

## GESTIÓN DE BASES DE DATOS PROYECTO INDIVIDUAL

### Propósito:

Resolver problemas de la clientela mediante la creación de base de datos, según normas y principios de diseño.

### Tipo de evidencias:

Desempeño, producto, actitud. Informe con la solución del proyecto empresarial mediante el diseño e implementación de la base de datos.

### Procedimiento:

1. Cualquier intento de fraude anula completamente su proyecto.
2. Este proyecto individual debe presentarse por etapas o fases en las fechas establecidas por el docente, a continuación se realiza el desglose del proyecto individual:
  - a. **Fase 1:** Diseño de la base de datos: **Trabajo escrito, 25%**
  - b. **Fase 2:** Implementación de la base datos: **Tablas (Scripts) 25%**
  - c. **Fase 3:** Diseño de consultas: **Consultas, Triggers, SP, Funciones (Scripts) 40%**
3. Debe realizar un análisis del problema asignado para solucionarlo mediante el diseño e implementación de la base de datos, además del diseño de consultas, procedimientos almacenados, funciones y triggers.
4. Utilice los datos de formato y de presentación para elaborar el documento con el diseño de la base de datos, fase 1.
5. Implemente la solución en una base de datos relacional según el diseño planteado, fase 2.
6. Diseñe las consultas, procedimientos almacenados, funciones y triggers, según los requerimientos planteados para resolver el problema, fase 3.

### Descripción del problema:

### Instituto Especializado de Entrenamiento en Triatlón (IEET)

Usted debe crear una base de datos para un Instituto que forma atletas en triatlón, se requiere que la base de datos administre la matrícula de estudiantes con sus respectivos pagos de inscripción y mensualidades, la información de las personas entrenadoras y los programas impartidos. Un programa se imparte a un grupo de atletas, tiene una fecha de inicio y una fecha de fin que el sistema debe calcular automáticamente.

La información básica que usted maneja para crear el sistema es la siguiente:

1. Se atiende grupos de un mínimo de 5 personas y un máximo de 15 personas

2. El programa comprende 4 módulos:
  - a. Gimnasio, 80 horas
  - b. Natación, 110 horas
  - c. Maratón, 120 horas
  - d. Ciclismo, 90 horas
3. Cada entrenamiento diario tiene una duración de dos horas diarias (no fragmentadas), por ejemplo una persona podría tener su sesión de entrenamiento de 9:00 a.m. a 11:00 a.m., pero no es posible que vaya una hora de 9:00 a.m a 10:00 a.m y en la tarde vaya otra hora de 3:00 p.m. a 4:00 p.m.
4. El sistema debe registrar los pagos mensuales y generar informes para determinar quiénes están al día y quiénes no. El costo de la matrícula es de 15 000 colones, y cada módulo cuesta 125 000 colones.
5. Las personas entrenadoras tienen derecho a 5 semanas de vacaciones al año, las cuales pueden fragmentar y disfrutar en la fecha en que cada persona lo decida. Las vacaciones se ingresan al sistema a inicio de año para cada persona, para poder calcular las fechas. Si una persona cambia sus fechas de vacaciones en el transcurso del año, esto modificará las fechas de los programas que imparte.
6. Las personas entrenadoras pueden tener incapacidades, permisos u otras situaciones no planificadas que también pueden afectar las programaciones, y la base de datos debe ser capaz de recalculas las fechas involucradas.
7. Las personas entrenadores tienen certificaciones en cada una de las 4 áreas, no necesariamente todas las personas entrenadoras tienen certificación para impartir los 4 módulos.
8. Las personas entrenadoras tienen un horario de trabajo definido por días y horas, por ejemplo una persona entrenadora puede trabajar de lunes a viernes o podría trabajar sólo algunos días a la semana, además tienen un horario, por ejemplo su horario podría ser de 10:00 a.m. a 2:00 p.m.
9. Cada curso tiene calificaciones finales, cuya nota mínima es de 70 para continuar en el programa
10. Un estudiante no puede avanzar al módulo siguiente a menos que haya aprobado el módulo anterior
11. Cada programa nuevo tiene asignado un horario, el cual corresponde a 2 horas diarias. Este horario se mantiene igual para los 4 módulos.
12. La base de datos debe de ser capaz de:
  - a. Ingresar la información de los atletas
  - b. Ingresar la información de las personas entrenadoras, y administrar también sus fechas de vacaciones, periodos de incapacidades, licencias u otros
  - c. Crear un programa nuevo a impartir y matricular atletas en un programa nuevo, cada programa tendrá un código independiente
  - d. En un mismo programa, es posible los 4 módulos los imparta la misma persona entregadora o también que sean diferentes personas entrenadoras quienes impartan cada módulo (dependiendo si tienen las certificaciones requeridas, lo cual debe verificar el sistema)
  - e. Al abrir un nuevo programa se ingresa: los atletas que formarán el grupo (previamente matriculados y previo pago del curso), la persona instructora que impartirá cada uno de los módulos, el horario en que se ejecutará el programa (debe verificarse que las personas instructoras asignadas no tengan choque de horario con otro curso ya que estén impartiendo, y además que el horario del módulo no esté fuera del horario de la persona docente)
  - f. La base de datos calculará automáticamente la fecha de fin de todo el programa para ese grupo.
  - g. En caso de presentarse alguna situación especial que afecte las fechas, la base de datos calculará automáticamente el cambio en la fecha final.

