



## **Proyecto de Sistemas de Bases de Datos I**

Plataforma de gestión de academias interdisciplinarias en **Unidad Educativa Particular Daniel Comboni**

**Sistemas de Bases de Datos I  
Segundo Término 2025-2026**

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL  
FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN**

<b>Materia:</b> Sistemas de Bases de Datos 1	<b>Facultad de Ingeniería en Electricidad y Computación</b>
<b>Profesor:</b> MSc. Irene M. Cheung Ruiz	<b>ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL</b>
<b>Última Revisión:</b> Fecha	<b>Pág. 1 de 15</b>



## Índice

### Tabla de contenido

<i>Integrantes</i> .....	<b>3</b>
<i>Título del Proyecto</i> .....	<b>3</b>
<i>Objetivos del Negocio del cual se basa el Proyecto</i> .....	<b>3</b>
<i>Descripción General</i> .....	<b>3</b>
<i>Descripciones Funcionales</i> .....	<b>3</b>
<i>Modelo Conceptual</i> .....	<b>7</b>
<i>Modelo Lógico</i> .....	<b>7</b>
<i>Reportes</i> .....	<b>8</b>
<i>Diccionario de Datos</i> .....	<b>10</b>
<i>Referencias</i> .....	<b>15</b>



## Integrantes

- Toledo Arcos Adrian Enrique
- Apolo Balcazar Diego Gonzalo
- Campozano Bravo Frank Jaren
- Motoche Motoche David Sebastian

## Título del Proyecto

*Plataforma de gestión de academias interdisciplinarias en Unidad Educativa Particular Daniel Comboni*      **Etapa de Desarrollo: Análisis del Sistema**

## Objetivos del Negocio del cual se basa el Proyecto

*Reemplazar el antiguo sistema basado en Excel mediante la creación de un sistema de base de datos que optimice la recepción de información para los miembros de la academia (entrenadores, coordinadores).*

**Etapa de Desarrollo: Análisis del Sistema**

## Descripción General

*Llevar un registro de los estudiantes registrados en las academias del colegio (basquetbol, fútbol sala, voleibol, cheerleader, robótica y boxeo), cada una teniendo también subcategorías. Guardar los datos del estudiante, sus pagos de la pensión, entre otros datos. Todo esto con una plataforma de fácil acceso para los docentes encargados y autoridades.*

**Etapa de Desarrollo: Análisis del Sistema**

## Descripciones Funcionales

### 1. Definir los procesos especificados en el desarrollo del proyecto.

<b>Nombre del Rol</b>	<b>Descripción</b>
<b>Coordinador</b>	Es el responsable de la gestión general de las academias, incluyendo la <b>aprobación del registro</b> y la <b>supervisión de los entrenadores</b> .
<b>Entrenador/Docente</b>	Es el encargado de impartir la enseñanza en una academia específica. Puede <b>registrar y editar información de entrenamientos/clases</b> y <b>marcar la participación</b> de los estudiantes en competencias.
<b>Representante</b>	Es la persona responsable del estudiante. Puede <b>consultar las deudas pendientes</b> y las <b>cláusulas de pago</b> del estudiante.
<b>Sistema (Plataforma)</b>	El sistema automatizado de bases de datos que gestiona los registros, los pagos, y genera los reportes.

<b>Materia:</b> Sistemas de Bases de Datos 1	<b>Facultad de Ingeniería en Electricidad y Computación</b>
<b>Profesor:</b> MSc. Irene M. Cheung Ruiz	<b>ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL</b>
<b>Última Revisión:</b> Fecha	<b>Pág. 3 de 15</b>

**2. Identificar cada uno de los diferentes roles de las personas y sistemas externos involucrados en los procesos y Detallar cada uno de los roles.**

