

# Администрирование корпоративных информационных систем

<http://vikchas.ru>

Лекция 1 «Цели и задачи администрирования корпоративных информационных систем»

**Часовских Виктор Петрович**

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

Екатеринбург 2024

# 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).

# Понятие корпоративной информационной системы

Современная практика проектирования информационных систем основывается на процессном подходе. Этот подход тесно связан с понятием «бизнес-процесс».

**Бизнес-процесс** - это последовательность действий, имеющая целью получение заданного результата, полезного для организации.

**Бизнес-процесс** - это последовательность действий, имеющая целью получение заданного результата, полезного для организации.

**Корпоративные информационные системы (КИС)** - это программные комплексы, управляющие информационными потоками всех бизнес-процессов организации.

КИС позволяют существенно повысить внутреннюю управляемость организацией. Следствием этого становится увеличение объемов продаж, снижение себестоимости, оптимизация складских запасов, сокращение сроков выполнения заказов, повышение

качества взаимодействия с контрагентами и, в конечном счете, повышение прибыли. Условно качества взаимодействия с контрагентами и, в конечном счете, повышение прибыли. Условно КИС можно разделить на четыре типа:

- локальные КИС;
- КИС управления финансами;
- средние интегрированные КИС;
- крупные интегрированные КИС.

**Локальные КИС** направлены на автоматизацию учета в одном направлении. Этим направлением может быть бухгалтерия, складской учет, сбыт, кадровый учет и т.д. Локальные системы отличаются универсальностью, их внедрение занимает минимум времени.

**КИС управления финансами** интегрируют деятельность предприятия по всем направлениям, которые оказывают влияние на финансовый учет. Такие КИС могут

содержать даже основные возможности для управления производством. Но их основная задача это все-таки не управление производством, а управление финансами. Поэтому подходят такие системы больше всего для финансового учета на непроизводственных предприятиях.

**Средние интегрированные КИС** предназначены, прежде всего, для производственного планирования и управления запасами на производственных предприятиях. Учетные функции больше всего для финансового учета на непроизводственных предприятиях.

**Средние интегрированные КИС** предназначены, прежде всего, для производственного планирования и управления запасами на производственных предприятиях. Учетные функции в них присутствуют, но не играют основной роли. Часто в таких КИС нельзя выделить отдельно модуль бухгалтерии, потому что данные актуальные для бухгалтерского учета поступают из различных модулей: модуля

складского учета, производственного модуля, модуля сбыта и т.д. Поскольку система интегрирует всю деятельность предприятия, ее внедрение может занять несколько месяцев и даже год.

**Крупные интегрированные КИС** отличаются от средних наличием поддержки процессов управления целыми группами предприятий в единой системе. Внедрение таких КИС актуально для холдингов и финансово-промышленных групп. Процесс внедрения, как правило, занимает больше года.

## Концепции КИС

Существуют следующие концепции, применяемые при построении КИС:

- **MPS** (Master Planning Schedule) - концепция объемно-календарного планирования. Она является основой всех КИС, основной задачей которых является планирование. Применяется главным образом на производственных предприятиях.

**MRP** (Material Requirements Planning) - концепция планирования потребности в

- материальных ресурсах. Основная задача - определять потребности в ресурсах на основе результатов объемно-календарного планирования и подробных спецификаций состава производимой продукции.

- **CRP** (Capacity Requirements Planning) - концепция планирования производственных результатов объемно-календарного планирования и подробных спецификаций состава производимой продукции.

- **CRP** (Capacity Requirements Planning) - концепция планирования производственных мощностей. В этой концепции основой является не спецификация состава изделия, а спецификация технологических операций, необходимых при производстве изделия. Система оперирует понятиями «производящее подразделение», «рабочий центр», «занятость рабочих центров», «последовательность операций». Как правило, методологии CRP и MRP используются совместно ввиду их тесной логической связи.

