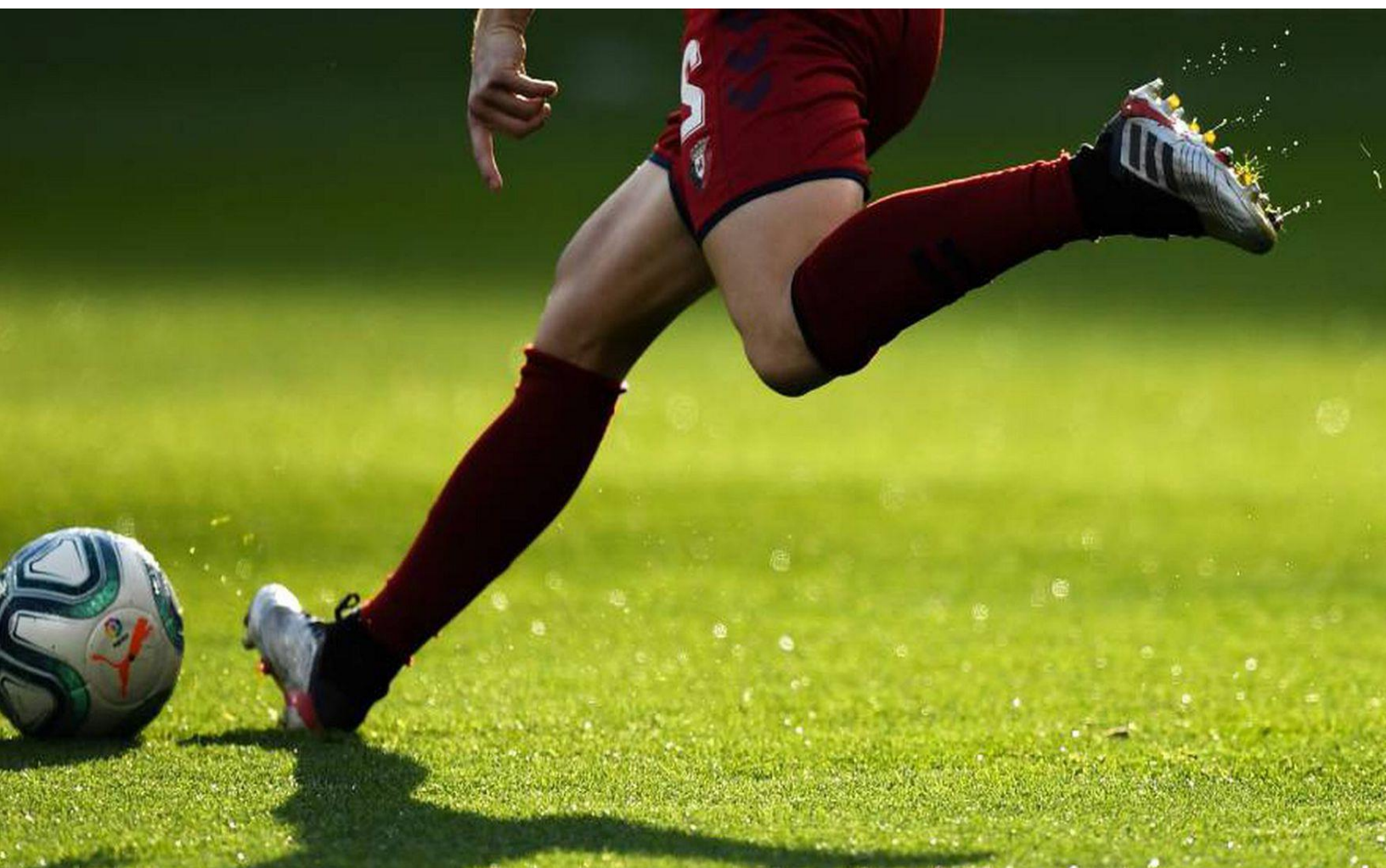


# ACADEMIAS DEPORTIVAS



**NOMBRE:** MARIA JOSE CERVANTES RUIZ

**ENTREGA N°1**

## PROBLEMATIA

El **proyecto de gestión de academias deportivas** tiene como objetivo principal crear un sistema eficiente y automatizado para administrar y controlar las operaciones diarias de un club deportivo. Este sistema debe permitir:

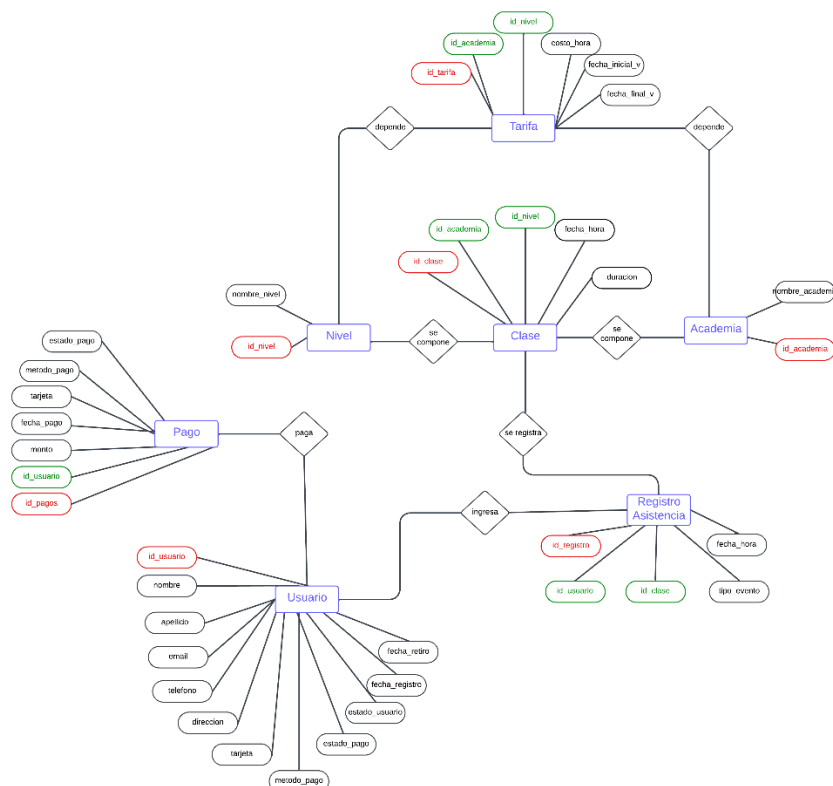
1. **Gestionar las inscripciones de usuarios** a diversas academias deportivas categorizadas por niveles de habilidad, de manera clara y organizada.
2. **Registrar y controlar la asistencia** de los usuarios a las clases programadas, asegurando un seguimiento adecuado de las horas de participación para calcular el monto de cobro correspondiente.
3. **Gestionar el cobro de tarifas** en función del deporte, nivel de habilidad y las horas de clase tomadas, asegurando que el pago sea procesado de manera eficiente y automática, utilizando métodos de pago como tarjetas de crédito o débito.
4. **Monitorear el estado de pago** de cada usuario, restringiendo el acceso a las clases si no se ha realizado el pago correspondiente, y permitiendo la generación de reportes mensuales para facilitar el seguimiento tanto para los administradores como para los usuarios.

## DESCRIPCIÓN DE LA BASE DE DATOS

Las academias que ofrece el club incluyen deportes como fútbol, tenis y natación. A continuación, se detallan los aspectos clave de la estructura y funcionalidad de la base de datos:

1. **Academias y Niveles:** El club deportivo ofrece tres academias principales: fútbol, tenis y natación. Cada academia tiene clases con horarios fijos establecidos a lo largo del día. Además, las academias están organizadas por niveles de habilidad: junior, semi-senior y senior. Los usuarios pueden suscribirse a una academia según el deporte y el nivel en el que desean participar.
2. **Registro de Clases y Control de Asistencia:** Cada clase de la academia tiene un horario específico y, para controlar la asistencia, la base de datos registra el **ingreso** y la **salida** de los usuarios a las clases. Estos registros son esenciales para llevar un control del tiempo de participación de cada usuario y calcular el cobro correspondiente según las horas de clase tomadas.
3. **Cobro por Hora y Nivel:** El cobro por la participación en las clases se realiza según el tipo de academia (fútbol, tenis o natación) y el nivel de habilidad (junior, semi-senior o senior). Cada tipo de academia y nivel tiene un costo diferente por hora de clase. La base de datos debe calcular el monto a cobrar al usuario de acuerdo con la cantidad de horas de clase que haya tomado en el mes, y este cobro se realiza de manera mensual.

- ### DIAGRAMA ENTIDAD-RELACIÓN:



## TABLAS DEL MODELO Y SU DESCRIPCIÓN

**Tabla 1: Usuario**

Campo	Tipo de Dato	Descripción
id_usuario	INT (PK)	Identificador único del usuario.
nombre	VARCHAR(100)	Nombre del usuario.
apellido	VARCHAR(100)	Apellido del usuario.
email	VARCHAR(255)	Correo electrónico del usuario.
telefono	VARCHAR(20)	Teléfono de contacto.
direccion	TEXT	Dirección del usuario.
tarjeta	VARCHAR(20)	Número de tarjeta de crédito o débito.
metodo_pago	VARCHAR(50)	Método de pago (tarjeta de credito o debito).
estado_pago	ENUM('pendiente', 'pagado')	Estado del pago (pendiente si no se ha realizado, pagado si el pago fue procesado con éxito).
estado_usuario	ENUM('activo', 'inactivo_por_mora', 'inactivo_por_retiro')	Estado de la suscripción del usuario.
fecha_registro	DATE	Fecha de registro del usuario.
fecha_retiro	DATE	Fecha en que el usuario solicitó su retiro, si aplica.

**Tabla 2: Academia**

Contiene las academias deportivas.

Campo	Tipo de Dato	Descripción
id_academia	INT (PK)	Identificador único de la academia.
nombre_academia	VARCHAR(100)	Nombre de la academia (fútbol, tenis, natación).

**Tabla 3: Nivel**

Define los niveles de habilidad de las academias.

Campo	Tipo de Dato	Descripción
id_nivel	INT (PK)	Identificador único del nivel.
nombre_nivel	VARCHAR(50)	Nombre del nivel (junior, semi-senior, senior).

**Tabla 4: Clase**

Cada clase está asociada a una academia y un nivel, y tiene un horario específico.

Campo	Tipo de Dato	Descripción
id_clase	INT (PK)	Identificador único de la clase.
id_academia	INT (FK)	Relación con la academia (referencia a Academias).
id_nivel	INT (FK)	Relación con el nivel (referencia a Niveles).
fecha_hora	DATETIME	Fecha y hora de la clase.
duracion	INT	Duración de la clase en minutos.

**Tabla 5: Registro\_asistencia (Ingreso y Salida)**

Esta tabla registrará tanto los **ingresos** como las **salidas** de los usuarios a las clases. Un usuario puede ingresar a una clase y luego salir, y esto se registrará con las horas correspondientes.

Campo	Tipo de Dato	Descripción
id_registro	INT (PK)	Identificador único del registro.
id_usuario	INT (FK)	Relación con el usuario (referencia a Usuarios).
id_clase	INT (FK)	Relación con la clase (referencia a Clases).
tipo_evento	ENUM('ingreso', 'salida')	Tipo de evento (ingreso o salida).
fecha_hora	DATETIME	Fecha y hora del evento.

**Tabla 6: Tarifa**

Campo	Tipo de Dato	Descripción
id_tarifa	INT (PK)	Identificador único del registro
id_academia	INT (FK)	Relación con la academia.
id_nivel	INT (FK)	Relación con el nivel.
costo_hora	DECIMAL(10,2)	Costo por hora para esa combinación de academia y nivel.
fecha_inicial_v	DATE	Fecha de vigencia inicial de la tarifa
fecha_final_v	DATE	Fecha de vigencia final de la tarifa

**Tabla 7: Pago**

Esta tabla almacena los pagos realizados por los usuarios por su participación en las clases.

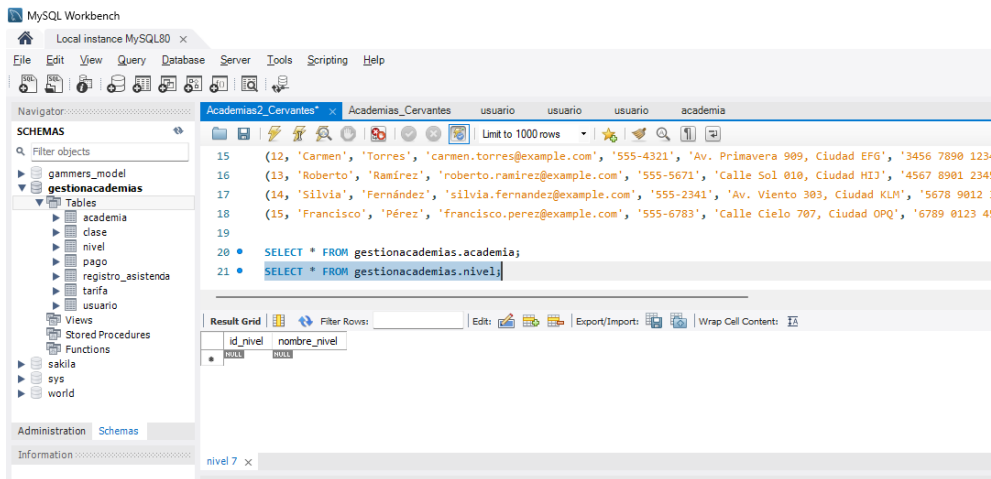
Campo	Tipo de Dato	Descripción
id_pago	INT (PK)	Identificador único del pago.
id_usuario	INT (FK)	Relación con el usuario (referencia a Usuarios).
Monto	DECIMAL(10,2)	Monto total a pagar.
fecha_pago	DATE	Fecha de realización del pago.
Tarjeta	VARCHAR(20)	Número de tarjeta de crédito o débito.
metodo_pago	VARCHAR(50)	Método de pago (tarjeta de crédito o débito).
estado_pago	ENUM('pagado', 'pendiente')	Estado del pago (pendiente si no se ha realizado, pagado si el pago fue procesado con éxito).

## INSERCIÓN DE LOS DATOS

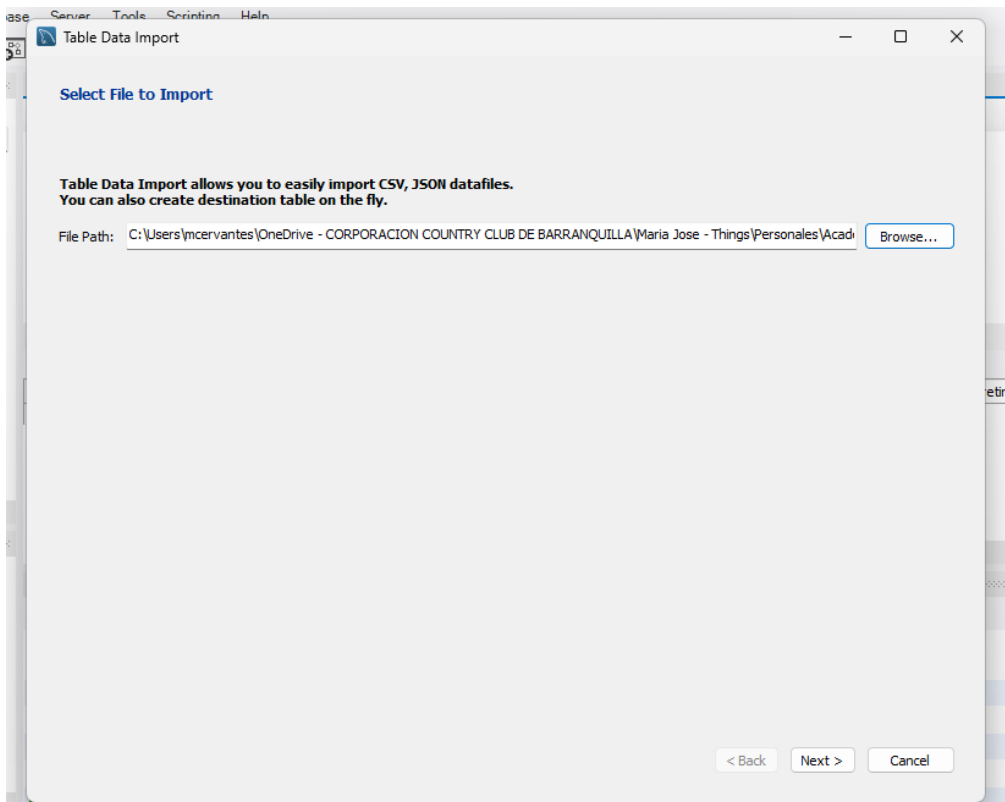
Para la inserción de datos en el sistema de gestión de academias deportivas, se utilizaron dos métodos: la importación de archivos externos y la inserción manual. Las tablas academia y nivel fueron llenadas mediante archivos externos para practicar la carga masiva de datos, lo que permitió una integración eficiente de la información. En cambio, las tablas relacionadas con usuario, pago, clase, tarifa y registro asistencia se insertaron manualmente, con el objetivo de practicar el manejo directo de datos. Este enfoque mixto optimizó el tiempo de implementación y proporcionó experiencia en ambos métodos de inserción.

## INSERCIÓN DE LOS DATOS DESDE ARCHIVO EXTERNO (ACADEMIA – NIVEL)

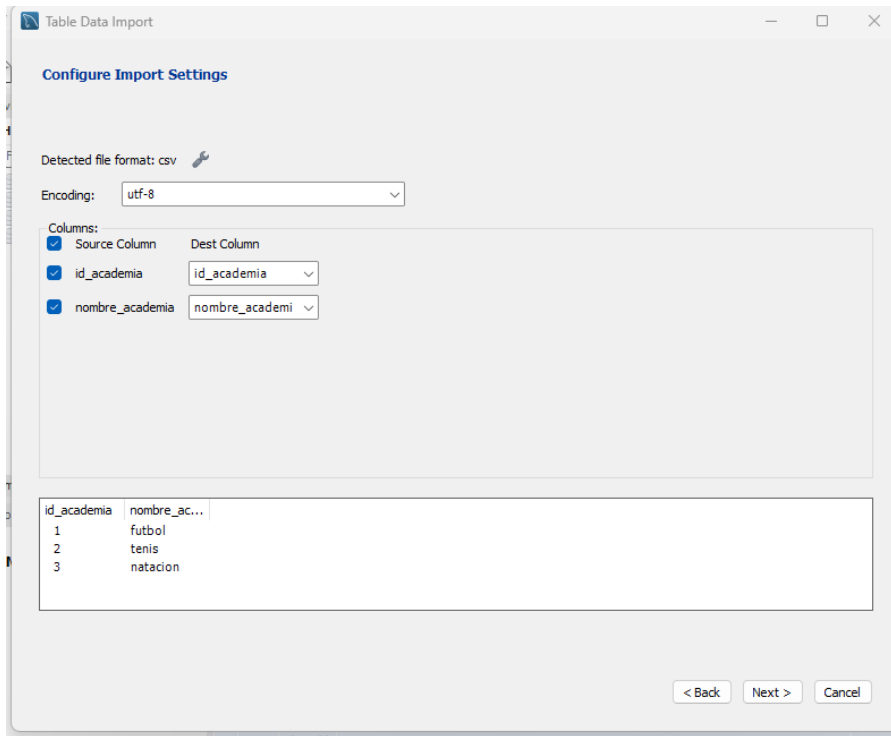
- Consultar las tablas



- Buscar los archivos en la ubicación



- Relacionar los archivos con las tablas

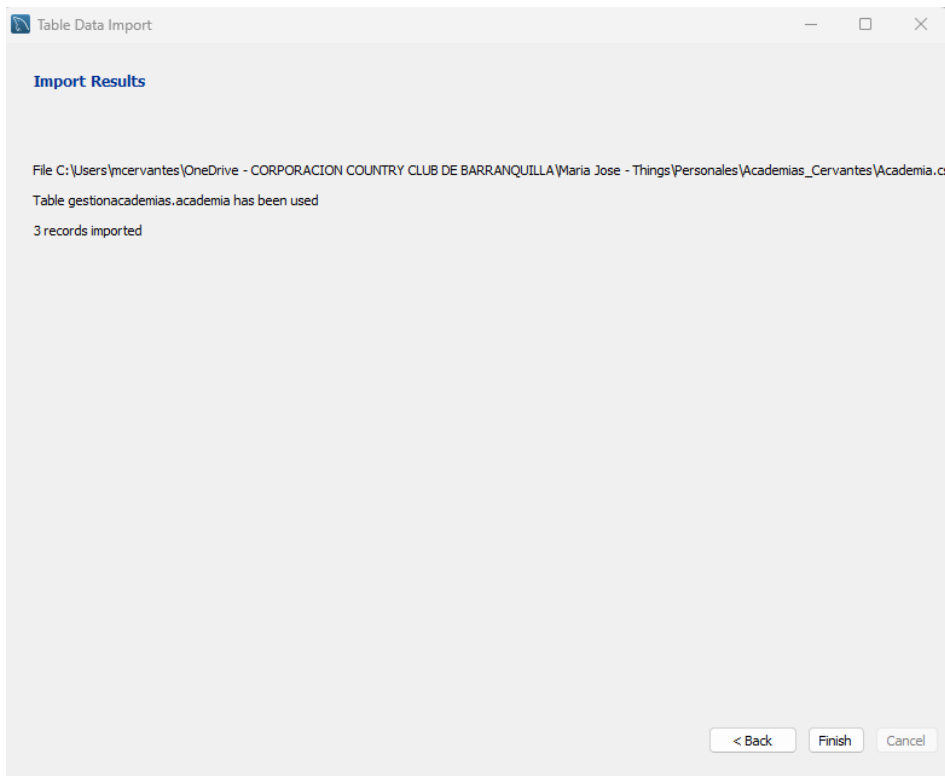


The dialog box is titled "Table Data Import" and "Configure Import Settings". It shows the detected file format as "csv" and the encoding as "utf-8". Under the "Columns" section, there are two columns: "Source Column" and "Dest Column". The "Source Column" has two entries: "id\_academia" and "nombre\_academia". The "Dest Column" has two entries: "id\_academia" and "nombre\_academi". Below this, there is a preview table with three rows of data.

id_academia	nombre_ac...
1	futbol
2	tenis
3	natacion

At the bottom right, there are buttons for "< Back", "Next >", and "Cancel".

- Importar los datos



The dialog box is titled "Table Data Import" and "Import Results". It shows the file path: "File C:\Users\ncervantes\OneDrive - CORPORACION COUNTRY CLUB DE BARRANQUILLA\Maria Jose - Things\Personales\Academias\_Cervantes\Academia.cs". It also shows the table name: "Table gestionacademias.academia has been used". Below this, it says "3 records imported". At the bottom right, there are buttons for "< Back", "Finish", and "Cancel".



## VISTAS

### 1. Vista\_usu\_activos\_pago

Esta vista proporciona información detallada de los usuarios que se encuentran en estado "inactivo por mora". Muestra el nombre completo del usuario, su correo electrónico, el estado del usuario (activo o inactivo) y el estado de su pago, además de su último pago realizado.

Facilita la identificación de los usuarios que no han pagado y se encuentran en "inactivo por mora". Este tipo de información es crucial para los administradores del sistema, ya que les permite gestionar de forma efectiva las cuentas pendientes de pago, tomar decisiones sobre el acceso a las clases, y comunicar a los usuarios sobre la necesidad de regularizar su situación. Además, permite el seguimiento de la situación financiera del usuario de manera eficiente.