Paso	Rol Involucrado	Detalle de la Función
<b>1. Ingreso de Datos</b>	Representante / Coordinador	Se ingresa la información completa del <b>Estudiante</b> (nombre, fecha de nacimiento, promedio, curso, etc.) y la del <b>Representante</b> (nombre, teléfono, correo).
<b>2. Selección de Academia</b>	Representante / Coordinador	Se selecciona una o más <b>Academias</b> (Baloncesto, Robótica, etc.) a las que se inscribe el estudiante. El sistema asocia el ID_estudiante con el ID_academia a través de la relación de inscripción.
<b>3. Verificación de Cupo</b>	Sistema	El sistema verifica el campo n_estudiantes en la tabla <b>Academia</b> para asegurar que el cupo máximo no se haya excedido.
<b>4. Asignación Financiera</b>	Sistema	El sistema crea automáticamente un registro en la tabla <b>Pago</b> con el monto de la pensión y una Deuda inicial si corresponde.
<b>Paso</b>	<b>Rol Involucrado</b>	<b>Detalle de la Función</b>
<b>5. Aprobación y Activación</b>	Coordinador	El Coordinador revisa y aprueba el registro. El sistema activa el acceso del estudiante a la academia.

**3. Detallar cada una de las reglas de procedimiento haciendo uso del prototipo.**

**1.- Autenticación y Acceso (Regla #5):** El sistema requiere que los usuarios se identifiquen. El prototipo lo muestra con el ícono de "**Cuenta**" y el nombre del usuario logueado, "**Diego**", indicando una sesión activa.

**2.-Validación Académica (Reglas #4, #6):** El sistema verifica el promedio del estudiante (mínimo 8.5) antes de la inscripción. Esto se refleja en la columna "**Promedio**" de la tabla de Estudiantes y es una condición para la función "**Agregar estudiante**".

**3.-Validación Administrativa (Regla #3):** Secretaría debe validar la matrícula antes de que un alumno pueda ser añadido usando la función "**Agregar estudiante**".

**4.-Registro de Asistencia (Regla #9):** El docente registra la asistencia en cada sesión. El prototipo tiene una pestaña dedicada de "**Asistencia**" para esta función específica.

**5.-Control de Pagos y Deudas (Reglas #7, #10):** El sistema gestiona el estado de pensiones. La pestaña "**Deudas**" permite al docente verificar quiénes tienen pagos pendientes y aplicar la restricción de participación.

Materia: Sistemas de Bases de Datos 1	Facultad de Ingeniería en Electricidad y Computación
Profesor: MSc. Irene M. Cheung Ruiz	ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
Última Revisión: Fecha	Pág. 4 de 15



**6.-Límite de Inscripción (Regla #8):** El sistema bloquea las inscripciones después de dos meses. Esto significa que la función "**Agregar estudiante**" debe desactivarse automáticamente pasada esa fecha límite.

#### 4. Detallar las restricciones que posea el sistema.

- 1.-Restricción Académica:** Un estudiante no puede inscribirse ni permanecer en la academia si su promedio general es inferior a 8.5.
- 2.- Restricción Administrativa:** Un estudiante no puede ser agregado a una academia si no está previamente matriculado en la unidad educativa (validado por Secretaría).
- 3.-Restricción Financiera:** Un estudiante no puede participar en las actividades (y se bloquea su registro de asistencia) si acumula más de dos meses de atraso en las pensiones.
- 4.-Restricción Temporal:** La inscripción a las academias se bloquea automáticamente para los estudiantes después de los dos primeros meses del inicio de clases.
- 5.- Restricción de Acceso (Roles):** El acceso a la información está segregado por roles. Un docente (como "Diego") solo puede ver y gestionar los estudiantes, horarios y asistencias de las academias que tiene asignadas (ej. "Baloncesto"), pero no de otras (ej. "Fútbol" o "Robótica").

ACADEMIAS

Diego

UNIDAD EDUCATIVA

DANIEL COMBONI

Baloncesto

Futbol

Cheerleader

Kickboxing

Robótica

Voleibol

Contáctanos: [secretaria@uepdc.edu.ec](mailto:secretaria@uepdc.edu.ec) +593 98 765 4321



Diego


## BALONCESTO

23 estudiantes

Estudiantes	Asistencia	Horario	Deudas
<a href="#">Docente encargado</a>			
Bráulio Erick Sánchez Arellano	<a href="#">Editar</a>		
Lenin Alberto Pérez (coordinador)	<a href="#">Editar</a>		
<a href="#">Estudiantes</a>			
<a href="#">Aregar estudiante</a>			
<a href="#">Quitar estudiante</a>			
<a href="#">Actualizar datos de estudiante</a>			