**FRP** (Finance Requirements Planning) - концепция планирования финансовых

- ресурсов.
  - **MRP II** (Manufacturing Resources Planning) - концепция планирования ресурсов производства. Включает MRP, CRP и FRP во взаимосвязи.
  - **ERP** (Enterprise Resources Planning) - концепция планирования ресурсов предприятия. Эта методология может сочетать MPS или MRP с CRP или FRP. В ERP есть много общего с MRP II, но ERP несколько гибче. Если MRP II предполагает применение исключительно в производственной сфере, то ERP применима и в производственной сфере, и в торговле, и в сфере услуг.

**CSRP** (Customer Synchronized Resources Planning) - планирование ресурсов синхронизированное с клиентом. Эта концепция включает в себя всю цепь от проектирования будущего продукта на основе требований заказчика до продажи и последующего сервисного обслуживания. Главный смысл CSRP-систем это высокая степень включенности заказчика в производственный процесс. Обычно заказчик сам

- размещает заказ в системе из личного кабинета и имеет возможность отслеживать его состояние. Предприятию же система дает много ценной информации о тенденциях спроса.
- **SCM** (Supply Chain Management) - концепция управления цепочками поставок. SCM-системы ориентированы на оптимизацию логистических цепей. Как правило, является составной частью ERP и MRPI 1-систем.
- **CRM** (Customer Relationship Management) -управление обслуживанием клиентов.

Применяется на предприятиях общественного питания, супермаркетах. Также может быть составной частью ERP и MRPI 1-систем.

## Какую систему следует выбирать

Конечная цель внедрения КИС для всех организаций одинакова - повышение прибыли посредством увеличения управляемости. Однако, путь к этой цели может быть разным для разных компаний. Одна компания для увеличения прибыли может нуждаться в

выходах на новый рынок, другая в изменении ассортимента, третья в более тесном сотрудничестве с клиентами, четвертая в изменении взаимодействия с поставщиками и т.д. Поэтому хорошая КИС для данной компании - это такая КИС, которая наилучшим образом поддерживает бизнес-процессы необходимые для выбранной стратегии развития.

Не рекомендуется при выборе КИС стремиться внедрить наиболее полную систему, которая поддерживает все существующие и потенциально возможные процессы. Многие ее функции могут никогда не понадобиться, а финансовые вложения во внедрение не окупятся.

## КИС как управленческая идеология

Корпоративные информационные системы в целом можно рассматривать как управленческую идеологию, которая реализована на основе информационных систем и использует ключевые бизнес-стратегии организации или предприятия. Эти системы предназначены в первую очередь для упрощения выполнения задач по систематизации информации и автоматизации различных процессов, связанных с хранением и обработкой данных.

Преобладающее большинство средних и крупных компаний рано или поздно сталкиваются с проблемой, когда обычные офисные программные продукты (например, программы пакета MS Office), используемые для бухгалтерского, управленческого и других видов учета, не справляются с возросшим объемом данных.

Для охвата всех аспектов хозяйственной деятельности организации требуется внедрение КИС, которая позволит повысить эффективность функционирования предприятия в целом – от работы рядовых сотрудников, до менеджеров любого звена.

В качестве наглядного примера можно рассмотреть деятельность мелкого фермера, имеющего в своем распоряжении несколько соток земли, с обработкой которой он может справиться собственными силами. Расширив свои землевладения до нескольких гектаров, он будет нуждаться в сельхозтехнике, которая в некоторой степени автоматизирует процессы: вспашка земли, посадка, сборка урожая. Подобно сельхозтехнике, КИС упрощает и автоматизирует выполнение большинства процессов на предприятии.

# Основные цели и задачи внедрения корпоративной информационной системы

Ключевая задача приобретения и внедрения корпоративной информационной системы состоит в получении надежного инструмента для управления предприятием и повышения эффективности управленческих процессов.

Например, КИС для промышленных компаний позволяет собирать информацию о производственных процессах, финансовых операциях, закупках сырья, рынках сбыта. На основе этой информации управленцы могут вносить корректировки в деятельность предприятия.

