

Tecnologías Usadas

Menú

1.	Back End	-02
A)	<u>Definicion de BackEnd</u>	-02
B)	<u>C#</u>	02
C)	<u>Framework</u>	03
D)	<u>.Net Core 6.0</u>	03
E)	<u>Entity Framework Core</u>	03
F)	<u>Entity Framework Core Jet</u>	03
G)	<u>Asp.Net Core</u>	03
H)	<u>Api Web</u>	-04
I)	<u>Dependency Injection</u>	04
J)	<u>Patron Repository</u>	04
K)	<u>Patrón MVC</u>	05
I.	<u>Modelo</u>	-05
II.	<u>Vista</u>	-05
III.	<u>Controlador</u>	05
IV.	<u>Enrutamiento</u>	-06
2.	Base de datos	06
A)	<u>MSAccess</u>	-06
B)	<u>Entity Framework en .NET Core</u>	-06
C)	<u>Comandos scaffold-dbcontext</u>	-07
3.	Frond-End	07
A)	<u>Definición</u>	07
B)	<u>HTML</u>	07
C)	<u>CSS</u>	07
D)	<u>JavaScript</u>	08
E)	<u>Bootstrap 5</u>	08
F)	<u>Iquery 3</u>	-08
I.	<u>Ajax</u>	08
II.	<u>Iquery & Ajax</u>	-09
4.	Referencias	-09

1. BackEnd

A. Definición

Es todo aquello que el usuario no ve, de hecho, se le denomina también “el lado del servidor” al estar alojado en la parte del servidor y únicamente ser accesible para los webmasters o programadores de la web o aplicación.

Lenguajes de backend

- Python.
- Node.js.
- PHP.
- Go.
- Ruby.
- C#.

B. C#

Es un lenguaje de programación diseñado para la infraestructura de lenguaje común. Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma .NET, similar al de Java.

C. Framework

Un framework es un esquema o marco de trabajo que ofrece una estructura base para elaborar un proyecto con objetivos específicos, una especie de plantilla que sirve como punto de partida para la organización y desarrollo de software. Los programadores utilizan frameworks en su día a día para desarrollar un software, crear una app, etc. porque les ayuda a ser más ágiles y productivos.

Es una herramienta que permite acelerar el trabajo y favorecer que este sea colaborativo, reducir errores y obtener un resultado de más calidad. El uso de frameworks permite principalmente agilizar procesos de desarrollo porque podemos reutilizar herramientas o módulos. Generalmente, los frameworks son usados por programadores porque permiten acelerar el trabajo y favorecer que este sea colaborativo, reducir errores y obtener un resultado de más calidad.

D. .NET Core

Es una plataforma de desarrollo de Microsoft de código fuente abierto, multiplataforma y de alto rendimiento para la creación de todo tipo de aplicaciones. Es un framework informático administrado, gratuito y de código abierto para los sistemas operativos Windows, Linux y macOS. Es un sucesor multiplataforma de .NET Framework y se lanzó el 27 de junio de 2016.

La versión más reciente de LTS (Long Term Support) es .NET 6, que se publicó en noviembre de 2021 y su soporte técnico dura tres años, hasta noviembre de 2024.

.NET 6 ofrece las partes finales del plan de unificación de .NET que se inició con .NET 5. .NET 6 unifica el SDK, las bibliotecas base y el entorno de ejecución en aplicaciones móviles, de escritorio, de IoT y en la nube.

E. Entity Framework Core

Entity Framework Core es una versión ligera, extensible, de código abierto y multiplataforma de la popular tecnología de acceso a datos Entity Framework. EF Core puede actuar como un asignador relacional de objetos, lo que permite a los desarrolladores de .NET trabajar con una base de datos utilizando objetos .NET. EF Core es compatible con muchos motores de base de datos.

F. Entity Framework Core Jet

Entity Framework Core Jet es un proveedor de Entity Framework Core para bases de datos Microsoft Jet/ACE (que admiten los formatos de archivo de base de datos de Microsoft Access MDB y ACCDB). El proveedor es compatible con .NET Standard 2.0, por lo que se puede usar con .NET (Core) 2.0+ y .NET Framework 4.6.1+.

