

Desarrollo de ASP.NET Core 6

Código: MSE-133

Propuesta de Valor: MICROSOFT

Duración: 32 Horas



.NET 6.0 es la próxima versión principal de .NET Core después de la versión 5. ASP.NET Core 6.0 se basa en .NET 6 pero conserva el nombre "Core" para evitar confusiones con ASP.NET MVC 6. .NET 6 también va acompañado de una nueva versión de C # e incluye importantes mejoras de rendimiento en muchos componentes.

Este curso de formación de desarrollo de ASP.NET Core 6 enseña a los asistentes cómo diseñar y crear aplicaciones API y de interfaz de usuario web bien diseñadas que siguen el patrón de diseño MVC. Este curso incluye 25 ejercicios prácticos de laboratorio individuales. Se discuten tecnologías relacionadas como gRPC, Blazor y Docker, pero no se tratan en detalle. Nota: Este curso de capacitación se puede extender con temas adicionales relacionados con el lenguaje de programación C # o Entity Framework Core.



AUDIENCIA

- Este curso está dirigido a desarrolladores web profesionales que usan Microsoft Visual Studio en un entorno de desarrollo basado en equipos individuales o de pequeño a gran tamaño. Los candidatos para este curso están interesados ??en desarrollar aplicaciones web avanzadas y desean administrar el HTML procesado de manera exhaustiva. Quieren crear sitios web que separen la interfaz de usuario, el acceso a los datos y la lógica de la aplicación con ASP.NET Core 6.



PRE REQUISITOS

Todos los asistentes deben tener:

- Experiencia previa en el desarrollo de aplicaciones basadas en web con C #.
- Alguna familiaridad con HTML, CSS y JavaScript.



OBJETIVOS

- Comprender los objetivos y beneficios de ASP.NET Core 6
- Aprenda a tomar buenas decisiones sobre la arquitectura de la aplicación y la elección de la tecnología de acceso a datos.
- Utilice el sistema de enrutamiento de ASP.NET para lograr una arquitectura de estilo REST
- Aprenda a crear una interfaz de usuario HTML atractiva y fácil de mantener utilizando el motor de visualización de Razor y JavaScript del lado del cliente.

- Obtenga experiencia en la construcción de un servicio que hace que los datos estén disponibles a través de una API web moderna
- Obtenga una introducción a Blazor, Razor Pages y gRPC
- Comprender las diferentes opciones de implementación multiplataforma disponibles, incluso a través de contenedores Docker.



CERTIFICACIÓN DISPONIBLE

- Certificado oficial de **MICROSOFT**.



CONTENIDO

1. INTRODUCCIÓN

- 1.1. EVOLUCIÓN DE .NET Y .NET CORE
- 1.2. .NET SKD Y TIEMPOS DE EJECUCIÓN
- 1.3. VISUAL STUDIO Y VISUAL STUDIO CODE

2. .NET 5.0 SDK

- 2.1. INSTALACIÓN
- 2.2. GESTIÓN DE VERSIONES
- 2.3. INTERFAZ DE LÍNEA DE COMANDOS (CLI)

3. NOVEDADES DE C # 9

- 3.1. TIPOS DE REGISTRO
- 3.2. ESTABLECEDORES DE SOLO INICIALIZACIÓN
- 3.3. TIPOS DE REFERENCIA QUE ACEPTAN VALORES NULL

4. ARQUITECTURA DE APLICACIONES ASP.NET CORE

- 4.1. PAQUETES NUGET
- 4.2. INICIO DE LA APLICACIÓN
- 4.3. ENTORNOS DE ALOJAMIENTO
- 4.4. MIDDLEWARE Y CANALIZACIÓN DE SOLICITUDES
- 4.5. SERVICIOS E INYECCIÓN DE DEPENDENCIA
- 4.6. MVC FRENTE A RAZOR PAGES

5. CONFIGURACIÓN DE LA APLICACIÓN

- 5.1. CONFIGURAR Y CONFIGURAR SERVICIOS
- 5.2. PROVEEDORES DE CONFIGURACIÓN Y FUENTES
- 5.3. API DE CONFIGURACIÓN
- 5.4. PATRÓN DE OPCIONES
- 5.5. HTTPS Y HTTP / 2