Таким образом, КИС позволяет увидеть предприятие «изнутри», оценить эффективность функционирования основных систем, выявить

«лишние» издержки и факторы, мешающие увеличению прибыльности. Особую ценность КИС представляет для руководства холдингов, так как система позволяет консолидировать данные, поступающие из филиалов, осуществлять дистанционный мониторинг деятельности всех подразделений.

**К типовым целям внедрения КИС относят:**

- Получение быстрого доступа ко всем необходимым данным в удобочитаемом формате, понятном для руководителей любого звена.
- Упрощение процессов регистрации, хранения и обработки информации.
- Формирование единого информационного пространства, объединяющего все уровни управления предприятием.

- Уменьшение финансовых и трудовых затрат на организацию процессов планирования, управления и учета деятельности компании.
- Устранение проблем, связанных с предоставлением недостоверных данных, а также дублированием информации.
- Автоматизация процессов консолидации информации на предприятиях с распределенной организационной структурой (компаний, имеющих филиалы и подразделения).

Главное, что должны понимать владельцы организаций и предприятий, это то, что КИС не способствует повышению прибыльности напрямую. Основная задача системы заключается в ускорении и автоматизации процессов обработки данных для получения более точной и оперативной информации, необходимой для принятия ключевых решений.

# Основные проблемы внедрения корпоративной информационной системы

На текущий момент доступен широкий выбор программного обеспечения, предназначенного для использования в качестве автоматизированных систем управления организацией. Каждый из программных комплексов предназначен для решения тех или иных задач, имеет свои преимущества и недостатки. Тем не менее, не каждое внедрение КИС оказывается успешным, о чем свидетельствует мировая статистика. Основные сложности с внедрением информационных систем связаны с тремя ключевыми аспектами:

Возможности системы, заложенные ее разработчиками, и ее конфигурация.

Уровень подготовки специалистов, занимающихся внедрением КИС.

Степень готовности предприятия к внедрению и особенности его структуры.

Первая проблема, вызывающая сложности во внедрении системы, - выбор КИС без учета специфики функционирования предприятия. В наибольшей степени это касается специфики ведения производственного учета, что должно быть отражено в выбираемой КИС. Например, учитывая тип производства (непрерывное, единичное, массовое и т. д.), требования к системе могут различаться в значительной степени. Учесть все особенности работы организации до внедрения системы можно только путем проведения полного обследования ее бизнес-процессов. При этом предпочтительным вариантом являются специалисты извне, так как собственные сотрудники зачастую не имеют возможность дать объективную оценку.

Вторая проблема напрямую связана с возможностями самой системы, а именно – ее гибкостью. Поскольку предприятия функционируют в изменчивой среде, в которой увеличивается конкуренция, вводятся дополнительные нормы и правила, зачастую приходится вносить корректировки в бизнес-процессы. Поэтому, внедряемая система должна предоставлять возможности в настройке и модификации собственной конфигурации.

Третья проблема связана с квалификацией команды, которая занимается внедрением. Поскольку любая КИС представляет собой весьма сложное программное решение, для успешного внедрения требуется объединение усилий как разработчиков, так и представителей самого предприятия. Во-первых, любая система не может функционировать самостоятельно. Для управления ее работой требуется квалифицированный персонал, который в идеале должен получить все необходимые навыки на этапе внедрения системы. Во-вторых, разработчик

или поставщик КИС не должен перекладывать ответственность по внедрению системы на заказчика. При этом особое значение имеет квалификация исполнителей – участие малоопытных специалистов может привести к неудовлетворительном результату.

Четвертая проблема напрямую связана с самим предприятием, а именно – некорректной постановки задачи его руководством. Во-первых, некоторые владельцы компаний имеют чрезмерно завышенные ожидания, предполагая, что внедрение системы позволит мгновенно превратить убыточный бизнес в процветающий. Во-вторых, часто возникает ситуация, когда руководитель самоустраняется от решения задач по внедрению КИС, перекладывая их на плечи руководителей низшего звена.