13. Estas son las descripciones básicas, cualquier otra información que usted requiera definir para crear el sistema, **puede definirla en su análisis** y anotarlo como suposiciones en su estrategia.

**La base de datos debe tener los siguientes requerimientos mínimos:**

1. Confeccionar el informe escrito de la base de datos, donde se incluya todo el análisis de la fase 1 (Los 13 aspectos descritos en el apartado siguiente)
2. Al menos 6 tablas (utilizar formatos básicos, normalización y propiedades de campos)
3. Establecer relaciones entre tablas (con integridad referencial)
4. En las tablas al menos 6 campos o atributos y al menos 10 registros en cada tabla
5. Al menos **20 consultas**, desglosadas de la siguiente manera:
  - a. **6 consultas de selección simples** con varias tablas y utilizando criterios (utilice varias formas de join)
  - b. **8 consultas de selección utilizando funciones agregadas** (having, group by), utilice funciones aritméticas, de manejo de cadenas y de fecha
  - c. **2 consultas de selección ordenando datos**
  - d. **2 consultas de selección con unión o con subconsultas**
  - e. **2 consultas** del tipo que usted seleccione
6. Guardar todas las consultas cada una en una vista (o el menos las que lo permitan)
7. Al menos 4 **procedimientos almacenados** (CRUD "Crear, Leer, Actualizar, Borrar") para cada tabla
8. Al menos 2 **procedimientos almacenados** donde se evidencie el manejo de transacciones y la lógica planteada
9. Al menos 2 **Funciones**
10. Al menos 1 **Trigger** por cada tabla
11. No debe usar la vista de diseño, solamente el editor de consultas
12. Sea creativo, ordenado y puntual

#### FASE 1 (valor 25%)

**Objetivo de la Fase 1:**

- Realizar el diseño de la base de datos de acuerdo al caso proporcionado.

**Objetivos específicos de la Fase 1:**

1. Realizar el análisis del problema que debe solucionar la base de datos
2. Determinar los requerimientos funcionales
3. Determinar las reglas básicas del negocio
4. Crear el modelo conceptual de la base de datos
5. Plantear a nivel conceptual la solución
6. Crear el modelo lógico de la base de datos
7. Crear el diccionario de datos de la base de datos

**Estructura del documento de la fase 1:**

1. Portada
2. Tabla de contenidos
3. Introducción
4. Objetivo principal y específicos
5. Definición del problema
6. Análisis de requerimientos funcionales
7. Reglas básicas de negocio
8. Modelo conceptual de la base de datos
9. Planteamiento de nivel conceptual de la solución
10. Modelo lógico de la base de datos
11. Diccionario de datos
12. Conclusión
13. Anexos (de ser necesario)

**Formato del documento y notas importantes:**

1. El formato que debe usar para el informe final es el siguiente:
  - a. Título 1: Fuente Verdana, tamaño 14 con negrita, espaciado anterior y posterior de 6 ptos
  - b. Título 2: fuente Verdana. Tamaño 12 subrayado, espaciado anterior y posterior de 6 ptos
  - c. Texto Normal: Fuente **Calibri Light**, tamaño 11, sin estilos, espaciado anterior y posterior de 6 ptos, interlineado sencillo, sin sangrías
2. El informe debe estar ordenado y sin faltas de ortografía
3. Los anexos deben ser cualquier documentación adicional que se considere necesaria
4. El proyecto no se recibirá en una fecha que no sea la indicada

**FASE 2 (valor 25%) Y FASE 3 (valor 40%)**

Las fases 2 y 3 no tienen informe escrito, ya que ambas **se basan en la Fase 1**.

