

Semana 1

Curso:
Desarrollo de aplicaciones web

Ciclo: 2023-1

Unidad I: Introducción a las aplicaciones web



UNIVERSIDAD
esan

UNIDAD DE APRENDIZAJE I: INTRODUCCIÓN A LAS APLICACIONES WEB

RESULTADOS DE APRENDIZAJE:

- Comprende el ecosistema .NET.
- Comprende sobre la arquitectura de aplicaciones web estáticas y dinámicas.
- Entiende la diferencia sobre .NET Framework y .NET Core.
- Comprende el patrón de diseño MVC.
- Entiende la importancia de la programación orientada a objetos y el principio SOLID para la construcción de aplicaciones web empresariales.
- Comprende la comunicación de información entre los diversos componentes web.

CONTENIDO:

- 1.1. Introducción a .NET.
- 1.2. Introducción al desarrollo de aplicaciones web.
- 1.3. Páginas estáticas y dinámicas.
- 1.4. Fundamentos de HTML y CSS.
- 1.5. ASP.NET Framework.
- 1.6. ASP.NET Core.

Desarrollo de aplicaciones web – Semana 1

INTRODUCCIÓN A .NET



Introducción a .NET

- Acontecimientos
 - Anders Hejlsberg: Padre del C# (antes J++) y colaborador con Turbo Pascal, Delphi.
 - Necesidad de mejorar Visual Studio 6 (1998).
 - Respuesta de Microsoft sobre Sun Microsystems.
 - Lanzamiento Febrero 2002: Visual Studio .NET.
 - .NET Framework 1.0



Anders Hejlsberg



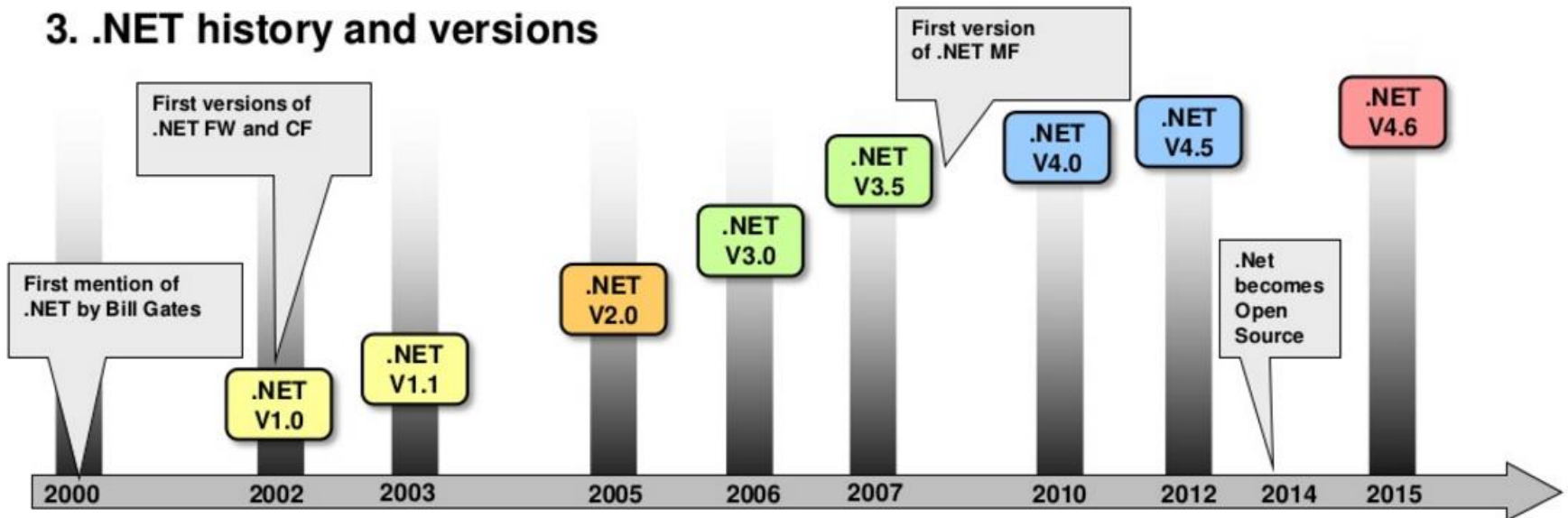
.NET Mitos y Verdades

- ¿Qué **NO** es .NET?
 - Un sistema operativo.
 - Un lenguaje de programación.
 - Un entorno de desarrollo(IDE's)
- ¿Qué **SI** es .NET?
 - Un entorno de ejecución administrado.
 - Un conjunto de lenguajes de programación.
 - Un conjunto de bibliotecas y clases reutilizables.

“.NET es un estándar de ECMA que tiene distintas implementaciones: .NET Framework, Mono, Unity y, ahora, .NET Core.”

.NET Framework

3. .NET history and versions



.NET Framework:

Standard framework for desktop and server hosts.

.NET Compact Framework:

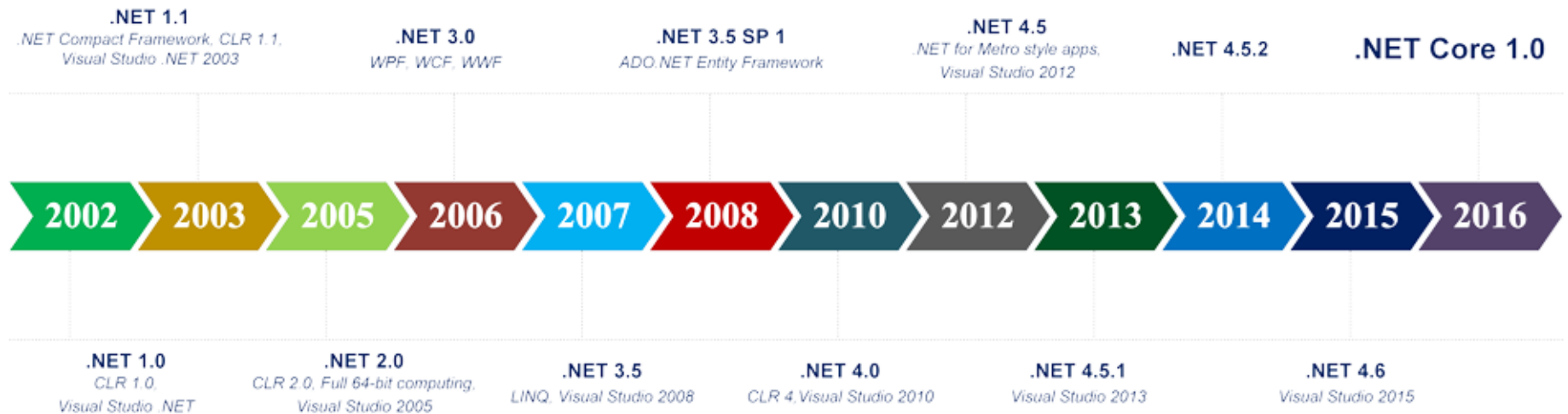
Stripped-down .NET FW for embedded devices, based on Windows CE / Windows Embedded Compact.

.NET Micro Framework:

Variant for „deeply“ embedded devices (no OS required, .NET is the OS).

Timeline .NET

The release history of .NET Framework



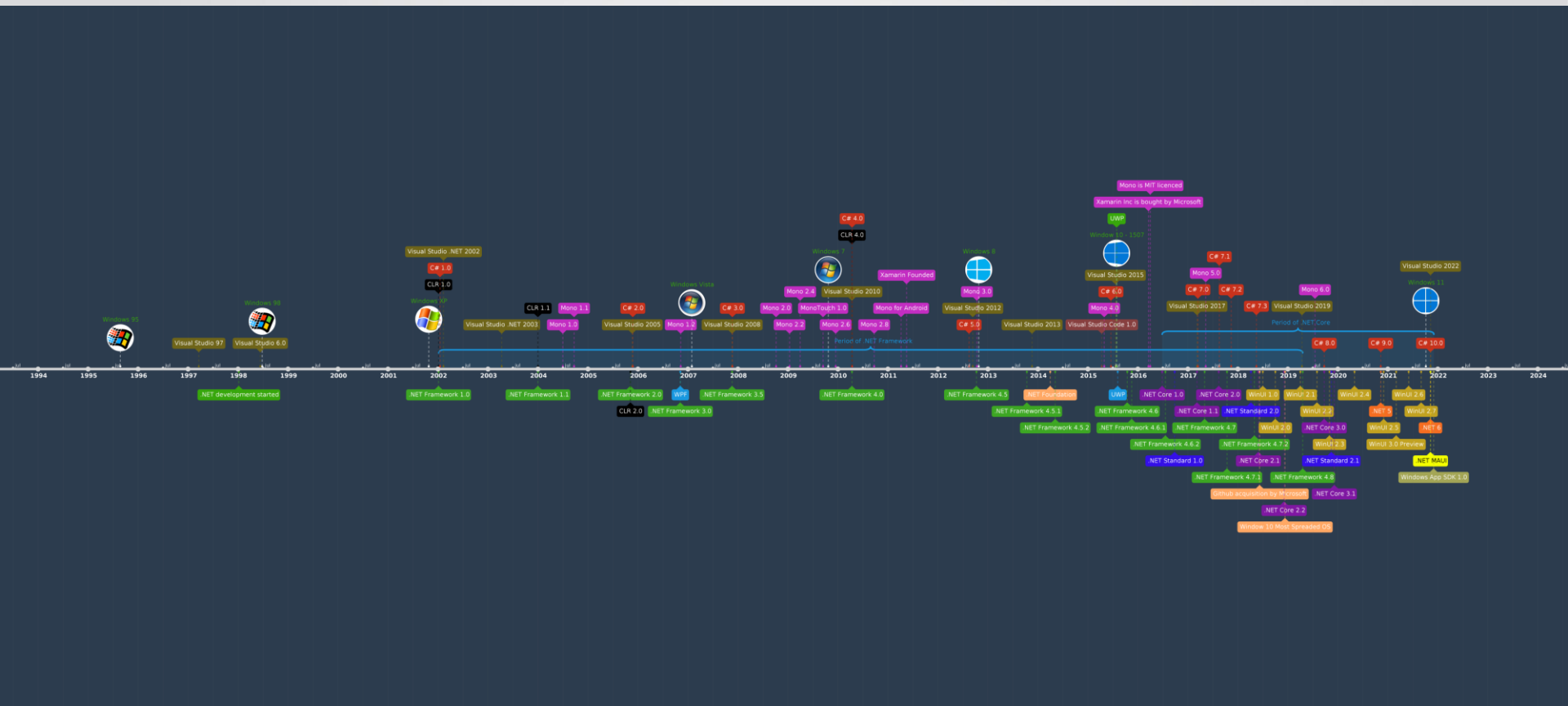
.NET
Core
.NET 6

2022

.NET
Framework
.NET Framework 4.8

Supported versions

Version	Status	Latest release
.NET 6.0	Preview	6.0.0-preview.2
.NET 5.0 (recommended)	Current	5.0.4
.NET Core 3.1	LTS	3.1.13
.NET Core 2.1	LTS	2.1.26



Download
.NET 7

C#11
.NET 7

The diagram illustrates the .NET ecosystem across different platforms and tools. At the top, icons represent various platforms: CLOUD, WEB, DESKTOP, MOBILE, GAMING, IoT, and AI. In the center, a large blue box labeled ".NET" is connected to these platform icons. Below the ".NET" box, there are three main categories: TOOLS, OPERATING SYSTEM, and ECOSYSTEM. The TOOLS category includes Visual Studio Family, Visual Studio Code, and CLI. The OPERATING SYSTEM category includes Windows, Linux, and macOS. The ECOSYSTEM category includes NuGet, GitHub, and Components, tools, library vendors.

.NET Framework & .NET Core

- .NET Framework: Es un framework para desarrollar aplicaciones para Windows (desktop, mobile, services), ***Nivel del madurez: Alta.***
- .NET Core: Entorno de ejecución Cross-Platform. ***Nivel del madurez: media-alta.***

¿Cuándo utilizar .NET Core?

- .NET Framework: Es un framework para desarrollar aplicaciones para Windows (desktop, mobile, services), **Nivel de madurez: Alta.**
- .NET Core: Entorno de ejecución Cross-Platform. Nivel de madurez: **Media-Alta.**

¿Cuándo utilizar .NET Core?

Recomendable Utilizar .NET CORE cuando:

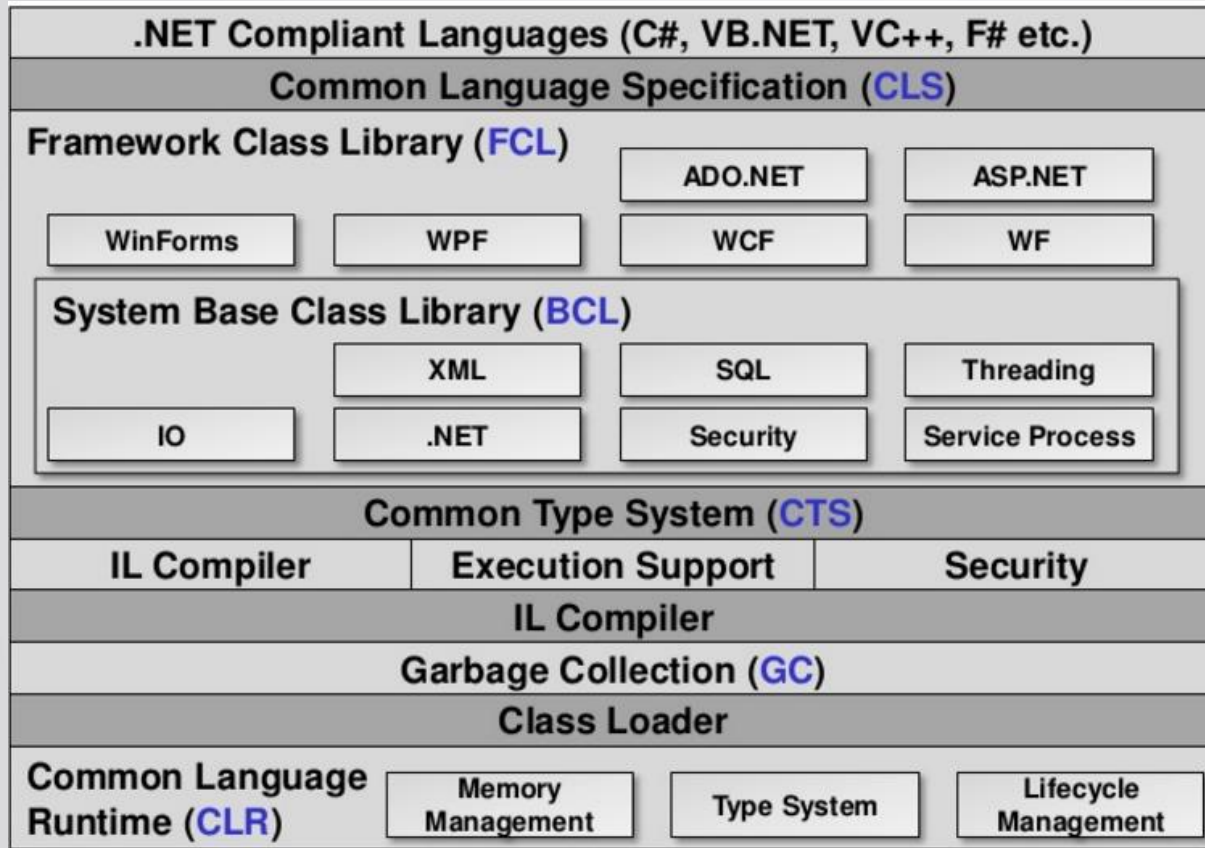
- Nuevos proyectos.
- Cross-Platform.
- Arquitectura basada en microservicios.
- Escalable y con altas exigencias de rendimiento.

NO recomendable Utilizar .NET CORE cuando

- Migración de aplicaciones actuales.
- Dependencia de Entity Framework, Windows Communication Foundation, Workflow Foundation.

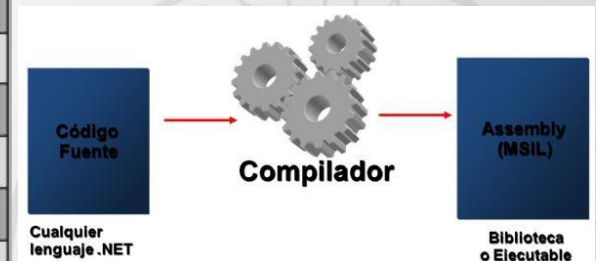
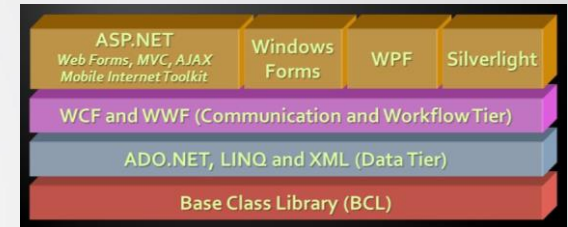


.NET Framework Overview



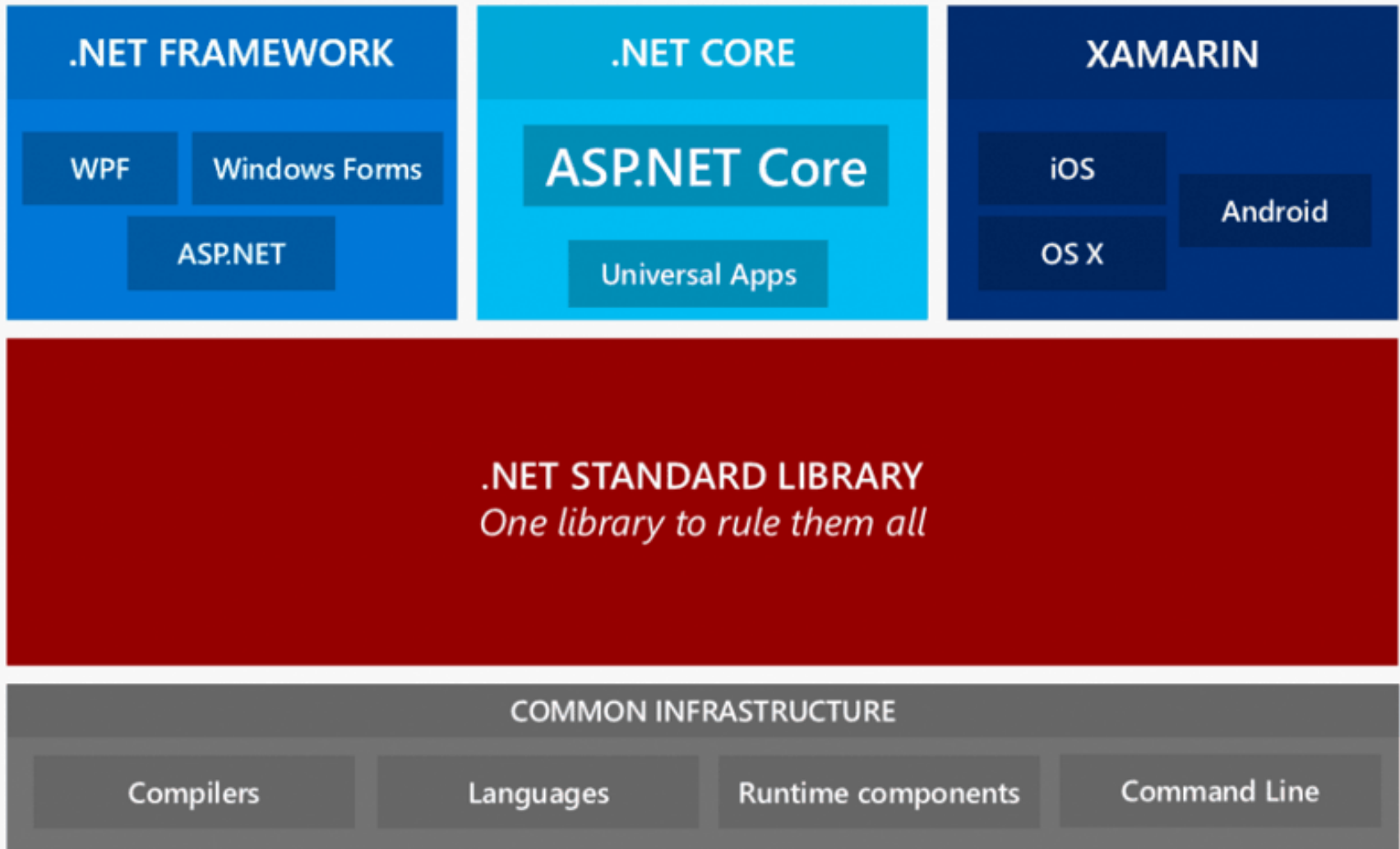
Fuente: Peter Egli 2015

FCL
(Framework Class Library)

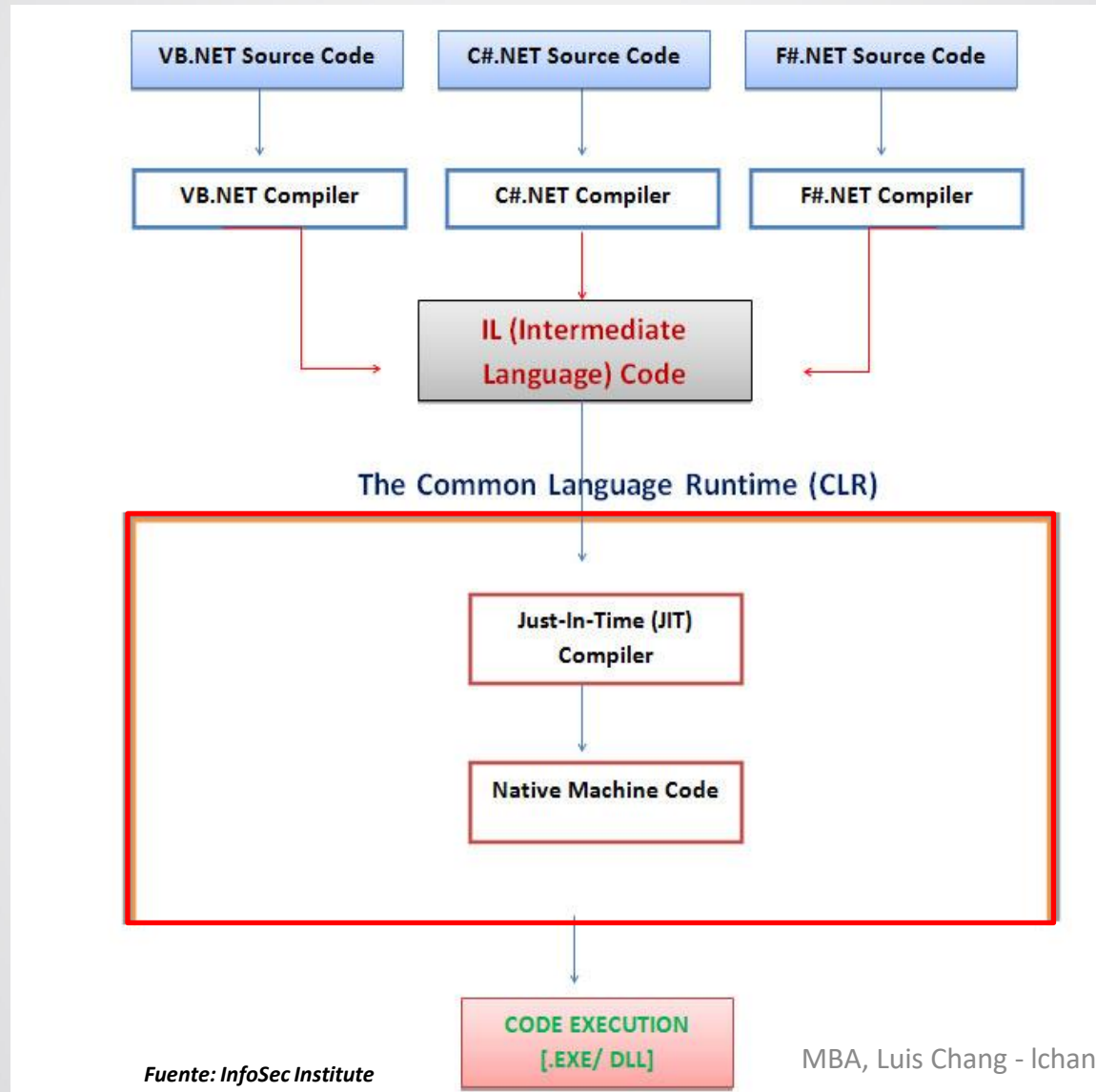


CLR
(Common Language Runtime)

.NET Framework Overview



Common Language Runtime (CLR)

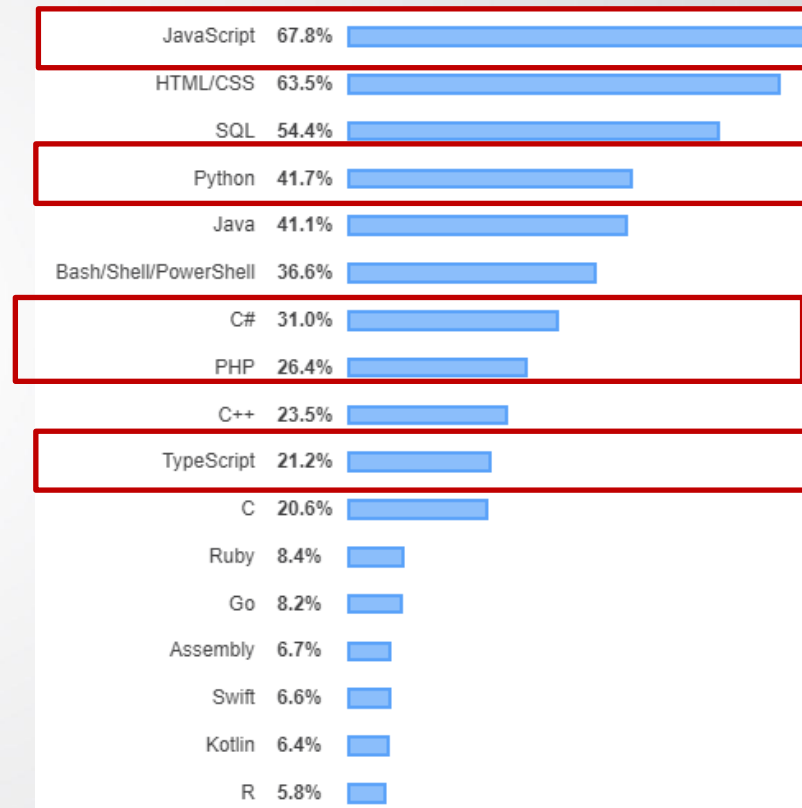
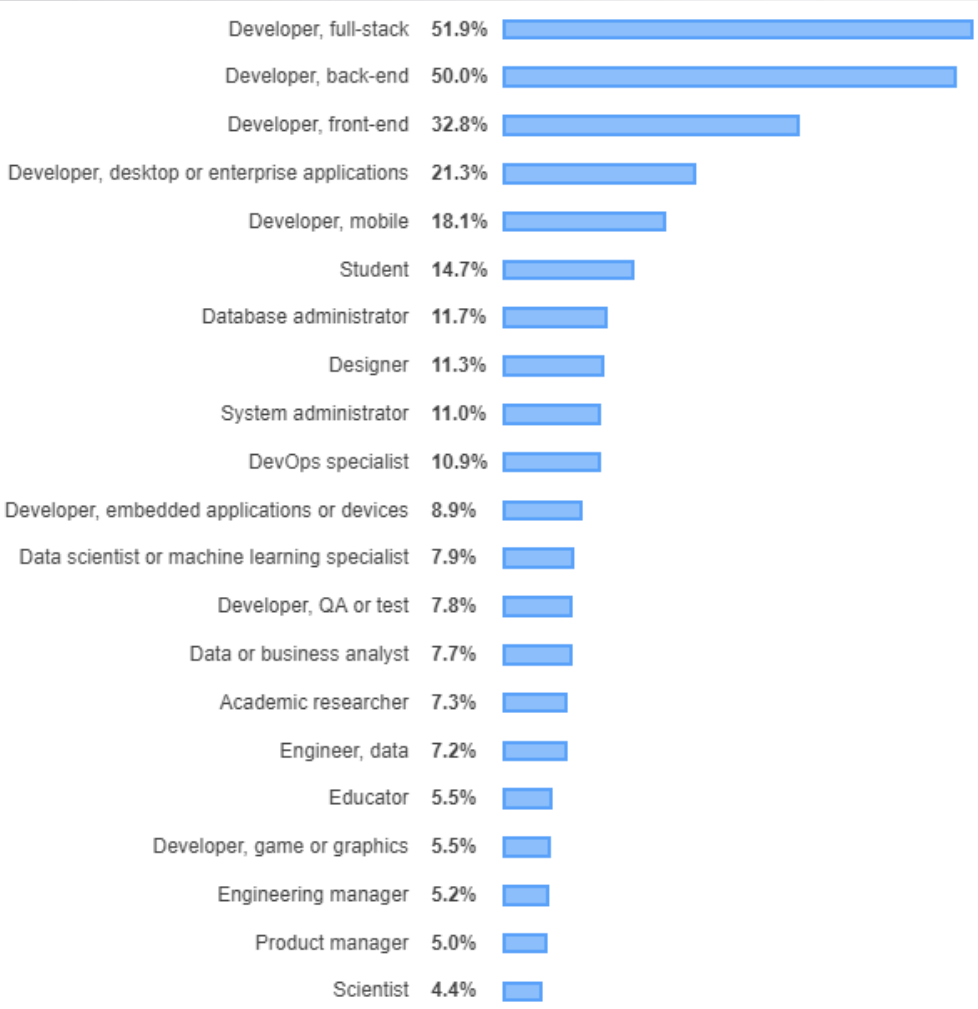


Desarrollo de aplicaciones web – Semana 1

INTRODUCCIÓN AL DESARROLLO DE APLICACIONES WEB



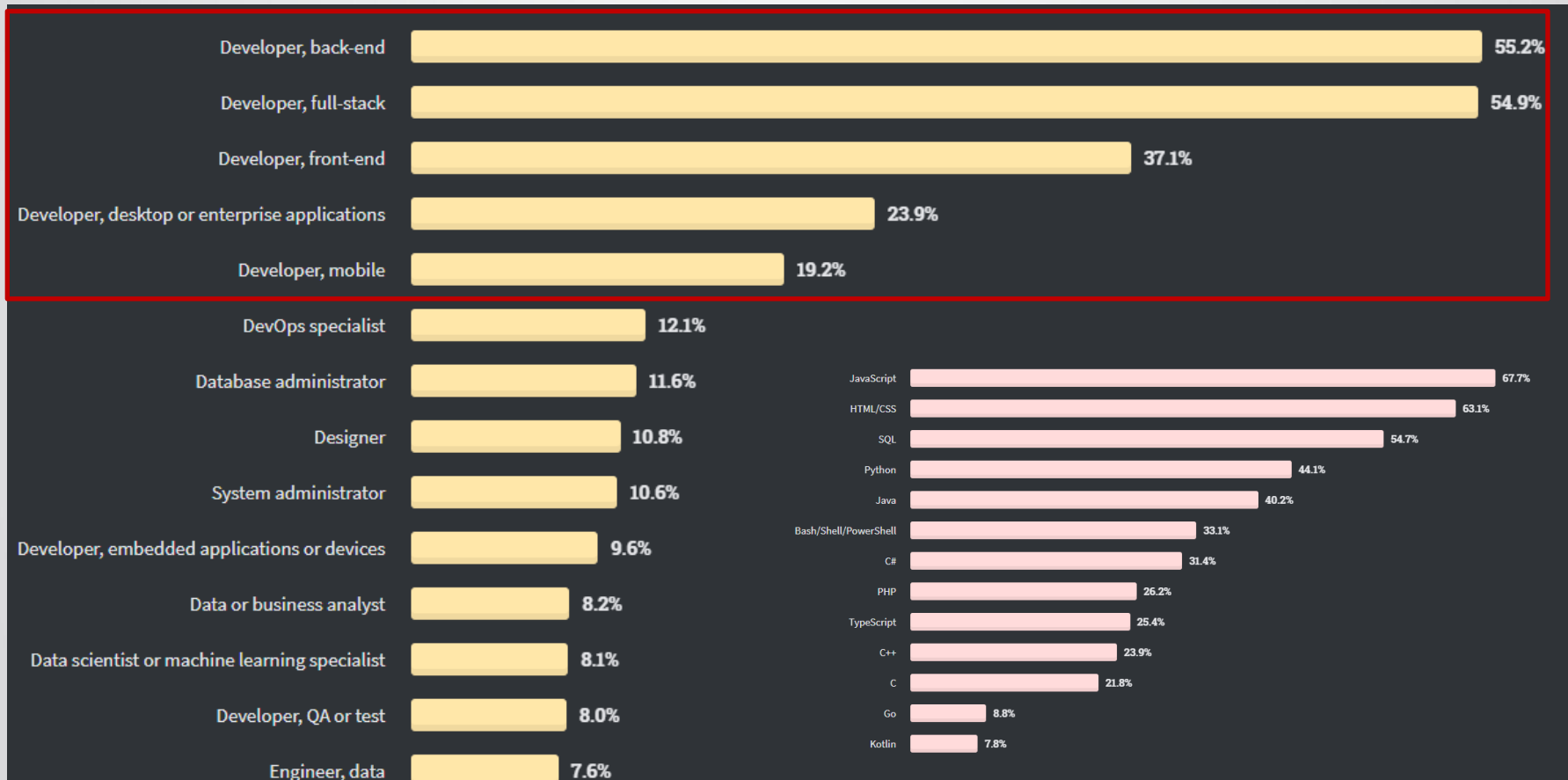
Tendencias sobre el desarrollo de software



Fuente: [Stack Overflow Developer Survey Results 2019](#)

90,000 responses;

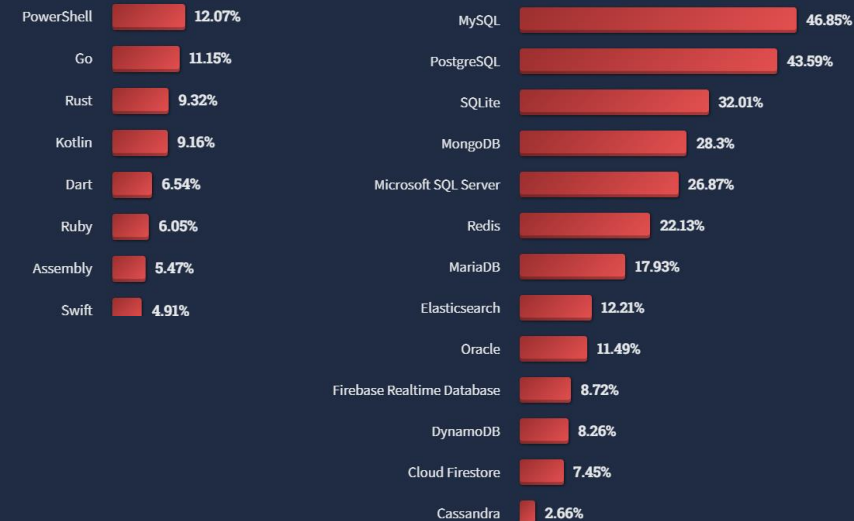
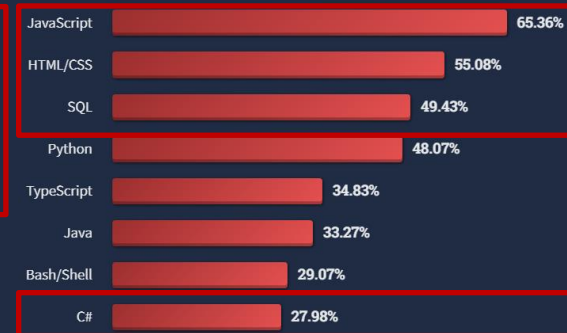
Tendencias sobre el desarrollo de software



Fuente: [Stack Overflow Developer Survey Results 2020](#)

90,000 responses;

Tendencias sobre el desarrollo de software

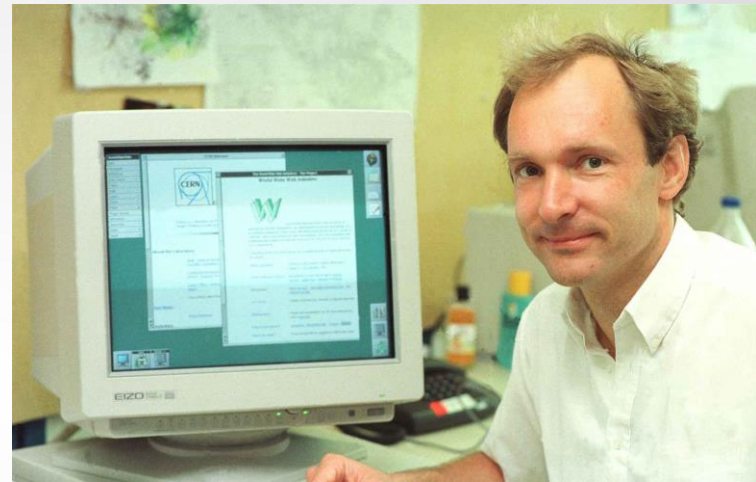


Fuente: [Stack Overflow Developer Survey Results 2022](#)

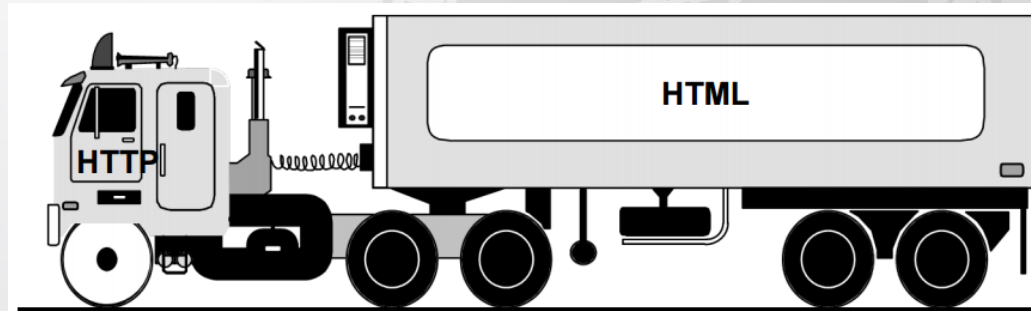
61,302 responses;

World Wide Web - History

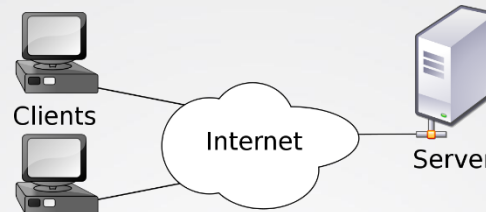
- [Creada en 1989 por Tim Berners.](#)
- Basado en hipertextos.
- Hyper Text Markup Language.
- Hyper Text Transfer Protocol.
- Uniform Resource Locator/ Identifier



Fuente: Web Foundation Org.



Web Page – Request/Response



```
<html>
<head>
  <meta content="text/html;
  charset=UTF-8" http-equiv="Content-Type"/>
  <title>My lovely web page</title>
</head>
<body>
  <h1>This is my lovely web page</h1>
  <p>
    It has lots of lovely content. It has some
    <em>emphasized text</em>
    and look at this, a blockquote:
  </p>
  <blockquote>
    <p>You fools, I will destroy you
    all!</p>
  </blockquote>
  <h2>And here's a subheading</h2>
  <p>That about covers it, I think</p>
  <hr/>
</body>
</html>
```

This is my lovely web page

It has lots of lovely content. It has some *emphasized text* and look at this, a blockquote:

You fools, I will destroy you all!

And here's a subheading

That about covers it, I think

Desarrollo de aplicaciones web – Semana 1

PÁGINAS WEB ESTÁTICAS Y DINÁMICAS



Estándares Web

- Componentes básicos: **HTML**, **CSS** y **JavaScript**.
- **HTML**: Hyper Text Markup Language, es un lenguaje de **ETIQUETAS** utilizado en la capa de presentación de las páginas web.
- **CSS**: Cascading Style Sheets, es un lenguaje para definir el “estilo” de la capa de presentación. Describe como debe ser renderizado el elemento estructurado en pantalla.
- **JavaScript**: Es un lenguaje de programación multiplataforma y orientado a objetos, no requiere compilación, funciona del lado del cliente y JS le permite agregar interactividad y características dinámicas a su sitio web, pero tiene limitaciones, lo que nos lleva a los lenguajes de programación del servidor y a las páginas web dinámicas.



Páginas web estáticas

- Diseñadas para mostrar información permanente.
- De uso informativo.
- No se utilizan Base de Datos.
- Interactúan por medio hipervínculos o enlaces.
- Ventajas económicas.
- Fácil implementación.
- Edición manual.
- Enfocados a proyectos pequeños.



Páginas web dinámicas



Desarrollo de aplicaciones web – Semana 2

FUNDAMENTOS DE HTML Y CSS



Recursos necesarios



https://www.w3schools.com/html/html_exercises.asp

<https://learngitbranching.js.org/>

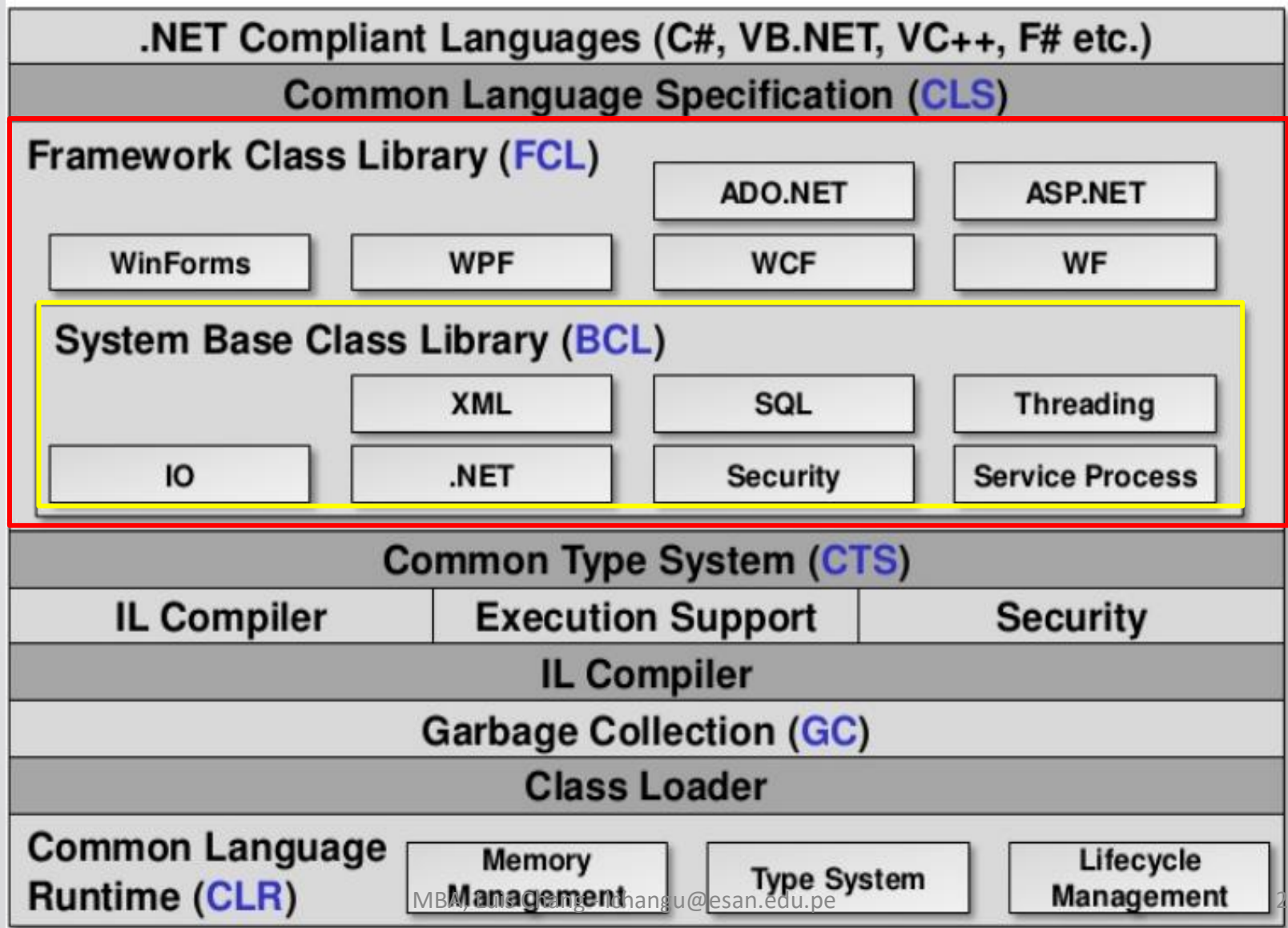
https://sqlzoo.net/wiki/SQL_Tutorial

Desarrollo de aplicaciones web – Semana 1

ASP.NET FRAMEWORK



.NET FRAMEWORK



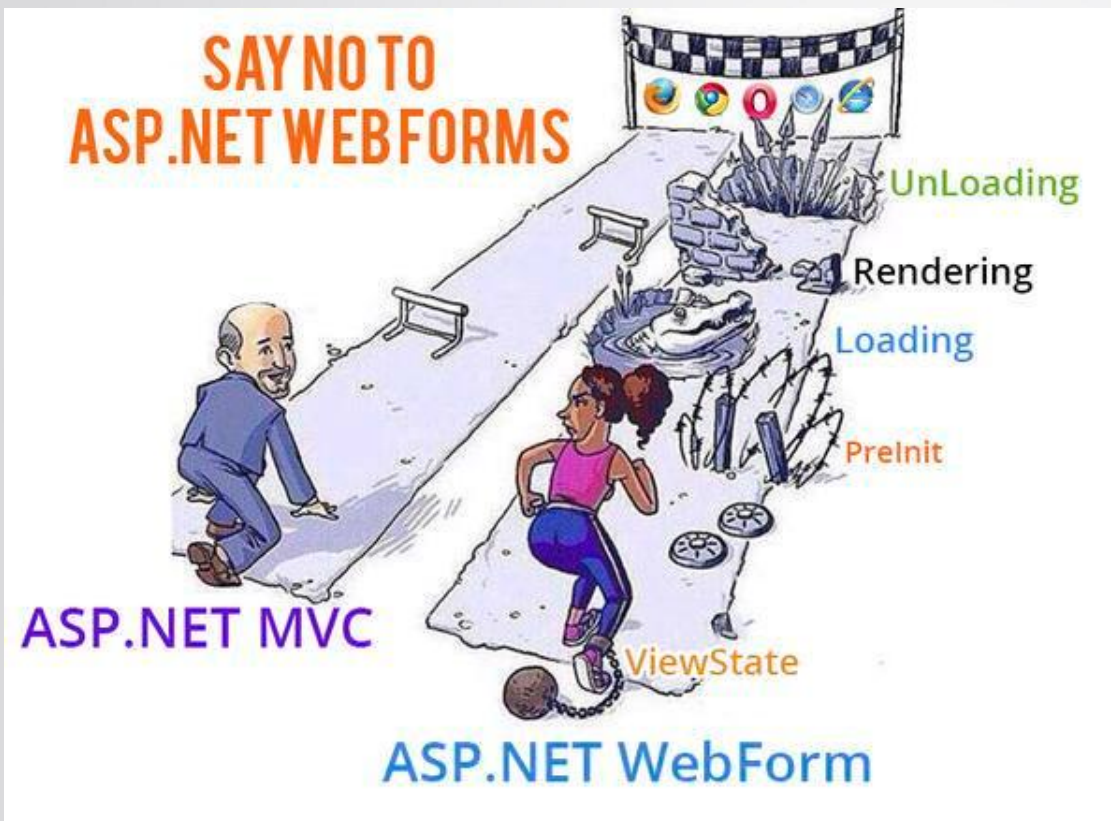
Active Server Page .NET

“Es un modelo de desarrollo Web unificado que incluye los servicios necesarios para crear aplicaciones Web empresariales con el código mínimo. El código de las aplicaciones puede escribirse en cualquier lenguaje compatible con el Common Language Runtime (CLR), entre ellos Microsoft Visual Basic, C#, JScript .NET y J#”.

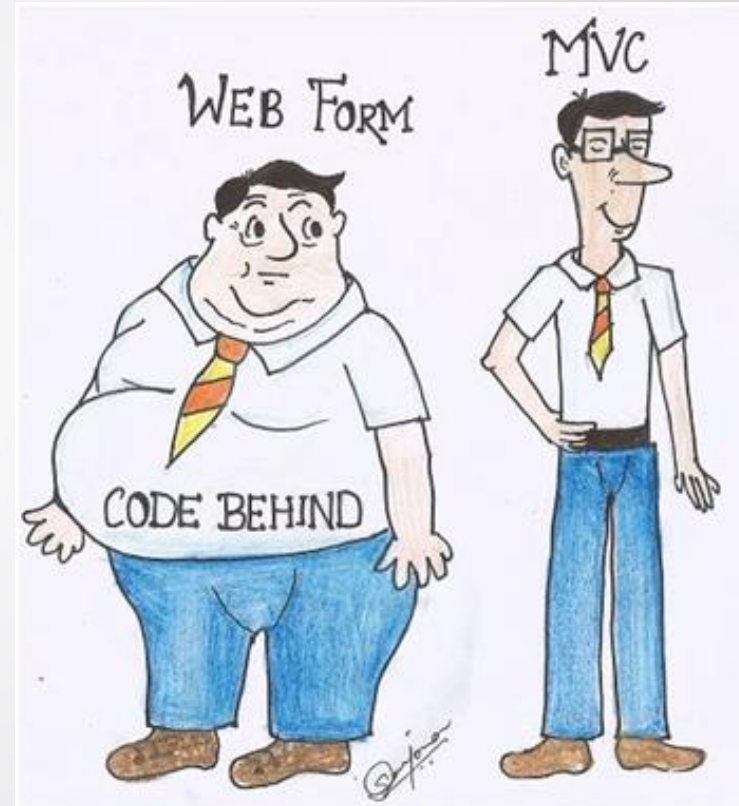
Fuente: MSDN – ASP.NET



ASP.NET: Web Forms & MVC



Fuente: Kemal Durand



Fuente: CodeProject: Web Forms vs MVC

Desarrollo de aplicaciones web – Semana 1

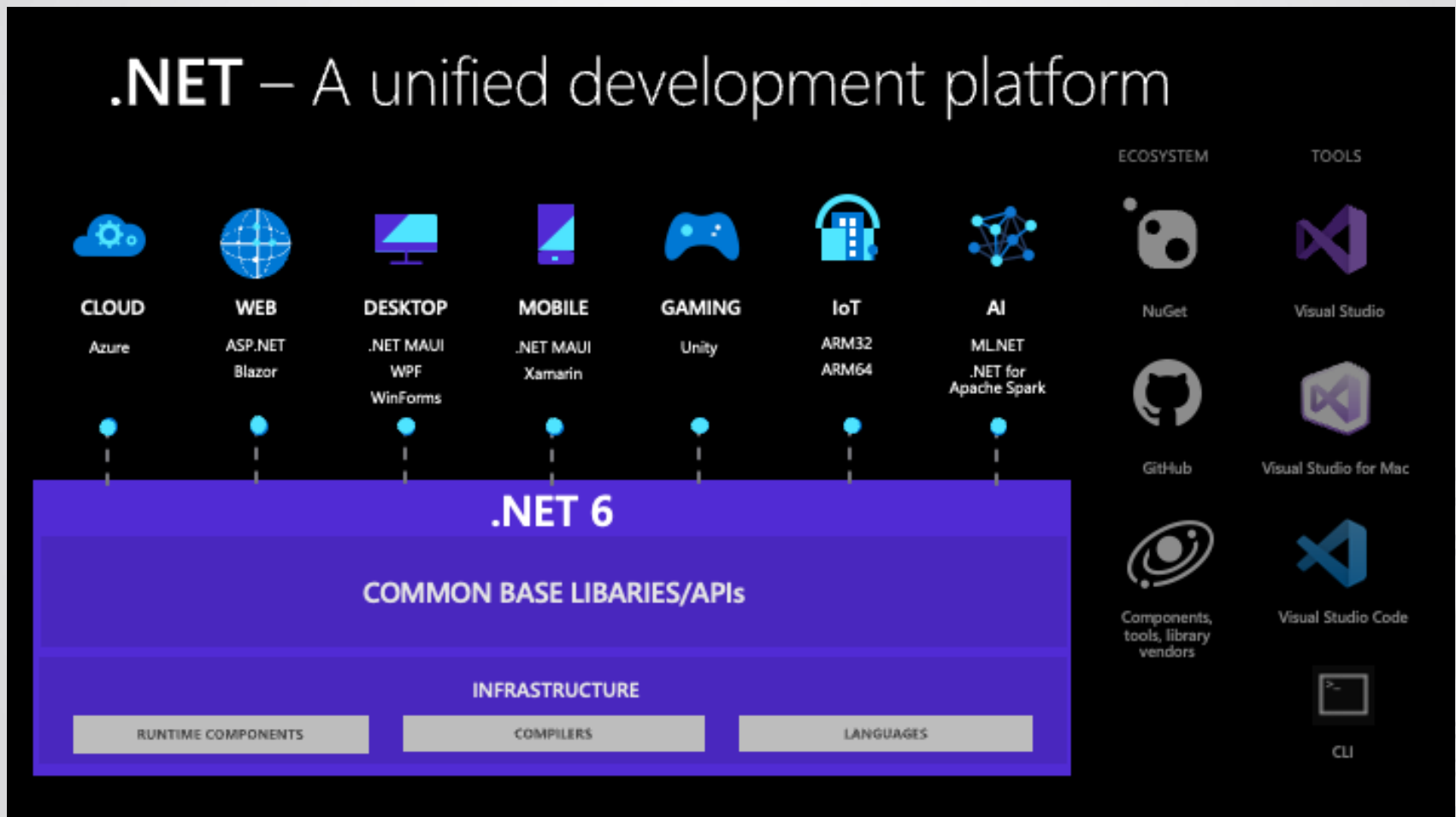
ASP.NET CORE



Overview .NET Core



Overview .NET Core



Referencias

- Price, M. (2020). C# 9 and .NET 5 - Modern Cross-Platform Development. Birmingham: Packt Publishing Ltd. Chapter 1 Pages 8 – 17
- Frain, B. (2020). Responsive Web Design with HTML5 and CSS. Birmingham: Packt Publishing Ltd. Chapter 2 Pages 25 - 53