Кроме того, причиной негативного результат внедрения может быть отказ от реорганизации бизнес-процессов.

Учитывая все вышесказанное, можно выделить ключевые факторы, влияющие на успех внедрения КИС на предприятиях любых масштабов:

- Насколько представители руководящего звена понимают цели и задачи внедрения системы?
- Принимают ли участие в процессе внедрения специалисты как со стороны разработчика (поставщика) программного продукта, так и самого предприятия?
- Готово ли руководство компании к внесению изменений в бизнес-процессы в связи со внедрением КИС?
- Насколько глубоко понимание специфики работы внедряемой системы и ее возможностей?

Кроме всего, один из этапов внедрения должен предусматривать тестовый период опытной эксплуатации системы с полным охватом всех участков учетной и управленческой деятельности. Это лучший способ проверки степени пригодности

системы с относительно невысокими затратами в сравнении с убытками, которые возникнут при полноценном внедрении неподходящего варианта КИС.

## **Преимущества внедрения корпоративной информационной системы**

Как известно, глобальной целью внедрения КИС является увеличение эффективности работы организации в целом. Окупаемость системы рассчитать крайне сложно, поскольку она способствует снижению трудноизмеримых транзакционных затрат и оптимизирует структуру организации в целом. Таким образом, дать количественную оценку приросту доходности предприятия, на котором проводилось внедрение информационной системы, не представляется возможным. Однако, можно оценить степень влияния на логистику, поскольку КИС способствует оптимизации материальных потоков и уменьшению потребности

в оборотных средствах. Система финансового контроллинга, поставленная на базе корпоративной информационной системы, позволяет уменьшить накладные затраты, исключить нерентабельные продукты и малоэффективные (или убыточные) подразделения.

Исходя из вышесказанного, можно сформулировать несколько ключевых преимуществ внедрения КИС:

- Оперативные и достоверные данные о деятельности компании в целом и ее конкретных подразделениях;
- Прирост эффективности управленческих процессов;
- Снижение временных и трудовых ресурсов на выполнение большинства рабочих операций;
- Улучшение общих показателей эффективности.

# ERP как корпоративная информационная система

ERP является одной из разновидностью корпоративных информационных систем, которая предназначена для создания единого информационного пространства организации и обеспечения эффективного управления всеми ее ресурсами, включая производственные процессы, ведение продаж и учет заказов. ERP обеспечивают полноценный контроль для операционной и административной деятельностью предприятий, создавая единую цепочку, в которую входят: учет финансовой деятельности, планирование, управление материальными потоками, производственные процессы, сбыт продуктов, сотрудничество с поставщиками сырья и партнерскими организациями.

Простыми словами, основная цель ERP – держать под полным контролем все ресурсы предприятия, объединяя все его части в единый механизм. В отличие от CRM и прочих типов КИС, ERP позволяет отладить работу всей компании в целом, а не только отдела продаж. Наиболее востребованными сферами для внедрения ERP-систем являются торговля, машиностроение, пищевая и химическая промышленность, а также строительство. **Результаты внедрения КИС ERP**

Данная система позволяет объединить все бизнес-процессы в рамках единой системы, в чем и заключается ее ключевое преимущество. В результате внедрения ERP достигают следующих эффектов:

- Все данные становятся более доступными без необходимости дополнительной сверки или согласования. Таким образом, владелец получает полноценное представление о текущем состоянии его бизнеса.
- Благодаря единой БД, все данные являются согласованными и доступными для обработки без дополнительного подтверждения.
- Мгновенная синхронизация всех данных. Если сотрудники одного подразделения внесли какие-либо изменения, об этом станет известно всем другим подразделениям и руководителям компании.
- Минимизация ошибок, обусловленных человеческим фактором.

Полноценный набор инструментов, связанных между собой. Например, при создании счета-фактуры, бухгалтерская документация будет сгенерирована автоматически.