Horario						
Hora de entrenamiento: 14:30 – 17:00 <a href="#">Editar</a>						
Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo

Academia	
<a href="#">Modificar mensualidad</a>	
<a href="#">Suspender academia</a>	

## DOCENTE

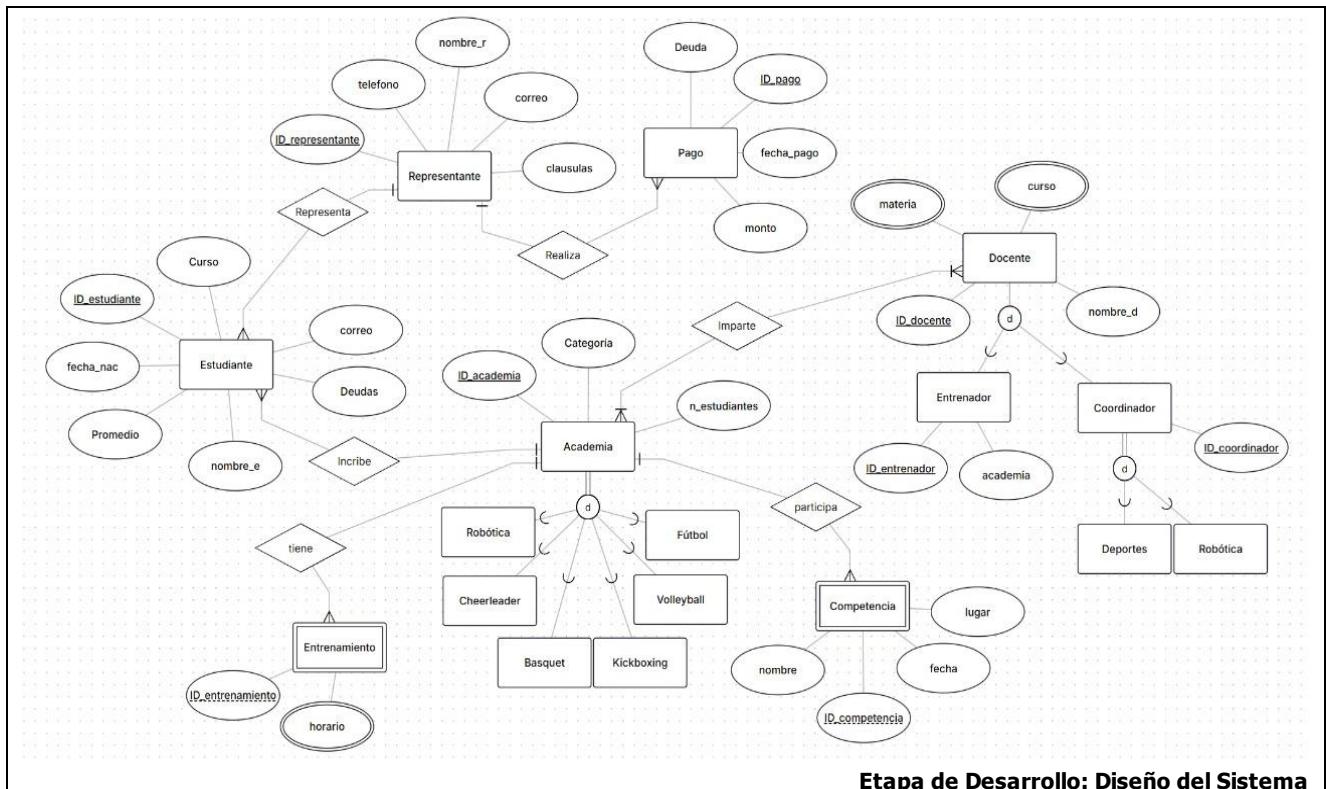
Bráulio Sánchez  
brausancheza@docente.uepdc.edu.ec  
0987654321



