

PRUEBA TÉCNICA PRÁCTICA PRESENCIAL

Duración	2h 30m
Modalidad	Presencial
Tipo	Evaluación Individual

El objetivo de esta prueba es evaluar sus conocimientos técnicos en desarrollo backend, manejo de base de datos, buenas prácticas de programación y resolución de problemas en un entorno similar a un sistema empresarial real.

Se evaluará:

- Lógica de programación
- Estructura y calidad del código
- Manejo de base de datos SQL Server
- Manejo de errores
- Uso de transacciones
- Buenas prácticas de desarrollo

CONTEXTO DEL CASO

La empresa requiere implementar un servicio para registrar pedidos dentro de un sistema transaccional.

Cada pedido debe:

1. Registrarse en base de datos
2. Registrar el detalle de productos
3. Validarse mediante un servicio externo simulado
4. Registrar eventos de auditoría
5. Mantener consistencia mediante transacciones

CONTROL DE VERSIONES (REQUERIDO)

Durante la prueba deberá:

- Crear un repositorio Git personal (GitHub, GitLab o local)
- Inicializar el repositorio desde el inicio del desarrollo
- Realizar commits durante el desarrollo
- Mantener un historial coherente de cambios
- Entregar la URL del repositorio al finalizar (si es remoto) o carpeta .git

REQUERIMIENTO PRINCIPAL

Desarrollar un API REST en .NET que permita registrar pedidos.

Debe crear el siguiente endpoint:

POST /api/pedidos

El endpoint debe recibir un objeto JSON con la información del pedido.

Ejemplo de request

```
{
  "clienteId": 1001,
  "usuario": "usuario.prueba",
  "items": [
    { "productoId": 1, "cantidad": 2, "precio": 10 },
    { "productoId": 2, "cantidad": 1, "precio": 20 }
  ]
}
```

BASE DE DATOS

Debe crear las tablas necesarias para almacenar la información.

Estructura mínima esperada

PedidoCabecera

- Id
- ClientId
- Fecha
- Total
- Usuario

PedidoDetalle

- Id
- PedidoId
- ProductId
- Cantidad
- Precio

LogAuditoria

- Id
- Fecha
- Evento
- Descripcion

Puede proponer mejoras a la estructura si lo considera necesario.

VALIDACIÓN CON SERVICIO EXTERNO

Antes de confirmar el pedido, debe consumir un servicio externo de validación de cliente.

Puede utilizar el siguiente endpoint público:

`https://jsonplaceholder.typicode.com/users/1`

Consideraciones

- Si la validación es correcta → continuar proceso
- Si ocurre error en la llamada → manejar el error
- Debe contemplar fallas del servicio externo

MANEJO TRANSACCIONAL (OBLIGATORIO)

El proceso completo debe ejecutarse dentro de una transacción.

Si ocurre cualquier error:

- No debe guardarse información parcial
- Debe realizar rollback

Si todo es correcto:

- Confirmar transacción

MANEJO DE ERRORES

Debe manejar adecuadamente:

- errores de base de datos
- errores de servicio externo
- datos inválidos
- excepciones generales

Debe retornar respuestas HTTP coherentes.

REGISTRO DE EVENTOS (LOGGING)

Debe registrarse información básica del proceso:

- inicio del registro
- errores producidos
- confirmación exitosa

El registro puede realizarse en:

- tabla de base de datos
- archivo

- consola
- ILogger

CONSIDERACIONES TÉCNICAS

Se evaluará:

- Organización del proyecto
- Separación por capas (controller, lógica, acceso datos)
- Claridad del código
- Buenas prácticas
- Manejo de excepciones
- Uso adecuado de transacciones

Puede utilizar:

- .NET Web API
- SQL Server
- ADO.NET, Dapper o Entity Framework

ENTREGA

Al finalizar el tiempo deberá entregar:

- Proyecto fuente completo
- Script de base de datos
- Instrucciones breves de ejecución (si aplica)

CRITERIOS DE EVALUACIÓN

Se evaluará:

- Correcto funcionamiento
- Calidad del código
- Diseño de la solución
- Manejo de errores
- Uso de transacciones
- Orden y claridad
- Uso correcto de Git
- Frecuencia de commits
- Claridad de mensajes de commit
- Organización del repositorio

TIEMPO DISPONIBLE

Tiempo total: 2 horas 30 minutos

Se notificará cuando resten:

- 60 minutos
- 30 minutos
- 10 minutos

IMPORTANTE

- La prueba es individual
- No se permite copiar código de otros candidatos
- Puede realizar consultas técnicas generales al evaluador si alguna instrucción no es clara
- Se evaluará tanto el resultado como el enfoque utilizado