ERP – системы являются мощными, гибкими и универсальными программными решениями, содержащими функционал для реализации большинства бизнес-процессов. На этапе внедрения может использоваться лишь часть доступного функционала, при этом остальные возможности будут подключены в будущем с минимальными временными и финансовыми издержками.

Таким образом, выбор в пользу ERP-системы можно рассматривать как инвестицию в будущее, которая будет раскрывать свой потенциал по мере роста и развития компании.

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 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.

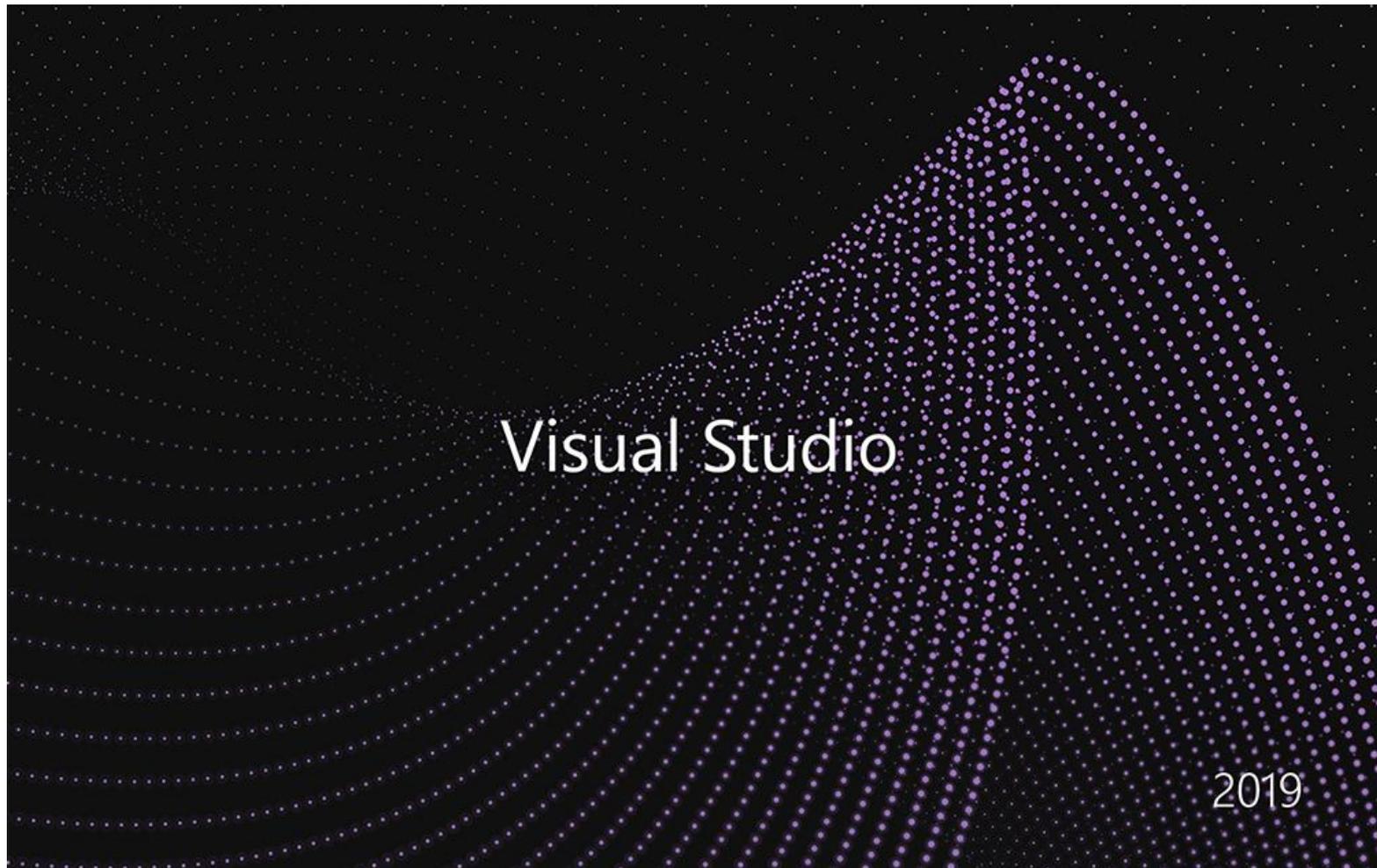
В каждом отчете должна присутствовать таблица из двух столбцов – в первом указываются составляющие администратора, во-втором составляющие пользователя.