Las fases 2 y 3 consisten en **implementar** lo que usted describió en la Fase 1.

Sin embargo, en las fases 2 y 3 debe **volver a presentar** el informe de la Fase 1 **realizando todas las correcciones** que le indique la persona docente.

PROYECTO FASE 1. Valor 25%  
Escala numérica de evaluación

Estudiante:

Lista de rasgos observables	Criterios		
	CC	CI	II
	2	1	0
<b>En cuanto al plazo de entrega</b>			
Entrega la fase 1 en la fecha indicada			
<b>En cuanto a la presentación, incluye:</b>			
Portada con todos sus datos y los datos del módulo			
Tabla de contenidos creada con la herramienta respectiva de MS Word			
Numera las páginas			
Incluye introducción redactada en tercera persona y acorde al proyecto			
Incluye el objetivo principal y objetivos específicos			
Incluye el problema empresarial a resolver			
Incluye conclusión			
Utiliza el formato de fuente y párrafo solicitados			
El documento presentado refleja orden y buena presentación			
<b>En cuanto al desarrollo:</b>			
Realiza un análisis de los requerimientos funcionales			
Especifica las reglas del negocio			
Crea el <b>modelo conceptual</b> de la base de datos			
- El modelo conceptual determina las entidades importantes			
- El modelo conceptual incluye las relaciones entre las entidades			
- El modelo conceptual <b>no</b> especifica atributos ni claves principales			
- Utiliza un software de modelado de datos			
Plantea la solución a nivel conceptual			
- La solución planteada describe adecuadamente todos los procesos necesarios para resolver el problema			
Crea el <b>modelo lógico</b> de la base de datos			
- El modelo lógico incluye todas las entidades y las relaciones entre ellas			
- El modelo lógico especifica todos los atributos de cada entidad			
- La clave principal para cada entidad está especificada			
- Se especifican las claves externas			
- El modelo lógico está normalizado según las normas establecidas			
- Utiliza un software de modelado de datos			
Crea el <b>diccionario de datos</b> de la base de datos			
- Define los nombres de los campos y son nombres significativos			
- Define de forma correcta el tipo de datos de cada campo			

- Define la propiedad de requerido o no requerido para cada campo de forma lógica			
- Define las restricciones en los campos que las requieran y dichas restricciones son correctas			
- Define llaves primarias, secundarias, valores permitidos y demás observaciones en los campos que lo requieran y dichas definiciones son correctas y funcionales			
Total:			
<b>Simbología:</b>  <b>CC:</b> correcto y completo <b>CI:</b> correcto pero incompleto <b>II:</b> incorrecto e incompleto	<b>Puntos obtenidos:</b>		
	<b>Porcentaje obtenido:</b>		
	<b>Calificación:</b>		

Observaciones:

**PROYECTO FASE 2. Valor 25%**  
**Escala numérica de evaluación**

**Objetivo:** Implementar base de datos relacional según los requerimientos de la clientela.

**Estudiante:**

Lista de rasgos observables	Criterios		
	CC	CI	II
	2	1	0
<b>En cuanto al plazo de entrega</b>			
Entrega la fase 2 en la fecha indicada			
<b>En cuanto a la estructura</b>			
Los archivos de los Scripts se guardaron con nombres significativos y claros			
<b>En cuanto a la presentación</b>			
En el Script documenta las tablas			
En el Script documenta las restricciones			
Los comentarios en los Scripts no contienen errores ortográficos ni gramaticales			
El código del Script se encuentra en orden lógico y tabulado correctamente			
<b>En cuanto al desarrollo de la Fase 2</b>			
Crea los Scripts de la base de datos utilizando la herramienta SQL Server			
Asigna un nombre lógico y significativo a la base datos			
Realiza las correcciones a la FASE 1 y presenta el documento corregido			
Las correcciones realizadas a la FASE 1 son correctas			
Crea las tablas (mínimo 6) según el diseño planteado en la FASE 1			
El nombre de las tablas es un nombre lógico			
Crea los atributos a cada una de las tablas			
Los atributos fueron creados del tipo correcto			
Utiliza las propiedades de los atributos eficientemente			
Crea las llaves primarias según el diseño			
Crea llaves foráneas según el diseño de forma lógica y eficiente			
Define restricciones de integridad			
Gestiona índices			
Crea un Script que inserte datos en todas las tablas (mínimo 10 registros por tabla)			
Los datos se insertan correctamente			
Los datos insertados respetan las reglas de integridad			
Todos los Scripts se ejecutan sin errores			
Utiliza buenas prácticas para el diseño de la base de datos			
Total:			
<b>Puntos obtenidos:</b>			