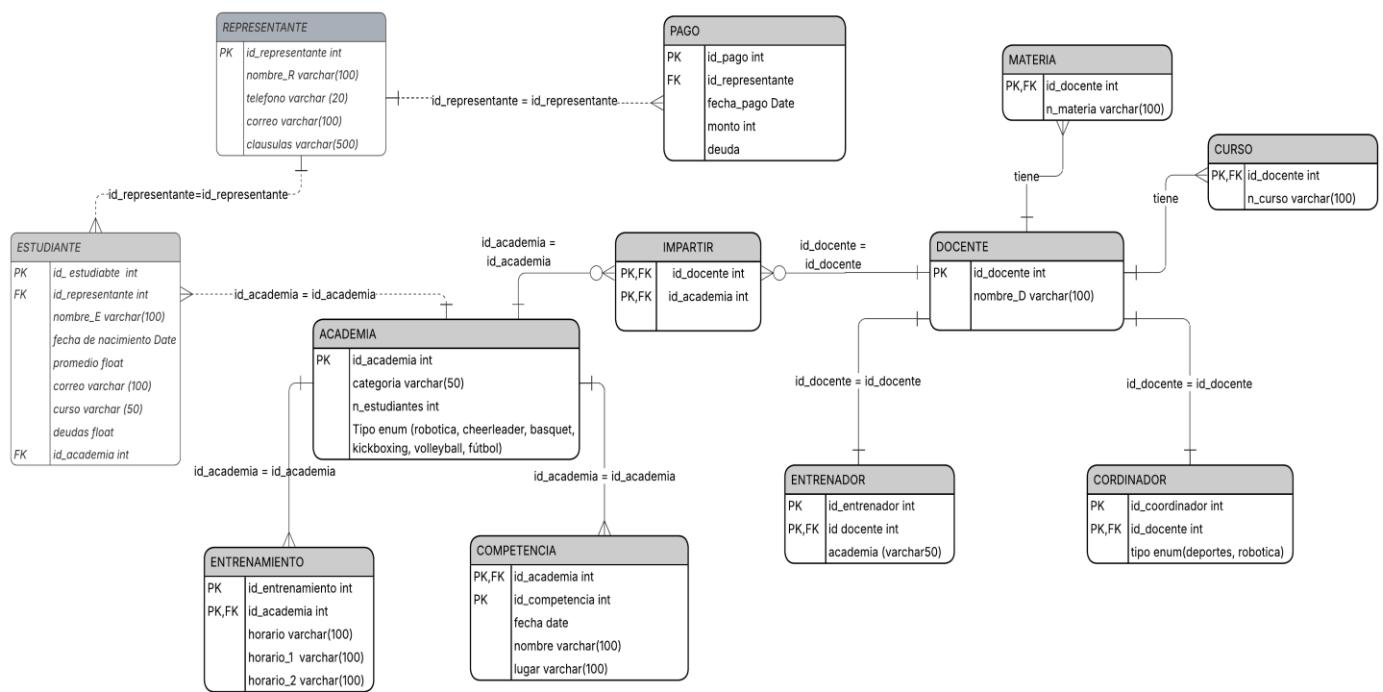
## **Etapa de Desarrollo: Diseño del Sistema**

**Materia:** Sistemas de Bases de Datos 1    **Facultad de Ingeniería en Electricidad y Computación**  
**Profesor:** MSc. Irene M. Cheung Ruiz    **ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**  
**Última Revisión:** Fecha    **Pág. 6 de 15**

## Modelo Conceptual



## Modelo Lógico



**Etapa de Desarrollo: Diseño del Sistema**

<b>Materia:</b> Sistemas de Bases de Datos 1	<b>Facultad de Ingeniería en Electricidad y Computación</b>
<b>Profesor:</b> MSC. Irene M. Cheung Ruiz	<b>ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL</b>
<b>Última Revisión:</b> Fecha	<b>Pág. 7 de 15</b>



## Reportes

### 1. Nombres de estudiantes con deudas en una academia específica.

Este reporte busca el nombre y el monto de la deuda de los estudiantes que pertenecen a una academia de categoría 'Deportiva' y que tienen deudas.

#### Álgebra Relacional:

$$\pi \text{ nombre\_E, deudas} (\sigma \text{ deudas}>0 \wedge \text{categoria}=\text{'Deportiva'} \\ (\text{ESTUDIANTE} \bowtie \text{ACADEMIA}))$$

#### Explicación:

1. **Join** : Combina ESTUDIANTE y ACADEMIA usando id\_academia.
2. **Selección**: Filtra el resultado para las filas donde deudas > 0 y categoria = 'Deportiva'.
3. **Proyección**: Muestra solo las columnas nombre y deudas.

### 2. Nombre de los docentes que también son coordinadores.

Este reporte genera una lista de los nombres de los docentes que también aparecen en la tabla COORDINADOR, ordenada alfabéticamente.

#### Álgebra Relacional:

$$\tau \text{ nombre\_D asc} (\pi \text{ nombre\_D} (\text{DOCENTE} \bowtie \text{COORDINADOR}))$$

#### Explicación:

1. **Join Natural**: Combina DOCENTE y COORDINADOR usando la columna común (id\_docente). El resultado *solo* incluirá a las personas que están en ambas tablas.
2. **Proyección**: Extrae solo la columna nombre del resultado.
3. **Ordenamiento**: Ordena la lista de nombres alfabéticamente.



### 3. Número de estudiantes por cada categoría de academia

Este reporte cuenta cuántos estudiantes hay en cada tipo de academia, agrupando los resultados por el nombre de la categoría.

#### Álgebra Relacional:

$$\gamma \text{ categoria; count(id\_estudiante)} \rightarrow n\_estudiantes (\text{ACADEMIA} \bowtie \text{ESTUDIANTE})$$

#### Explicación:

1. **Join natural:** Combina ACADEMIA y ESTUDIANTE usando id\_academia.
2. **Agrupamiento:** Agrupa las filas por categoria y cuenta (COUNT) los id\_estudiante en cada grupo.
3. **Proyección:** Muestra la categoria y el conteo (num\_estudiantes).

### 4. Estudiantes de Alto Rendimiento que Requieren Ayuda Financiera

Este reporte ayuda a identificar a los estudiantes que sobresalen académicamente (alto promedio) pero que podrían estar teniendo dificultades financieras (tienen deudas pendientes). Esto es crucial para la asignación de becas y la retención del talento superior.

#### Álgebra Relacional:

$$\pi \text{ nombre\_E, promedio, deudas, nombre\_R, telefono } (\sigma \text{ promedio} \geq 80 \wedge \text{deudas} > 0) (\text{ESTUDIANTE} \bowtie \text{REPRESENTANTE})$$

#### Explicación:

1. **Natural Join:** Combina la información de las tablas ESTUDIANTE y REPRESENTANTE usando su columna común id\_representante
2. **Selección:** Filtra las filas resultantes para incluir solo aquellos estudiantes que tienen un promedio  $\geq 80$  Y tienen deudas  $> 0$ .
3. **Proyección:** Muestra únicamente las columnas nombre\_E, promedio, deudas, nombre\_R y telefono para el reporte final.

**Etapa de Desarrollo: Diseño del Sistema**

<b>Materia:</b> Sistemas de Bases de Datos 1	<b>Facultad de Ingeniería en Electricidad y Computación</b>
<b>Profesor:</b> MSc. Irene M. Cheung Ruiz	<b>ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL</b>
<b>Última Revisión:</b> Fecha	<b>Pág. 9 de 15</b>



## Diccionario de Datos

Se debe de detallar cada una de las relaciones del modelo lógicos, siga el siguiente ejemplo.

Tabla: Academia			
Atributo	Tipo de Dato	Dominio	Descripción
ID_Academia	Int	Contiene todos los caracteres posibles	Identificador único de cada academia.
Nombre_a	VarChar(255)	Contiene todas las letras del alfabeto	Guarda el nombre de la academia.
Categoría	VarChar(30)	Contiene todos los caracteres posibles separando los numéricos de los alfabéticos por un guion	Guarda la categoría de las academias según el límite de edad
Tipo_a	Enum(robótica, cheerleader, basquet, kickboxin, volleyball, futbol)	Contiene una opción elegir la especialización	Identificar el área de especialidad de la academia

Tabla: Docente			
Atributo	Tipo de Dato	Dominio	Descripción
Id_docente	Int	Contiene todos los caracteres posibles	Identificador único de cada docente
Nombre	VarChar(255)	Contiene todos los caracteres posibles	Guarda el nombre de cada docente
Curso	VarChar(30)	Contiene todos los caracteres posibles	Guarda el curso donde imparte catedra
Materia	VarChar(255)	Contiene todos los caracteres posibles	Representa la materia impartida

Tabla: Entrenador			
Atributo	Tipo de Dato	Dominio	Descripción
Id_Entrenador	Int	Contiene todos los caracteres posibles	Identificador único de cada entrenador
Nombre_e	VarChar(255)	Contiene todas las letras del alfabeto	Guarda el nombre del entrenador
Academia	VarChar(30)	Contiene todas las letras del alfabeto	Guarda la academia donde se especializa el entrenador

Tabla: Coordinador			
Atributo	Tipo de Dato	Dominio	Descripción
Id_Coordinador	Int	Contiene todos los caracteres posibles	Identificador único para cada Coordinador
Nombre_Coord	VarChar(255)	Contiene todas las letras del alfabeto	Guarda el nombre del coordinador
Tipo_c	Enum	Contiene una opción elegir la especialización del coordinador	Identificar el área de especialización del coordinador



<b>Tabla: Entrenamiento</b>			
Atributo	Tipo de Dato	Dominio	Descripción
ID_Estramiento	Int	Contiene todos los caracteres posibles	Identificador único de cada sesión
Fecha/Hora	DateTime	Formato fecha abarca desde el periodo actual	Guarda la fecha del estudiante
<b>Tabla: Representante</b>			
Atributo	Tipo de Dato	Dominio	Descripción
Id_Representante	Int	Contiene todos los caracteres posibles	Identificador único del representante.
Nombre_R	VarChar(255)	Contiene todas las letras del alfabeto	Guarda el nombre del representante
Clausulas	VarChar(255)	Contiene todas las letras del alfabeto	Indica el tipo de clausula que falta por cumplir
Teléfono	VarChar(30)	Contiene todos los caracteres posibles	Guarda el número de telf. del representante
Correo	VarChar(255)	Contiene todos los caracteres posibles	Guarda el correo del representante.
<b>Tabla: Estudiante</b>			
Atributo	Tipo de Dato	Dominio	Descripción
Id_Estudiante	Int	Contiene todos los caracteres posibles	Identificador único de cada estudiante
Nombre_e	VarChar(255)	Contiene todas las letras del alfabeto	Guarda el nombre del estudiante
Correo	VarChar(255)	Contiene todos los caracteres posibles	Guarda la dirección de correo del estudiante
Deudas	boolean	Contiene todos los caracteres posibles	Nos da verdadero o falso dependiendo de si el representante realizó el pago
Curso	VarChar(30)	Contiene todos los caracteres posibles	Guarda el curso escolar que está cursando el estudiante
Fecha_nac	Date	Contiene todas las letras del alfabeto	Guarda la fecha de nacimiento para no tener un valor volátil como la edad



Tabla: Competencia			
Atributo	Tipo de Dato	Dominio	Descripción
Id_Competencia	Int	Contiene todos los caracteres posibles	Identificador único de cada competencia
Nombre	VarChar(255)	Contiene todas las letras del alfabeto	Guarda el nombre de la competencia
Fecha	Date	Formato de fecha que abarca el inicio y fin de la competencia	Guarda la fecha de la competencia
Lugar	VarChar(255)	Contiene todos los caracteres posibles	Guarda la ubicación de donde se realiza la competencia.

Tabla: Pago			
Atributo	Tipo de Dato	Dominio	Descripción
Id_Pago	Int	Contiene todos los caracteres posibles	Identificador único de cada pago
Deuda	Float	Contiene datos flotantes para abarcar los centavos	Guarda la deuda acumulada por el representante hasta el límite (monto*3)
Fecha_pago	Date	Formato que abarca desde 2025 hasta el fin del periodo	Guarda la fecha de la comprobación del pago
Monto	Float	Contiene datos flotantes para abarcar los centavos	Guarda el monto a depositar mensualmente.