6. SOLICITAR ENRUTAMIENTO

- 6.1. SERVICIOS RESTFUL
- 6.2. ENRUTAMIENTO DE PUNTO FINAL
- 6.3. PLANTILLAS DE RUTA
- 6.4. RESTRICCIONES DE RUTA
- 6.5. PRIORIDAD DE LA PLANTILLA DE RUTA
- 6.6. ENRUTAMIENTO BASADO EN ATRIBUTOS

7. MODELOS

- 7.1. IGNORANCIA PERSISTENTE
- 7.2. INVERSIÓN DE DEPENDENCIA
- 7.3. ACCESO ASINCRÓNICO A DATOS
- 7.4. MAPEO RELACIONAL DE OBJETOS
- 7.5. ENTITY FRAMEWORK CORE
- 7.6. ORM ELEGANTE

8. CONTROLADORES

- 8.1. RESPONSABILIDADES
- 8.2. REQUISITOS Y CONVENCIONES
- 8.3. DEPENDENCIAS
- 8.4. RESULTADOS DE LA ACCIÓN

9. PUNTOS DE VISTA

- 9.1. RESPONSABILIDADES
- 9.2. CONVENCIONES
- 9.3. SINTAXIS DE RAZOR
- 9.4. DISEÑOS
- 9.5. VIEWDATA Y VIEWBAG
- 9.6. VISTAS FUERTEMENTE TIPADAS
- 9.7. VISTAS PARCIALES
- 9.8. AYUDANTES DE HTML Y URL
- 9.9. AYUDANTES DE ETIQUETAS
- 9.10. VER COMPONENTES
- 9.11. DEPENDENCIAS DEL LADO DEL CLIENTE
- 9.12. PÁGINAS DE RAZOR
- 9.13. VER MODELOS

10. FORMULARIOS HTML

- 10.1. AYUDANTES DE ETIQUETAS
- 10.2. ENVÍOS DE FORMULARIOS
- 10.3. ENCUADERNACIÓN DE MODELO

11. VALIDACIÓN DE ENTRADA

11.1. INTRODUCCIÓN

11.2. ANOTACIONES DE DATOS

11.3. ENCUADERNACIÓN DE MODELO

11.4. AYUDANTES DE ETIQUETAS DE ENTRADA

11.5. AYUDANTES DE ETIQUETAS DE VALIDACIÓN

12. ESTADO DE LA APLICACIÓN

12.1. DEL LADO DEL CLIENTE FRENTE AL LADO DEL SERVIDOR

12.2. HTTPCONTEXT.ITEMS

12.3. ESTADO DE LA SESIÓN

12.4. TEMPDATA

13. MANEJO DE ERRORES

13.1. MEJORES PRÁCTICAS

13.2. CÓDIGOS DE ESTADO DE ERROR HTTP

13.3. PÁGINAS DE CÓDIGOS DE ESTADO

13.4. PÁGINA DE EXCEPCIÓN PARA DESARROLLADORES

14. INICIO SESIÓN

14.1. CONFIGURACIÓN

14.2. ILOGGER

14.3. SERILOG Y SEQ

15. PRUEBAS

15.1. EXAMEN DE LA UNIDAD

15.2. XUNIDAD

15.3. CONTROLADORES DE PRUEBA

15.4. PRUEBAS DE INTEGRACIÓN

16. API WEB

16.1. CONTROLADORES API

16.2. OPENAPI / SWAGGER

16.3. PRUEBA DE API

16.4. OPERACIONES CRUD

16.5. USO COMPARTIDO DE RECURSOS DE ORIGEN CRUZADO (CORS)

17. SEGURIDAD

17.1. AUTENTICACIÓN

17.2. IDENTIDAD ASP.NET

17.3. AUTENTICACIÓN BASADA EN COOKIES

17.4. AUTORIZACIÓN

17.5. AUTENTICACIÓN DE API WEB

17.6. GESTIÓN DE SECRETOS

18. LLAMADAS A PROCEDIMIENTO REMOTO (GRPC)

18.1. INTRODUCCIÓN

18.2. PROTOBUF

18.3. SERVIDOR

18.4. CLIENTE

18.5. LIMITACIONES

19. BLAZOR

19.1. COMPONENTES DE LA MAQUINILLA DE AFEITAR

19.2. SERVIDOR BLAZOR

19.3. BLAZOR WEBASSEMBLY

20. DESPLIEGUE

20.1. DOTNET PUBLICAR

20.2. CERNÍCALO

20.3. IIS

20.4. ESTIBADOR

★ BENEFICIOS

- Al finalizar el curso, los participantes podrán comprender los objetivos y beneficios de ASP.NET Core 6.