G. Asp.Net Core

ASP.NET Core es un marco multiplataforma de código abierto para crear aplicaciones web modernas basadas en la nube en Windows, macOS y Linux. Los controladores de ASP.NET Core MVC y los controladores de API web están unificados.

H. Api Web

Una API web es una interfaz que permite a las aplicaciones comunicarse entre sí a través de Internet. Con ASP.NET Core, puedes crear una API web utilizando controladores o API mínimas. Los controladores de una API web son clases que se derivan de ControllerBase.

ASP.NET Web API es un marco que facilita la creación de servicios HTTP disponibles para una amplia variedad de clientes, entre los que se incluyen exploradores y dispositivos móviles. ASP.NET Web API es una plataforma ideal para compilar aplicaciones de RESTful en .NET Framework.

I. La inyección de dependencias

Es un patrón de diseño orientado a objetos que consiste en suministrar objetos a una clase en lugar de ser la propia clase la que cree dichos objetos. Estos objetos cumplen contratos que necesitan nuestras clases para poder funcionar, de ahí el concepto de dependencia. En la inyección de dependencias, un objeto recibe otros objetos en los que depende, y estos objetos se llaman dependencias.

La inyección de dependencias tiene como objetivo separar las preocupaciones de construir objetos y usarlos, lo que lleva a programas con un acoplamiento más bajo.

J. Patrón Repository

El patrón de repositorio es un patrón de diseño de software que se utiliza para abstraer el acceso a los datos de una aplicación. Este patrón se basa en la idea de que todos los datos de una aplicación deben estar almacenados en un único lugar, conocido como repositorio. De esta forma, la lógica de negocio de la aplicación puede acceder a los datos a través del repositorio, lo que permite separar la lógica de negocio de la lógica de acceso a datos. Esto facilita el mantenimiento y desarrollo de la aplicación, ya que permite cambiar fácilmente la fuente de datos utilizada sin tener que modificar el código de la lógica de negocio.

K. Patrón MVC

El patrón de arquitectura del controlador de vista de modelos (MVC) separa una aplicación en tres grupos de componentes principales: modelos, vistas y controladores. Este patrón permite lograr la separación de intereses. Con este patrón, las solicitudes del usuario se enrutan a un controlador que se encarga de trabajar con el modelo para realizar las acciones del usuario o recuperar los resultados de consultas. El controlador elige la vista para mostrar al usuario y proporciona cualquier dato de modelo que sea necesario.

I. Modelo

El modelo en una aplicación de MVC representa el estado de la aplicación y cualquier lógica de negocios u operaciones que esta deba realizar. La lógica de negocios debe encapsularse en el modelo, junto con cualquier lógica de implementación para conservar el estado de la aplicación. Las vistas fuertemente tipadas normalmente usan tipos ViewModel diseñados para que contengan los datos para mostrar en esa vista. El controlador crea y rellena estas instancias de ViewModel desde el modelo.

II. Vista

Las vistas se encargan de presentar el contenido a través de la interfaz de usuario. Usan el Razor motor de vista para insertar código .NET en el marcado HTML. Debería haber la mínima lógica entre las vistas y cualquier lógica en ellas debe estar relacionada con la presentación de contenido. Si ve que necesita realizar una gran cantidad de lógica en los archivos de vistas para mostrar datos de un modelo complejo, considere la opción de usar un componente de vista, ViewModel, o una plantilla de vista para simplificar la vista.

III. Controlador

Los controladores son los componentes que controlan la interacción del usuario, trabajan con el modelo y, en última instancia, seleccionan una vista para representarla. En una aplicación de MVC, la vista solo muestra información; el controlador controla y responde a la interacción y los datos que introducen los usuarios. En el patrón de MVC, el controlador es el punto de entrada inicial que se encarga de seleccionar con qué tipos de modelo trabajar y qué vistas representar (de ahí su nombre, ya que controla el modo en que la aplicación responde a una determinada solicitud).

Nota

Los controladores no deberían complicarse con demasiadas responsabilidades. Para evitar que la lógica del controlador se vuelva demasiado compleja, saque la lógica de negocios fuera el controlador y llévela al modelo de dominio.

Sugerencia

Si piensa que el controlador realiza con frecuencia los mismos tipos de acciones, mueva estas acciones comunes en filtros.

IV. Enrutamiento

El enrutamiento es responsable de hacer coincidir las solicitudes HTTP entrantes y de enviarlas a los puntos de conexión ejecutables de la aplicación. Los puntos de conexión son las unidades de código de control de solicitudes ejecutable de la aplicación. Se definen en la aplicación y se configuran al iniciarla. El proceso de búsqueda de coincidencias de puntos de conexión puede extraer valores de la dirección URL de la solicitud y proporcionarlos para el procesamiento de la solicitud. Con la información de los puntos de conexión de la aplicación, el enrutamiento también puede generar direcciones URL que se asignan a los puntos de conexión.

2. Base de Datos

A. MSAccess

Microsoft Access es un sistema de gestión de bases de datos incluido en las ediciones profesionales de la suite Microsoft Office. Es el sucesor de Embedded Basic.

Access es un gestor de datos que utiliza los conceptos de bases de datos relacionales y pueden manejarse por medio de consultas e informes. Está adaptado para recopilar datos de otras utilidades como Excel, SharePoint, etc.

La aplicación permite recopilar información relativa a un asunto o propósito particular, como el seguimiento de pedidos de clientes o el mantenimiento de una colección de música, videojuegos, huso horario, etc.

Muy sencillo de utilizar, es un programa con una base de datos que se utiliza para guardar información importante en diferentes casos.

B. Entity Framework en .NET Core

- Entity Framework (EF) Core es una versión ligera, extensible, de código abierto y multiplataforma de la popular tecnología de acceso a datos Entity Framework.
- EF Core puede actuar como asignador relacional de objetos, que se encarga de lo siguiente:
- Permite a los desarrolladores de .NET trabajar con una base de datos usando objetos .NET.
- Permite prescindir de la mayor parte del código de acceso a datos que normalmente es necesario escribir.

EF Core es compatible con muchos motores de base de datos; vea [Proveedores de bases de datos](#) para más información.

C. Comandos scaffold-dbcontext

Los comandos Scaffold-DbContext de Entity Framework ayudan a crear scaffolding de clases de tipo de entidad y una clase DbContext basada en un esquema de base de datos, automatizando así la técnica de generación de código relacionada con el acceso a la base de datos.

Disminuye la cantidad de código escrito manualmente y el mantenimiento requerido para las aplicaciones orientadas a datos.

3. Front-end

A. Definición

El **desarrollo web *front-end*** consiste en la conversión de datos en una interfaz gráfica para que el usuario pueda ver e interactuar con la información de forma digital usando HTML, CSS y JavaScript.

B. Hyper Text Markup Language (HTML)

HTML, siglas en inglés de Hyper Text Markup Language (lenguaje de marcas de hipertexto), es la columna vertebral de cualquier proceso de desarrollo web, sin el cual las páginas web no existirían. Hipertexto significa que el texto tiene enlaces, denominados hipervínculos, incrustados en el mismo. Cuando el usuario hace clic en una palabra o una frase que tiene un hipervínculo, se realiza una llamada a otra página web. El lenguaje de marcas indica que el texto puede convertirse en imágenes, tablas, enlaces, y otras representaciones. El código HTML es el que provee un marco general de cuál será el aspecto del sitio. HTML fue desarrollado por Tim Berners-Lee. La última versión de HTML se llama HTML5 y se publicó en 28 de octubre de 2014, por recomendación de W3C. Esta versión contiene nuevas y eficientes maneras de manejar elementos como archivos de vídeo y audio.

C. Cascading Style Sheets (CSS)

Las hojas de estilo en cascada, en inglés Cascading Style Sheets, controlan la apariencia visual del sitio web y permiten darle un aspecto único. Esto se consigue con hojas de estilo que se superponen sobre otras reglas de estilo y se lanzan basándose en el tipo de entrada, como el tamaño y la resolución de la pantalla del dispositivo.

D. JavaScript

JavaScript es un lenguaje de programación imperativa basado en eventos (a diferencia del modelo de programación declarativa de HTML) que se utiliza para transformar una página HTML estática en una interfaz dinámica. El código JavaScript puede utilizar Document Object Model (DOM), proporcionado por el estándar HTML, para manipular una página web en la respuesta a eventos, como las entradas de usuario.

Utilizando una técnica llamada AJAX, el código JavaScript también puede recoger contenido activamente de la web (independientemente de la página HTML original recibida), y además reacciona ante eventos del lado del servidor, dotando de una naturaleza realmente dinámica a la experiencia de la página web.

E. Bootstrap

Bootstrap es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como extensiones de JavaScript adicionales. A diferencia de muchos frameworks web, solo se ocupa del desarrollo front-end.

Bootstrap es el segundo proyecto más destacado en GitHub¹ y es usado por la NASA y la MSNBC, entre otras organizaciones.

F. jQuery

Es una biblioteca multiplataforma de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC. De acuerdo a un análisis de la Web (realizado en 2017) jQuery es la biblioteca de JavaScript más utilizada, por un amplio margen.

jQuery, en su núcleo, es una biblioteca de manipulación de Modelo de Objetos del Documento (DOM). El DOM es una representación en estructura de árbol de todos los elementos de una página web. jQuery simplifica la sintaxis para buscar, seleccionar y manipular estos elementos DOM. Por ejemplo, jQuery se puede usar para encontrar un elemento en el documento con una determinada propiedad (por ejemplo, todos los elementos con una etiqueta h1), cambiar uno o más de sus atributos (por ejemplo, color, visibilidad) o hacer que responda a un evento (por ejemplo, un clic del ratón).

I. Ajax

AJAX = JavaScript asíncrono y XML.

En resumen; AJAX se trata de cargar datos en segundo plano y mostrarlos en la página web, sin recargar toda la página.

Ejemplos de aplicaciones que usan AJAX: Gmail, Google Maps, YouTube y Pestañas de Facebook.

II. JQuery & Ajax

jQuery proporciona varios métodos para la funcionalidad AJAX.

Con los métodos jQuery AJAX, puede solicitar texto, HTML, XML o JSON desde un servidor que utiliza HTTP Get y HTTP Post: y puede cargar los datos externos directamente en los elementos HTML seleccionados de su ¡página web!

Referencias

[¿Qué es el Backend? Descubre todo sobre este concepto \(epitech-it.es\)](#)

[C Sharp - Wikipedia, la enciclopedia libre](#)

[Framework: qué es, para qué sirve y algunos ejemplos \(edix.com\)](#)

[Framework - Wikipedia, la enciclopedia libre](#)

[.Net Core](#)

[.Net Core 6](#)

[Información general de Entity Framework Core - EF Core | Microsoft Learn](#)

[Releases · bubibubi/EntityFrameworkCore.Iet \(github.com\)](#)

[Enlace de parámetros en ASP.NET Web API: ASP.NET 4.x | Microsoft Learn](#)

[Creación de API web con ASP.NET Core | Microsoft Learn](#)

[\[https://es.wikipedia.org/wiki/Inyecci%25C3%25B3n_de_dependencias\]\(https://es.wikipedia.org/wiki/Inyecci%25C3%25B3n_de_dependencias\)](#)

[\[https://en.wikipedia.org/wiki/Dependency_injection\]\(https://en.wikipedia.org/wiki/Dependency_injection\)](#)

[Patrón Repository \(geekebrains.com\)](#)

[Información general de ASP.NET Core MVC | Microsoft Learn](#)

[Enrutamiento en ASP.NET Core | Microsoft Learn](#)

[Microsoft Access - Wikipedia, la enciclopedia libre](#)

[Información general de Entity Framework Core - EF Core | Microsoft Learn](#)

[EJEMPLO DE COMANDOS EFCORE SCAFFOLD-DBCONTEXT EN .NET CORE | THECODEBUZZ](#)

[Desarrollo web Front-end - Wikipedia, la enciclopedia libre](#)

[Bootstrap \(framework\) - Wikipedia, la enciclopedia libre](#)

[jQuery - Wikipedia, la enciclopedia libre](#)

[jQuery AJAX Introducción \(w3schools.com\)](#)