### 2. Vista\_clases\_progr:

Muestra las clases programadas para la academia, filtradas por fecha futura, detallando el nombre de la academia, el nivel de las clases, la fecha y hora de cada clase, así como su duración.

Esta vista permite tener una visión clara y organizada de todas las clases programadas, facilitando la planificación y gestión de las actividades. Para los usuarios, proporciona una forma sencilla de consultar las clases disponibles en el futuro.

### 3. Vista\_registro\_asist:

Proporciona información sobre los registros de asistencia de los usuarios, específicamente aquellos que han registrado su salida. Incluye detalles como el nombre del usuario, el nombre de la clase, la fecha y hora de la clase, y la fecha y hora del evento de salida.

Permite realizar un seguimiento adecuado de la asistencia de los usuarios a las clases, lo que es fundamental para calcular las horas de participación. Esto facilita la administración de las clases y proporciona información que puede ser útil para calcular tarifas y verificar la asistencia real del usuario.

## FUNCIONES

### 1. FN\_CLASES\_FECHA:

Esta función devuelve el número total de clases que se han programado para una academia y un nivel específicos dentro de un rango de fechas.

Permite a los administradores obtener rápidamente un conteo de las clases programadas, lo cual es útil para analizar la carga de trabajo y planificar recursos.

## **2. FN\_CANT\_PAGOS\_EST:**

Esta función devuelve la cantidad total de pagos realizados que tienen un estado específico, ya sea "pendiente" o "pagado", dentro de un rango de fechas.

Facilita el monitoreo del estado de los pagos de los usuarios, permitiendo a los administradores ver cuántos pagos se han realizado o están pendientes durante un periodo determinado.

## **PROCEDIMIENTOS**

### **1. SP\_EST\_P\_EST\_U:**

Este procedimiento recupera el estado de pago y el estado del usuario en función de su nombre, apellido y correo electrónico.

Este procedimiento permite obtener de manera rápida el estado de pago y la situación del usuario, lo que es muy útil para los administradores al momento de gestionar el acceso a las clases. Si un usuario tiene un estado de pago "pendiente", el sistema puede restringir su acceso a futuras clases o generar alertas. Esto asegura una correcta gestión administrativa y evita que los usuarios que no han pagado accedan a las clases sin haber regularizado su situación.

### **2. SP\_TARIFAS:**

Este procedimiento devuelve la tarifa correspondiente para una combinación de academia y nivel, junto con las fechas de inicio y finalización de la tarifa.

Permite a los administradores obtener rápidamente la tarifa actual vigente para un determinado nivel y academia, lo cual es esencial para calcular los pagos de los usuarios. De esta manera, se asegura que los cobros sean realizados con la tarifa correcta, tomando en cuenta las posibles variaciones de precios en diferentes periodos. Además, al acceder a la tarifa más reciente, se evita el cobro incorrecto a los usuarios.

## **TRIGGERS**

### **1. trg2\_log\_usuario:**

Este trigger registra los cambios de estado de pago realizados en la tabla usuario. Cuando un usuario actualiza su estado de pago, se crea un registro en la tabla log2\_usuario, proporcionando información detallada sobre la modificación.

Ayuda a llevar un historial detallado de los cambios realizados en el estado de pago de los usuarios. Esto es crucial para la trazabilidad y el control de las modificaciones en los datos.

## **2. trg2\_log\_usuario:**

Este trigger registra los cambios de estado de pago realizados en la tabla usuario. Cuando un usuario actualiza su estado de pago, se crea un registro en la tabla log2\_usuario, proporcionando información detallada sobre la modificación.

Ayuda a llevar un historial detallado de los cambios realizados en el estado de pago de los usuarios. Esto es crucial para la trazabilidad y el control de las modificaciones en los datos

**En general este sistema permite tener un control centralizado y eficiente sobre las operaciones diarias de la academia deportiva, asegurando que los usuarios reciban el servicio adecuado mientras los administradores pueden gestionar las clases y pagos de manera eficiente.**

## **SCRIPT**

[https://github.com/majo1234cr/Academias\\_Cervantes.git](https://github.com/majo1234cr/Academias_Cervantes.git)