Simbología:

**CC:** correcto y completo  
**CI:** correcto pero incompleto  
**II:** incorrecto e incompleto

Porcentaje obtenido:	
Calificación:	

Observaciones:



**PROYECTO FASE 3. Valor 40%**  
**Escala numérica de evaluación**

**Objetivo:** Diseñar consultas en lenguaje estructurado según los requerimientos de la clientela.

**Estudiante:**

Lista de rasgos observables	Criterios		
	CC	CI	II
	2	1	0
<b>En cuanto al plazo de entrega</b>			
Entrega la fase 3 en la fecha indicada			
<b>En cuanto a la estructura</b>			
Los archivos de los Scripts se guardaron con nombres significativos y claros			
<b>En cuanto a la presentación</b>			
En el Script documenta todos los procedimientos creados			
Los comentarios en los Scripts no contienen errores ortográficos ni gramaticales			
El código del Script se encuentra en orden lógico y tabulado correctamente			
<b>En cuanto al desarrollo de la Fase 3</b>			
No presenta errores de lógica			
Realiza al menos 6 <b>consultas de selección simples</b> con varias tablas y varios criterios, utilizando con varias formas de Join			
Las <b>consultas de selección simples</b> son útiles para el usuario, ayudan a la solución del problema y devuelven datos correctos.			
Realiza al menos 8 <b>consultas de selección con funciones agregadas</b> (having, group by)			
En el punto anterior utiliza funciones aritméticas, de manejo de cadenas y de fecha			
Las <b>consultas de selección con funciones agregadas</b> son útiles para el usuario, ayudan a la solución del problema y devuelven datos correctos.			
Realiza al menos 2 <b>consultas de selección ordenando datos</b>			
Las <b>consultas de selección ordenando datos</b> son útiles para el usuario, ayudan a la solución del problema y devuelven datos correctos.			
Realiza al menos 2 <b>consultas de selección con unión o con subconsultas</b>			
Las <b>consultas de selección con unión o con subconsultas</b> son útiles para el usuario y ayudan a la solución del problema			
Crea al menos 2 <b>consultas del tipo que elija</b> , son útiles para el usuario, ayudan a la solución del problema y devuelven datos correctos.			
Las consultas se guardan cada una en una vista (o al menos las que lo permitan)			
Crea al menos 4 <b>procedimientos almacenados</b> (CRUD) para cada tabla			
Crea al menos 2 <b>procedimientos almacenados</b> donde se evidencia el manejo de transacciones y la lógica planteada			
Los <b>procedimientos almacenados</b> son correctos y funcionales			
Los <b>procedimientos almacenados</b> evidencian la importancia de las transacciones			

Declara <b>variables</b> y las aplica correctamente			
Utiliza instrucciones de control de flujo correctamente			
Utiliza manejo de excepciones			
Crea al menos 2 <b>Funciones</b>			
Las <b>Funciones</b> creadas son útiles y aplicables			
Crea al menos un <b>Trigger</b> por cada tabla			
Los <b>Triggers</b> creados tienen una funcionalidad en la solución del problema			
Los <b>Triggers</b> y procedimientos almacenados <b>no replican funciones</b>			
Los Scripts se ejecutan sin producir errores			
Es creativo(a) y ordenado(a)			
Total:			
<b>Simbología:</b>  <b>CC:</b> correcto y completo <b>CI:</b> correcto pero incompleto <b>II:</b> incorrecto e incompleto	<b>Puntos obtenidos:</b>		
	<b>Porcentaje obtenido:</b>		
	<b>Calificación:</b>		

Observaciones: