

**PREGRADO**



UNIDAD 1 | OVERVIEW

# **WEB APPS FOUNDATION**

Al finalizar la unidad de aprendizaje, el estudiante describe el proceso de software realizado aplicando el paradigma orientado a objetos combinado con aspectos de la programación funcional, utilizando el lenguaje C# y frameworks de actualidad, para desarrollar aplicaciones web básicas en un ambiente de desarrollo colaborativo.

---

# AGENDA

WEB APPLICATIONS

.NET FRAMEWORK

C# LANGUAGE



## **Application Software**

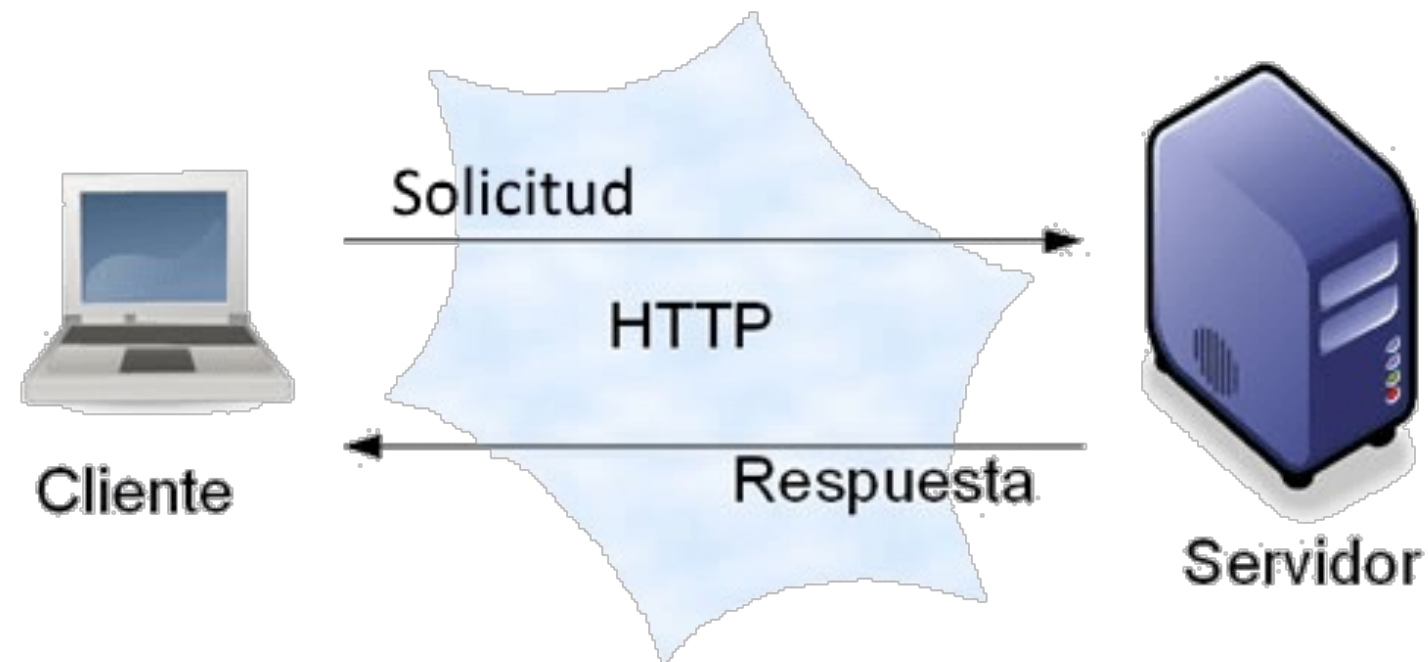
Denominada también **app**, es un programa o grupo de programas orientados a usuarios finales.

Este tipo de software es distinto de **system software**, que se relaciona con la ejecución del computador

## Web Application

En ingeniería de software se denomina aplicación web a un programa que se ejecuta en un web server.

Este tipo de aplicaciones se acceden a través de un web browser y operan bajo un enfoque cliente/servidor.



## **Web page**

**Web page** es un documento escrito en HTML (Hypertext Markup Language), el cual es traducido por el Web browser.

Las Web pages pueden ser estáticas (static pages) o dinámicas (dynamic pages).

**Web page** es distinto a **Web site** (o website). **Web site** es una **colección** de pages, mientras que **Web page** es un documento HTML **individual**.

# Ventajas

1

- No necesita ningún tipo de instalación (a nivel de usuario) → navegador.

2

- Es multiplataforma y multidispositivo.

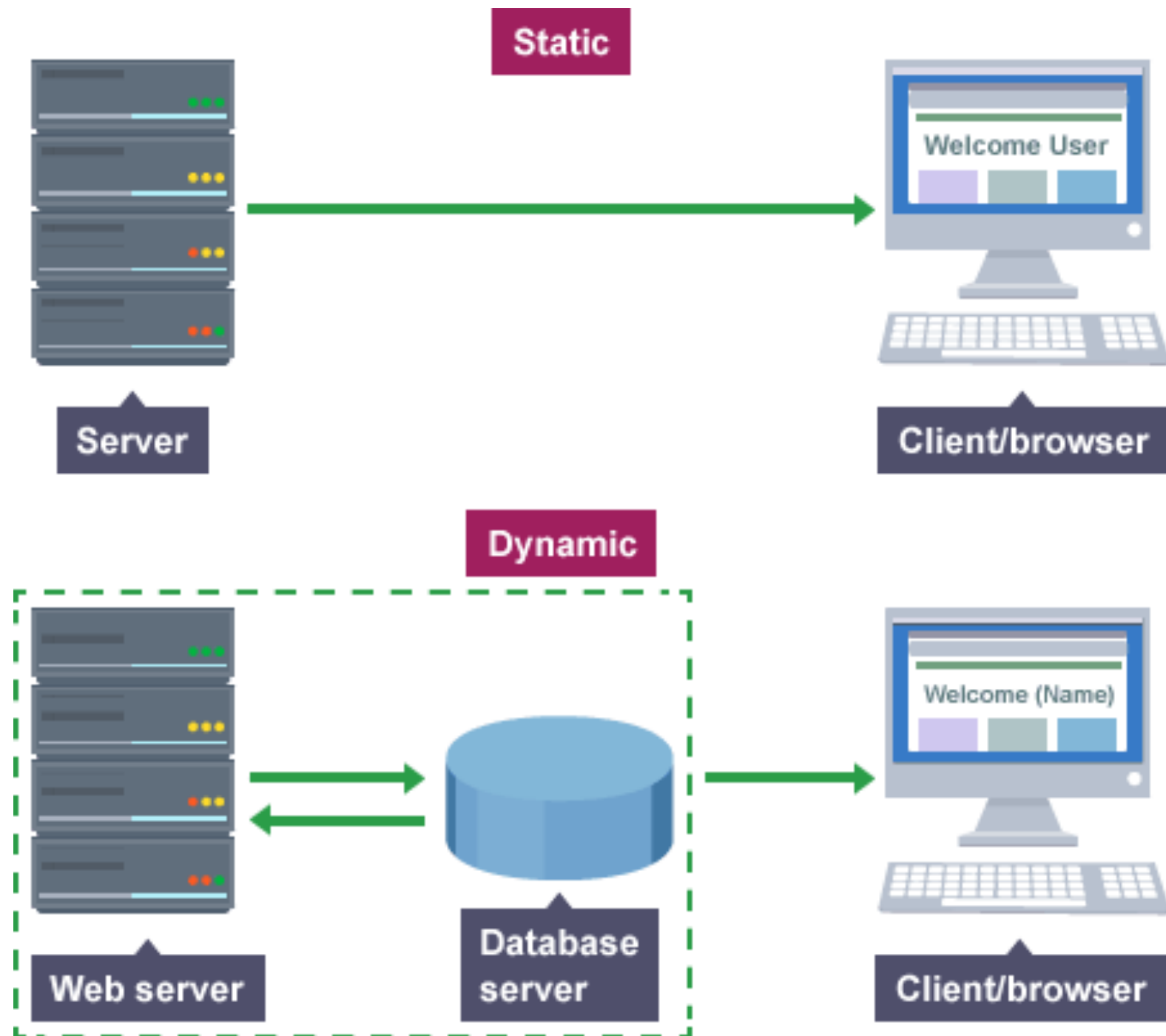
3

- El peso de la aplicación la soporta el servidor.

4

- Puede estar en intranet y en la nube.

# ¿Cómo funciona?





## Web service

Es una aplicación o fuente de datos que es accesible vía un protocolo web estándar (HTTP o HTTPS).

A diferencia de las *web applications*, los *web services* se diseñan para comunicarse con otros programas, en vez de con los usuarios directamente.

Los web services pueden proporcionar los datos en diversos formatos, siendo XML y JSON de los más conocidos.

Muchos web services ofrecen un API (Application Programming Interface) a modo de conjunto de funciones y comandos, que se puede utilizar para acceder a los datos.

Un API es un conjunto específico de comandos o guidelines que se utilizan para acceder a los datos, mientras que un web service es un servicio real, que ofrece una fuente basada en internet.

# Retos

- Las aplicaciones web modernas se enfrentan a mayores expectativas del usuario y más demanda que nunca. Se espera que las aplicaciones web de hoy estén disponibles 24/7, desde cualquier parte del mundo, y se puedan usar desde prácticamente cualquier dispositivo.
- Las aplicaciones web deben ser seguras, flexibles y escalables para satisfacer los niveles variables de demanda.
- Las aplicaciones web de hoy en día demandan experiencias de usuario enriquecidas y comunicándose eficientemente a través web services.

---

# AGENDA

WEB APPLICATIONS

.NET FRAMEWORK

C# LANGUAGE



# .NET

.NET Es una plataforma de desarrollo open-source, creada por Microsoft y que permite la construcción de diversos tipos de aplicaciones.

Las aplicaciones pueden ir dirigidas a la web, mobile, desktop, games y IoT (Internet-of-Things).



## **.NET**

- Su diseño modular permite que las aplicaciones dependan sólo de las características que realmente usan, lo que mejora la seguridad y el rendimiento de las aplicaciones y reduce los requisitos de recursos de alojamiento.

# Lenguajes

- C#, lenguaje de programación orientado a objetos, cross-platform, typesafe.
- F# lenguaje de programación funcional, cross-platform, open-source, con soporte para programación orientada a objetos e imperativa.
- Visual Basic, lenguaje de programación orientado a objetos, typesafe.

# Cross-platform

Implementaciones de .NET

**.NET Core.** Implementación cross-platform para websites, servidores y aplicaciones de consola, ejecutable en macOS, Windows y Linux.

**.NET Framework.** Implementación que soporta websites, desktop apps entre otros, sobre Windows.

**Xamarin/Mono.** Implementación para ejecutar apps en los principales sistemas operativos para móviles.

## **.NET Standard**

Es la especificación formal de los APIs los cuales son comunes a todas las implementaciones de .NET.

.NET Standard tiene versiones. Las bibliotecas construidas con una versión de .NET Standard, puede ejecutarse en cualquier implementación de .NET que implemente dicha versión de .NET Standard, o superior.



# **Libraries**

Es la especificación formal de los APIs los cuales son comunes a todas las implementaciones de .NET.

.NET Standard tiene versiones. Las bibliotecas construidas con una versión de .NET Standard, puede ejecutarse en cualquier implementación de .NET que implemente dicha versión de .NET Standard, o superior.

## **.NET Core**

- .NET Core es la versión lanzada en junio del 2016, que cambió el panorama con respecto al desarrollo de las aplicaciones .NET ya que, esta fue diseñado bajo el concepto de que sea open-source y cross-platform.
- Podemos tener aplicaciones .NET en Windows, Linux, Mac y/o aportar a la contribución del código mediante su repositorio en GitHub.
- A partir de versión 5, se elimina Core del nombre.



# Release History

Version	Release Date	Released with
.NET Core 1.0	27-06-2016	Visual Studio 2015 Update 3
.NET Core 1.1	16-11-2016	Visual Studio 2017 Version 15.0
.NET Core 2.0	14-08-2017	Visual Studio 2017 Version 15.3
.NET Core 2.1	30-05-2018	Visual Studio 2017 Version 15.7
.NET Core 2.2	04-12-2018	Visual Studio 2017 Version 15.9
.NET Core 3.0	23-09-2019	Visual Studio 2019 Version 16.3
.NET Core 3.1	03-12-2019	Visual Studio 2019 Version 16.4
.NET 5	10-11-2019	Visual Studio 2019 Version 16.9
.NET 6	08-11-2021	Visual Studio 2022 Version 17.3

## **C# language**

C# (pronounced "See Sharp") es un lenguaje de programación moderno, type-safe.

C# permite que los developers construyan muchos tipos de aplicaciones seguras y robustas que ejecutan en el .NET ecosystem. Incluye características como Garbage collection, Exception Handling, Lambda expressions, asynchronous operations.

# C# language

## Sintaxis

```
using System;
```

```
namespace HelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
        }
    }
}
```

<https://docs.microsoft.com/en-us/dotnet/csharp/>

---

# RESUMEN

## Recordemos

Hoy en día las aplicaciones web son muy importantes y de uso muy difundido. Aprovechan el uso del navegador para que el usuario envíe sus solicitudes y es útil porque son aplicaciones multiplataforma y multidispositivo.

Las aplicaciones web requieren ser seguras, flexibles y escalables, sobre esa base Microsoft lanzó la tecnología .NET, que cuenta con una implementación multiplataforma denominada .NET Core (ahora .NET). .NET se complementa con el uso del framework ASP.NET Core.



---

# REFERENCIAS

## Para profundizar

[https://developer.mozilla.org/en-US/docs/Learn/Common\\_questions/Pages\\_sites\\_servers\\_and\\_search\\_engines](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Pages_sites_servers_and_search_engines)

<https://docs.microsoft.com/en-us/dotnet/>



# PREGRADO

## Ingeniería de Software

Escuela de Ingeniería de Sistemas y Computación | Facultad de Ingeniería



### UPC

Universidad Peruana  
de Ciencias Aplicadas

Prolongación Primavera 2390,  
Monterrico, Santiago de Surco  
Lima 33 - Perú  
T 511 313 3333  
<https://www.upc.edu.pe>

***exígete, innova***