



**UNIVERSIDAD
DON BOSCO**

DSP404 G01T

INVESTIGACION APLICADA -1

Participantes:

- Francisco Javier Salmerón Cornejo – SC251614
- Kevin Alexander Figueroa Rodríguez - FR251703
- Emanuel Ronaldo Monroy Paz - MP252018
- Franklin David Lovato Molina - LM162734
- Evelyn Marisol Sandoval Guillén - SG210007

Link hacia Video:

<https://youtu.be/RsywLkW7tXM?si=7JE7TWK4ZtDwzv5K>

Link hacia Aplicacion:

<https://drive.google.com/file/d/14VjAmwDzPXDZPzwbSZ8zkuCEnUprwp>

¿Qué es ASP.NET Core y cómo ha evolucionado desde ASP.NET clásico?

ASP.NET Core es un framework de desarrollo web de código abierto y multiplataforma creado por Microsoft. Es la evolución del ASP.NET clásico, el cual estaba limitado al sistema operativo Windows. ASP.NET Core fue diseñado para ser más liviano, modular y flexible, permitiendo el desarrollo de aplicaciones modernas que pueden ejecutarse en Windows, Linux y macOS. Además, ofrece un mejor rendimiento, integración con la nube y soporte para microservicios.

Principales características y ventajas de usar ASP.NET Core para el desarrollo web

- Multiplataforma (Windows, Linux, macOS).
- Código abierto con una gran comunidad.
- Alto rendimiento y escalabilidad.
- Arquitectura modular y ligera.
- Integración con servicios en la nube y contenedores (Docker, Kubernetes).
- Compatibilidad con patrones modernos de desarrollo como microservicios y REST APIs.

Arquitectura de una aplicación ASP.NET Core: MVC, Razor Pages, Blazor

ASP.NET Core permite varias formas de organizar las aplicaciones:

- MVC (Model-View-Controller): Ideal para aplicaciones estructuradas y escalables.
- Razor Pages: Simplifica el desarrollo de páginas web dinámicas con menos complejidad que MVC.
- Blazor: Permite crear aplicaciones interactivas del lado del cliente usando C# en lugar de JavaScript.

Comparación de usos con otras tecnologías web (Node.js, Django, Laravel, Spring Boot)

- Node.js: Excelente para aplicaciones en tiempo real, pero depende de JavaScript.
- Django: Framework en Python, muy usado en proyectos de ciencia de datos e inteligencia artificial.
- Laravel: Framework PHP, fácil de usar y con sintaxis clara, ideal para aplicaciones rápidas.
- Spring Boot: Framework en Java, altamente robusto y utilizado en aplicaciones empresariales. ASP.NET Core se destaca por su alto rendimiento y soporte

empresarial, integrándose de manera eficiente en entornos Microsoft.

Herramientas necesarias para desarrollar

- IDE: Visual Studio o Visual Studio Code.
- .NET SDK (para compilar y ejecutar aplicaciones).
- Gestores de bases de datos como SQL Server, PostgreSQL o MySQL.
- Sistemas de control de versiones como Git.
- Opcional: Docker y Kubernetes para despliegues en contenedores.

Posibles desventajas o retos al trabajar con esta tecnología

- Curva de aprendizaje para quienes no conocen C# o el ecosistema .NET.
- Documentación extensa que puede resultar abrumadora.
- Comunidad menos masiva que la de Node.js o Python.
- Puede requerir mayor consumo de recursos en comparación con frameworks más ligeros.

Importancia de aprender C# y ASP.NET Core en el contexto laboral actual

Aprender C# y ASP.NET Core es altamente relevante, ya que muchas empresas utilizan el ecosistema Microsoft para sus aplicaciones. El framework es ampliamente demandado en el desarrollo de aplicaciones empresariales, servicios en la nube (Azure) y soluciones escalables. Dominar estas tecnologías abre oportunidades laborales en entornos corporativos y en proyectos de gran envergadura.