**Etapa de Desarrollo: Diseño del Sistema**



## DESCRIPCION DE TRIGGERS

Se han implementado dos triggers en la base de datos para garantizar la integridad de los cupos en cada academia de forma automática.

- Nombre: trg\_incrementar\_cupos

Evento: Se activa DESPUÉS DE INSERTAR (AFTER INSERT) un registro en la tabla estudiante.

Función: Aumenta automáticamente en 1 el contador n\_estudiantes de la academia correspondiente al nuevo alumno inscrito.

- Nombre: trg\_decrementar\_cupos

Evento: Se activa DESPUÉS DE ELIMINAR (AFTER DELETE) un registro en la tabla estudiante.

Función: Disminuye en 1 el contador de la academia cuando un estudiante es retirado, manteniendo el número real de inscritos siempre actualizado sin intervención manual.

## DESCRIPCIÓN DE STORED PROCEDURES

Se desarrollaron procedimientos almacenados para manejar las operaciones principales (CRUD) de manera segura, implementando transacciones (COMMIT/ROLLBACK) y manejo de errores.

- Nombre: sp\_crear\_estudiante

Función: Registra un nuevo estudiante en el sistema.

Lógica: Inicia una transacción, valida que el promedio esté entre 0 y 10, e inserta los datos. Si ocurre un error, realiza un ROLLBACK automático.

- Nombre: sp\_actualizar\_estudiante

Función: Modifica los datos de un estudiante existente.

Lógica: Permite cambiar nombre, apellido, promedio o curso, asegurando la consistencia de los datos mediante transacciones.

- Nombre: sp\_eliminar\_estudiante

Función: Da de baja a un estudiante del sistema.

Lógica: Elimina el registro por su ID dentro de una transacción segura, lo que a su vez dispara el trigger de actualización de cupos.

## DESCRIPCIÓN DE INDICES

Se crearon 5 índices no agrupados (Non-Clustered) para optimizar el rendimiento de las consultas más frecuentes:

1.- idx\_estudiante\_apellido: Optimiza la búsqueda de estudiantes por apellido para la toma de asistencia y reportes, evitando el escaneo completo de la tabla.

2.- idxRepresentante\_telefono: Acelera la validación de representantes existentes al momento de la inscripción, usando el teléfono como campo de búsqueda rápida.

3.- idx\_pago\_fecha: Mejora la velocidad de los reportes financieros que se filtran por rangos de fechas (mensuales o anuales).

4.- idx\_competencia\_fecha: Permite ordenar y filtrar cronológicamente los eventos deportivos de forma eficiente.

5.- idx\_docente\_apellido: Facilita la asignación rápida de docentes en los formularios del sistema mediante búsquedas por apellido.

<b>Materia:</b> Sistemas de Bases de Datos 1	<b>Facultad de Ingeniería en Electricidad y Computación</b>
<b>Profesor:</b> MSc. Irene M. Cheung Ruiz	<b>ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL</b>
<b>Última Revisión:</b> Fecha	<b>Pág. 13 de 15</b>



## DESCRIPCIÓN DE USUARIOS Y ROLES

Se definieron 5 usuarios con permisos específicos según el principio de menor privilegio:

- Usuario: admin\_sistema (Rol: Administrador)

Permisos: ALL PRIVILEGES

Alcance: Control total sobre la base de datos.

- Usuario: secretaria (Rol: Operativo)

Permisos: SELECT, EXECUTE

Alcance: Puede ver datos y ejecutar los Procedimientos Almacenados para gestionar estudiantes.

- Usuario: rector (Rol: Gerencial)

Permisos: SELECT

Alcance: Solo tiene lectura sobre las Vistas (Reportes). No tiene acceso directo a las tablas originales (tablas crudas).

- Usuario: cajero (Rol: Financiero)

Permisos: SELECT, INSERT

Alcance: Gestión exclusiva de la tabla pago para registrar cobros.

- Usuario: auditor (Rol: Auditoría)

Permisos: SELECT

Alcance: Lectura de tablas críticas como pago y asistencia para control interno.

