тесты, которые позволяют автоматизировать тестирование пользовательского интерфейса веб-сайтов в браузере Microsoft Edge (с использованием Microsoft WebDriver).  
C# Windows Веб Тестирование

Не нашли то, что искали?  
[Установка других средств и компонентов](#)

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

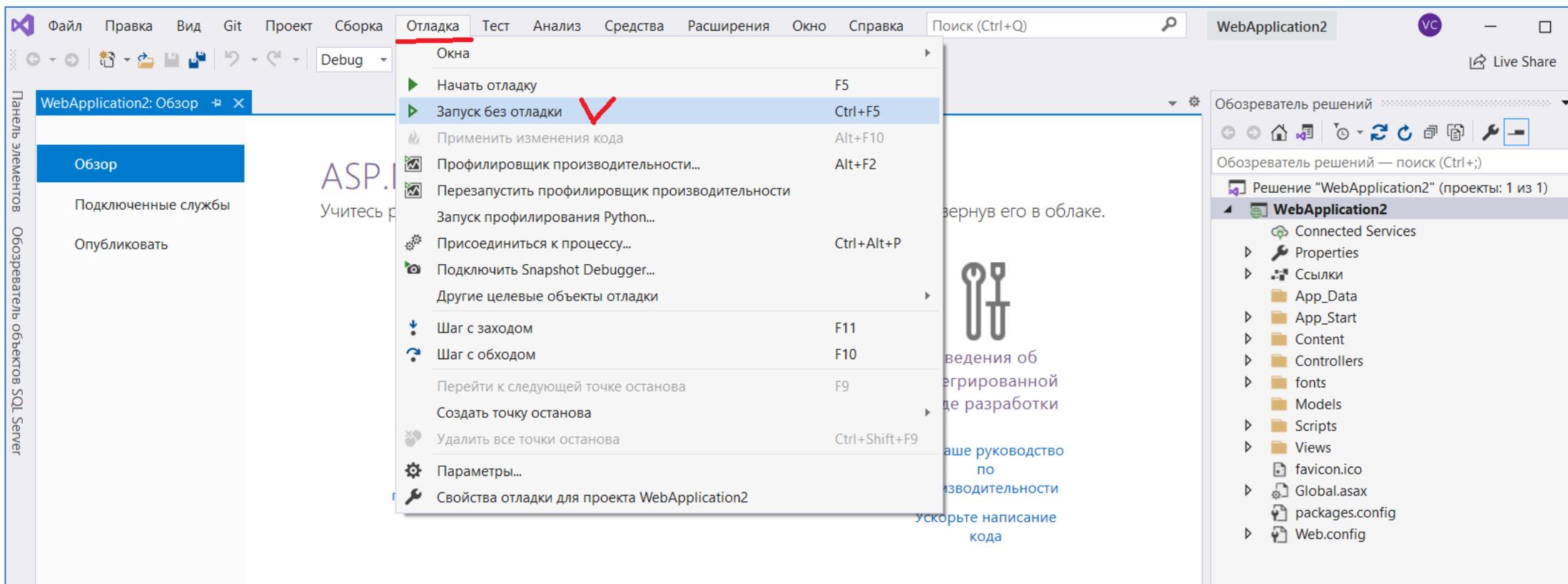
The screenshot shows the Visual Studio interface for a new ASP.NET project named 'WebApplication2'. The main content area displays the ASP.NET logo and the text: 'Учитесь работать с платформой .NET, создав свое первое приложение и развернув его в облаке.' Below this, there are three main sections:

- Разработка приложения** (Application Development): Includes links for 'Начало работы с ASP.NET' (Getting started with ASP.NET) and 'Архитектура приложений .NET' (.NET application architecture).
- Подключение к Azure** (Connect to Azure): Includes links for 'Опубликовать веб-сайт в Azure' (Publish website to Azure) and 'Начало работы с ASP.NET в Azure' (Getting started with ASP.NET in Azure).
- Сведения об интегрированной среде разработки** (Information about the integrated development environment): Includes links for 'См. наше руководство по производительности' (See our performance guide) and 'Ускорьте написание кода' (Speed up code writing).

The left sidebar contains 'Обзор' (Overview), 'Подключенные службы' (Connected services), and 'Опубликовать' (Publish). The right sidebar shows the 'Обозреватель решений' (Solution Explorer) with a tree view of the project files, including folders like 'App\_Data', 'App\_Start', 'Content', 'Controllers', 'fonts', 'Models', 'Scripts', 'Views' and files like 'favicon.ico', 'Global.asax', 'packages.config', and 'Web.config'. The bottom status bar shows 'Список ошибок' (Error List) with 0 errors and 0 warnings, and a search box for errors.

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 **Обзревателем 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.

# Благодарю за внимание!

