TEORICA

Describa los conceptos de los siguientes temas:

1. SSIS

SSIS es una herramienta para crear soluciones de integración de datos que pueden ayudar a las organizaciones a optimizar sus flujos de trabajo de procesamiento y análisis de datos, y hacer un mejor uso de sus datos.

1. Microservicios

Son un enfoque para el desarrollo de software que estructura una aplicación como una colección de pequeños servicios independientes. Cada servicio está diseñado para realizar una función empresarial específica y se comunica con otros servicios a través de protocolos ligeros como REST \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. MVC

MVC significa Modelo-Vista-Controlador, que es un patrón de software utilizado para desarrollar aplicaciones que separa la lógica de la aplicación en tres componentes interconectados: el modelo, la vista y el controlador.

El controlador hace de intermediario entre la vista y el modelo de datos, el modelo representa la lógica del negocio y manipulación de data y la vista representa lo que ve el usuario. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. SQL Server

SQL Server es una plataforma de base de datos robusta y confiable que proporciona potentes funciones para el almacenamiento, la gestión y el análisis de datos. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. .NET Core

.NET Core es un framework multiplataforma de código abierto muy potente y versátil para crear aplicaciones que pueden ejecutarse en diferentes sistemas operativos y dispositivos, con un alto rendimiento y una sólida comunidad de desarrolladores y colaboradores. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Azure DevOps Server

Azure DevOps Server (anteriormente conocido como Team Foundation Server o TFS) es una plataforma de colaboración para equipos de desarrollo de software, desarrollada por Microsoft. Proporciona un conjunto de herramientas para administrar todo el ciclo de vida del desarrollo de software, desde la planificación y la codificación hasta la creación, prueba e implementación de aplicaciones. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Practico**

1. **PRUEBA ETL -** Realizar paquete mediante el componente SSIS (SQL Server Integration Services), se debe leer 2 tablas a través de un **ETL.**

* **Persona** contiene los siguientes campos: Nombres, Apellidos, Cédula
* **Información** contiene los siguientes campos: Cedula, Teléfono, Correo, Dirección.

El paquete debe genera una nueva tabla con los datos de las tablas Persona, Información.

1. **PRUEBA DESKTOP -** Realizar un aplicativo que permita el mantenimiento de software utilizando lenguaje de programación C# (aplicativo desktop) en donde permita el ingreso de los siguientes datos en la tabla **Cliente**:
   * Nombre
   * Apellido
   * Cedula
   * Teléfono
   * Correo
   * Foto de perfil
   * Cursos

Considerar que se realice la validación de los dígitos de la cedula, y un correo valido, la foto de perfil debe estar en formato .png, .jpg.

**Nota: Debe utilizar capas**

1. **PRUEBA WEB -** La empresa NEXTI te encarga la construcción de un aplicativo para la venta de entradas para acontecimientos deportivos y culturales. Se quiere que la aplicación funcione en la web, de tal manera que los usuarios accedan a través de un navegador. Los usuarios podrán registrar, eliminar, modificar y consultar las características relativas al evento:

* Fecha evento
* Lugar evento
* # Entrada
* Descripción del evento
* Precio

Los datos permitidos de modificación son:

1. Fecha de Evento

La eliminación del evento debe ser lógica a nivel de BDD

1. Fecha de Evento

Se debe grabar un LOG de todas las acciones realizadas en la página web

Para lo cual se debe generar un Microervicio con diferentes endpoints para consultar, crear, modificar y eliminar los datos ingresados en la página web.

**La tecnología a utilizar es:**

* + **Frontend ASP.Net framework - C# (Jquery, ASP.NET, C#)**
  + **Backend C# Asp .Net Core 6 (API Rest) - Microservicios**
  + **Bases de datos SQL Server**

**Documentación PRUEBA WEB**

* **Flow Diagram:**

User --> |Web Application| --> |API| --> |Database|

* **Sequence Diagram:**

User --> |Web Application|: Enter event details

Web Application --> |API|: POST /events

API --> |Database|: Insert event

Database --> |API|: Return event ID

API --> |Web Application|: Return event ID

Web Application --> |User|: Show confirmation

* **ERD (Entity Relationship Diagram**

User 1 ---- \* Event