



The background of the slide is a collage of various data visualization elements. At the top left, there is a line graph with two series, one red and one yellow, with handwritten annotations like 'SELL!' and 'BUY!'. To its right is a bar chart with alternating blue and pink bars. Below the line graph is a world map with continents in different colors (blue for Europe/Asia, green for Africa, red for Australia). To the right of the map is a large green pie chart labeled '92%' and an orange pie chart labeled '75%'. Below these are three smaller pie charts with labels '74%', '82%', and '90%'. At the bottom, there are more bar charts and line graphs. The text 'Технология ASP.NET MVC FRAMEWORK' is centered over the collage.

Технология ASP.NET MVC FRAMEWORK

ASP.NET

ASP.NET – платформа для разработки кроссплатформенных веб-приложений, в состав которой входят:

- веб-сервисы;
- программная инфраструктура;
- модель программирования;
- и т. д.

ASP.NET входит в состав платформы .NET Framework и является развитием более старой технологии Microsoft ASP.

Платформа построена на основе общезыковой среды исполнения Common Language Runtime (CLR), где реализуется программирование по принципам ООП. Поэтому разработчики могут писать код для ASP.NET, используя ЯП, входящие в комплект .NET Framework (C#, Visual Basic.NET, J# и JScript .NET и т. д.).

Программная модель ASP.NET основывается на протоколе HTTP и использует его правила взаимодействия между сервером и браузером.

.NET Framework

.NET Framework — программная платформа, выпущенная компанией Microsoft в 2002 году. Основой платформы является общезыковая среда исполнения Common Language Runtime (CLR), которая подходит для разных ЯП.

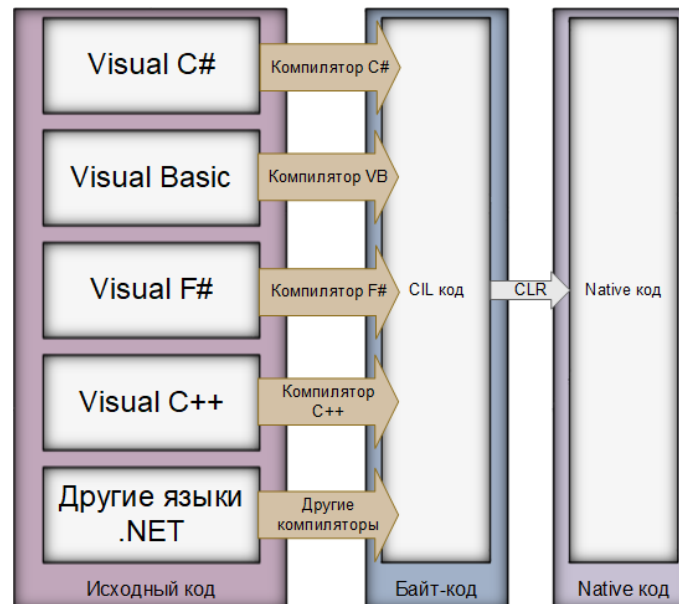
Платформа официально рассчитана на работу под операционными системы семейства Microsoft Windows, однако существуют независимые проекты (Mono и Portable.NET), позволяющие запускать программы .NET на некоторых других операционных системах.

В настоящее время .NET Framework получает развитие в виде .NET Core, изначально предполагающей кроссплатформенную разработку и эксплуатацию, но из-за большой свежести .NET Core имеет некоторые проблемы.

CLR

CLR – это исполняющая среда для байт-кода CIL, в которой компилируются программы, написанные на .NET-совместимых языках программирования (C#, Visual C++, Visual Basic .NET, F# и др.).

CLR компилирует код приложения на языке CIL (реализация компиляции которого компанией Microsoft называется MSIL) во время его исполнения, а также предоставляет MSIL-программам (программам, написанным на ЯП, поддерживающих .NET Framework) доступ к библиотеке классов .NET Framework, или так называемой .NET FCL (Framework Class Library).



ASP .NET MVC 5(6)

ASP .NET MVC 5(6) – это инфраструктура, которая предлагает высокопродуктивную модель программирования MVC (model-view-controller), которая способствует построению более чистой кодовой архитектуры, обеспечивает разработку через тестирование и поддерживает повсеместную расширяемость комбинаций со всеми преимуществами ASP.NET.

Разработка ASP .NET MVC 5 приложений, как правило может подразумевать в себе использование множества таких технологий как: JavaScript, CSS-3, HTML-5, react, angular, Entity framework для управления моделями, RAZOR – движок представления веб-страницы.

Основные преимущества при разработке на ASP NET

- Компилируемый код выполняется быстрее, большинство ошибок отлавливается ещё на стадии разработки;
- Значительно улучшенная обработка ошибок во время выполнения запущенной программы, с использованием блоков try...catch;
- ASP.NET опирается на многоязыковые возможности .NET, что позволяет писать код страниц на VB.NET, Delphi.NET, Visual C#, J# и т. д.;
- Возможность кэширования данных, используемых на странице;
- Возможность разделения визуальной части и бизнес-логики по разным файлам («code behind»);
- Расширяемая модель обработки запросов;
- Расширяемая модель серверных элементов управления;
- Наличие master-страниц для задания шаблонов оформления страниц;
- Встроенная поддержка AJAX;

Основные преимущества при разработке на ASP.NET

- Взаимодействие пользователя с приложением MVC осуществляется в соответствии с естественным циклом: пользователь предпринимает действие, в ответ на которое приложение изменяет свою модель данных и доставляет обновленное представление пользователю. Затем цикл повторяется. Это хорошо укладывается в схему веб-приложений, предоставляемых в виде последовательностей запросов и ответов HTTP.
- Инфраструктура MVC Framework построена в виде набора независимых компонентов, которые удовлетворяют интерфейсу .NET или созданы на основе абстрактного базового класса. Компоненты, подобные системе маршрутизации, механизму визуализации и фабрике контроллеров, можно легко заменять другими компонентами с собственной реализацией.
- Использование C#.
- Инфраструктура ASP.NET MVC генерирует ясный и соответствующий стандартам код разметки. Ее встроенные вспомогательные методы HTML производят соответствующий стандартам вывод

Entity Framework 6

Entity Framework представляет специальную объектно-ориентированную технологию на базе фреймворка .NET для работы с данными, т. е. если на физическом уровне мы оперируем с таблицами, индексами, первичными и внешними ключами, то на концептуальном уровне, который нам предлагает Entity Framework, мы уже работаем с объектами (которые описывают модели).

Центральной концепцией Entity Framework является понятие сущности или entity. Сущность представляет набор данных, ассоциированных с определенным объектом, при этом сущности могут быть связаны ассоциативной связью один-ко-многим, один-ко-одному и многие-ко-многим, подобно тому, как в реальной базе данных происходит связь через внешние ключи.

Entity Framework 6

Другим ключевым понятием является Entity Data Model. Эта модель сопоставляет классы сущностей с реальными таблицами в БД. Entity Data Model состоит из трех уровней:

1. Концептуальный. Здесь происходит определение классов сущностей, используемых в приложении.
2. Уровень хранилища. Он определяет таблицы, столбцы, отношения между таблицами и типы данных, с которыми сопоставляется используемая база данных.
3. Уровень сопоставления (маппинга). Он служит посредником между предыдущими двумя, определяя сопоставление между свойствами класса сущности и столбцами таблиц.

Способы взаимодействия с БД

Entity Framework предполагает три возможных способа взаимодействия с базой данных:

1. Database first: Entity Framework создает набор классов, которые отражают модель конкретной базы данных;
2. Model first: сначала разработчик создаёт модель базы данных, по которой затем Entity Framework создаёт реальную базу данных на сервере;
3. Code first: разработчик создаёт класс модели данных, которые будут храниться в БД, а затем Entity Framework по этой модели генерирует базу данных и её таблицы.

TRANSACT SQL

Transact-SQL (T-SQL) — процедурное расширение языка SQL, созданное компанией Microsoft (для Microsoft SQL Server).

Язык Transact-SQL является ключом к использованию MS SQL Server. Все приложения, взаимодействующие с экземпляром MS SQL Server, независимо от их реализации и пользовательского интерфейса, отправляют серверу инструкции Transact-SQL

Преимущества T-SQL

1. Управляющие операторы,
2. Локальные и глобальные переменные,
3. Различные дополнительные функции для обработки строк, дат, математики и т. п.,
4. Поддержка аутентификации Microsoft Windows.

Razor

Razor - это интеллектуальный обработчик программного кода динамических веб-страниц *ASP.NET*. Имеет простой, интуитивно понятный, синтаксис встраивания программного кода в веб-страницы. **Razor** - механизм визуализации (view engine) поддерживаемый *.NET Framework*, *.NET Core* в рамках *ASP.NET* и специально предназначен для создания веб-приложений.

Движок Razor даёт всю мощь *ASP.NET*, но использует упрощённый синтаксис, лёгкий для новичков и повышающий производительность для профессионалов. Razor позволяет писать программный код без множества открывающих и закрывающих меток по всему шаблону делая процесс разработки быстрым. Дизайн движка Razor использует минимальное количество символов для обозначения границ программного кода, не требуя явно обозначать серверные блоки в HTML-коде. Для более быстрой разработки Razor поддерживает так называемые «хелперы», которые позволяют в одну строку на основе входных данных генерировать большой массив HTML-кода, т. е. по сути являются готовыми шаблонами.

Razor

Пример RAZOR кода:

```
<div>
  <table>
    @for (int row = 1; row <= 4; row++)
    {
      <tr>
        @for(int column = 1; column <= 10; column++)
        {
          <td>@row x @column</td>
        }
      </tr>
    }
  </table>
</div>
```

Razor

Пример «хелпера»:

```
1 @helper Createlist(string[] all)
2 {
3     <ul>
4         @foreach (string s in all)
5         {
6             <li>@s</li>
7         }
8     </ul>
9 }
10 @{
11     string[] cities = new string[] { "Лондон", "Париж", "Москва" };
12 }
13 @{
14     string[] countries = new string[] { "Великобритания", "Франция", "Россия" };
15 }
16 <h3>Города</h3>
17 @Createlist(cities)
18 <br />
19 <h3>Страны</h3>
20 @Createlist(countries)
```