## Librerías de Python Necesarias

El sistema requiere las siguientes bibliotecas (instalar vía pip): - streamlit - mysql-connector-python - pandas Comando de instalación: > pip install streamlit mysql-connector-python pandas

## INSTALACIÓN Y CONFIGURACIÓN

Siga estos pasos antes de ejecutar la aplicación por primera vez:

PASO 1: Configuración de la Base de Datos 1. Abra su gestor de base de datos (MySQL Workbench).

2. Ejecute el script "BaseDeDatosComboni.sql" para crear la estructura de tablas y cargar los datos de prueba iniciales (10 registros por tabla).
3. Ejecute el script "Schema de la base de datos.sql" para instalar la lógica del negocio (Triggers, Stored Procedures, Vistas y Usuarios).

PASO 2: Configuración de la Conexión

1. Abra el archivo "CRUD.py" en un editor de texto o código.
2. Localice la función "create\_connection".
3. Verifique que la contraseña ('password') coincida con la de su instalación local de MySQL (por defecto en el código está configurada como 'admin').

<b>Materia:</b> Sistemas de Bases de Datos 1	<b>Facultad de Ingeniería en Electricidad y Computación</b>
<b>Profesor:</b> MSc. Irene M. Cheung Ruiz	<b>ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL</b>
<b>Última Revisión:</b> Fecha	<b>Pág. 14 de 15</b>



## GUÍA DE USO - MÓDULOS DEL SISTEMA

El sistema cuenta con un menú lateral de navegación con dos opciones principales:

### A. MÓDULO DE REPORTES

- Función: Visualización de datos consolidados.
- Uso: Seleccione "Reportes" en el menú.
- Resultado: Se mostrará la "Ficha de Estudiantes", una vista que integra información del alumno, su academia asignada y datos de contacto del representante. Esta información es de solo lectura.

### B. MÓDULO DE GESTIÓN DE ESTUDIANTES

Este módulo permite administrar la información mediante tres acciones:

#### 1. Registrar (Alta de Estudiante):

- Seleccione la opción "Registrar".
- Complete el formulario (Nombre, Apellido, Correo, Fecha Nacimiento, ID Representante, ID Academia, Promedio, Curso). - Haga clic en "Guardar".
- Nota: El sistema validará internamente que el promedio sea válido (0-10) y actualizará automáticamente el cupo de la academia (Trigger).

#### 2. Actualizar (Modificación de Datos):

Seleccione la opción "Actualizar".

- Ingrese el ID del estudiante a modificar.
- Llene los campos con la nueva información.
- Haga clic en "Actualizar" para guardar los cambios mediante SP.

#### 3. Eliminar (Baja de Estudiante):

- Seleccione la opción "Eliminar".

- Ingrese el ID del estudiante. - Haga clic en "Eliminar".

- Advertencia: Esta acción es irreversible y ajustará automáticamente el contador de estudiantes de la academia.

## SOLUCIÓN DE PROBLEMAS FRECUENTES

-ERROR: "Access denied for user 'root'@'localhost'"

SOLUCIÓN: La contraseña en el archivo CRUD.py no coincide con la de su MySQL.

Edite el archivo y coloque la contraseña correcta.

- ERROR: "Can't connect to MySQL server" SOLUCIÓN: Asegúrese de que el servicio de MySQL (XAMPP o Workbench) esté iniciado ("Running").

- ERROR: Promedio inválido SOLUCIÓN: El sistema rechaza promedios menores a 0 o mayores a 10. Verifique el dato ingresado.

## Referencias

Nuestras referencias para el desarrollo de los avances, nacieron a partir de entrevistas realizadas al personal participante dentro de las academias (Administradora general y entrenador):

<https://drive.google.com/drive/folders/1F56U5APUfi4eZJ31HKmMGrdr8hjO40aq>

<b>Materia:</b> Sistemas de Bases de Datos 1	<b>Facultad de Ingeniería en Electricidad y Computación</b>
<b>Profesor:</b> MSc. Irene M. Cheung Ruiz	<b>ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL</b>
<b>Última Revisión:</b> Fecha	<b>Pág. 15 de 15</b>