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...
-  **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:

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. Появится:

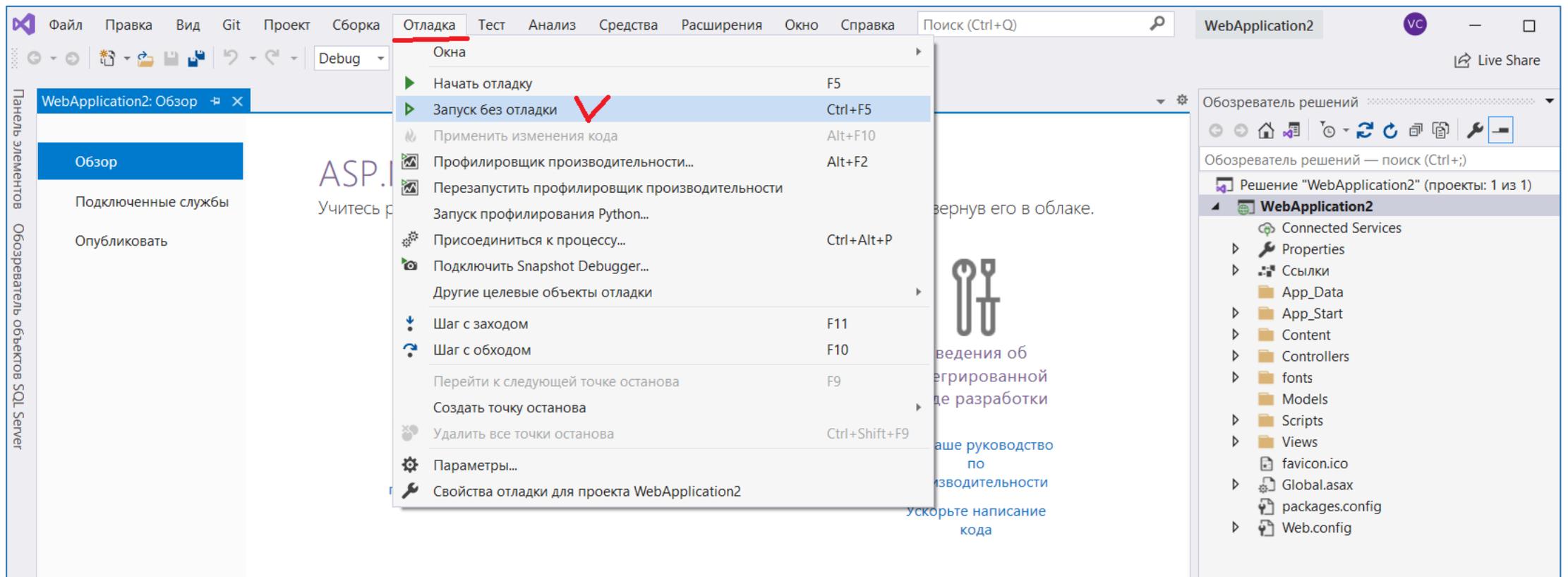
The screenshot shows the Visual Studio interface for a new ASP.NET project named 'WebApplication2'. The main content area displays the ASP.NET logo and the text: 'Учитесь работать с платформой .NET, создав свое первое приложение и развернув его в облаке.' Below this, there are three main sections:

- 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 structure, 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, 0 warnings, and 0 messages, 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!

