

# **PRACTICO 1B**

## **EJERCICIO TRANSVERSAL INTEGRADOR:**

**INTEGRANTES:** Fuentes Farfan Genaro, Pereyra  
Jorge Pedro Ezequiel y Fernandez Fedro  
**SISTEMA DE GESTION ESTUDIANTEL**

¿Cual es el problema a resolver?

-El problema es la gestion de la informacion academica de los estudiantes.

¿En que contexto se situa el problema?

-Se situa en la Universidad.

¿Quienes son los usuarios/actores que van a interactuar en el futuro con el sistema que desarrollamos?

-El personal de la Oficina de Alumnos, como administradores, los profesores y los estudiantes.

¿Para que lo van a usar?

-Los alumnos para consultar informacion academica, las correlatividades si es que las tienen, como va su progreso, de los profes para cargar notas o consultar los listados de sus alumnos y los administradores van a estar en la oficina de alumno recaudando la información academica de los alumnos y agendandolos, para luego sea mas facil pasar el listado de alumnos a los profesores.

```

    erDiagram
        Persona ||--o{ Alumno : "es"
        Persona ||--o{ Usuario : "es"
        Alumno ||--o{ Carrera : "tiene"
        Alumno ||--o{ Cursada : "tiene"
        Alumno ||--o{ Nota : "tiene"
        Alumno ||--o{ Materia : "tiene"
        Alumno ||--o{ Plan_De_Estudio : "tiene"
        Usuario ||--o{ Profesor : "es"
        Profesor ||--o{ Carrera : "tiene"
        Profesor ||--o{ Cursada : "tiene"
        Profesor ||--o{ Nota : "tiene"
        Profesor ||--o{ Materia : "tiene"
        Profesor ||--o{ Condicion : "tiene"
        Carrera ||--o{ Cursada : "tiene"
        Carrera ||--o{ Nota : "tiene"
        Carrera ||--o{ Materia : "tiene"
        Cursada ||--o{ Nota : "tiene"
        Cursada ||--o{ Materia : "tiene"
        Nota ||--o{ Materia : "tiene"
        Plan_De_Estudio ||--o{ Materia : "tiene"
        Condicion ||--o{ Materia : "tiene"

        Persona {
            string DNI
            string Nombre
            string Apellido
            string Telefono
            string Direccion
            string Email
        }
        Alumno {
            string Tipo_Alumno
        }
        Usuario {
            string Contraseña
            string DNI_Usuario
        }
        Carrera {
            string Nombre
            string Codigo
        }
        Cursada {
            string Año_Cursada
            string Tipo_Progreso
        }
        Nota {
            float Valor
            string Fecha
        }
        Materia {
            string Nombre_Materia
            string Codigo
            string Carga_Horaria
            string Tipo_Periodo
        }
        Plan_De_Estudio {
            string Año
            string Cantidad_Materias
        }
        Condicion {
            string Tipo_Condicion
        }
  
```

El diagrama de Entidad-Relación (ER) para el sistema de gestión de una institución educativa se compone de las siguientes entidades, atributos y relaciones:

- Entidad Persona:**
  - Atributos: +DNI, +Nombre, +Apellido, +Telefono, +Direccion, +Email.
- Entidad Usuario:**
  - Atributos: +Contraseña, +DNI\_Usuario.
- Entidad Alumno:**
  - Atributo: +Tipo\_Alumno.
- Entidad Profesor:**
  - Atributo: +Tipo\_Cargo.
- Entidad Carrera:**
  - Atributos: +Nombre, +Codigo.
- Entidad Cursada:**
  - Atributos: +Año\_Cursada, +Tipo\_Progreso.
- Entidad Nota:**
  - Atributos: +Valor, +Fecha.
- Entidad Materia:**
  - Atributos: +Nombre\_Materia, +Codigo, +Carga\_Horaria, +Tipo\_Periodo.
- Entidad Plan\_De\_Estudio:**
  - Atributos: +Año, +Cantidad\_Materias.
- Entidad Condicion:**
  - Atributo: +Tipo\_Condicion.

Las relaciones entre las entidades se definen con los siguientes cardinalidades:

- Persona y Alumno:** 1 a +1 (Alumno es un tipo de Persona).
- Persona y Usuario:** 1 a +1 (Usuario es un tipo de Persona).
- Alumno y Carrera:** 1 a +1 (Alumno pertenece a una Carrera).
- Alumno y Cursada:** 1 a +1 (Alumno cursa una Cursada).
- Alumno y Nota:** 1 a +1 (Alumno tiene una Nota).
- Alumno y Materia:** 1 a +1 (Alumno tiene una Materia).
- Alumno y Plan\_De\_Estudio:** 1 a +1 (Alumno tiene un Plan\_De\_Estudio).
- Usuario y Profesor:** 1 a +1 (Profesor es un tipo de Usuario).
- Profesor y Carrera:** 1 a +1 (Profesor pertenece a una Carrera).
- Profesor y Cursada:** 1 a +1 (Profesor cursa una Cursada).
- Profesor y Nota:** 1 a +1 (Profesor tiene una Nota).
- Profesor y Materia:** 1 a +1 (Profesor tiene una Materia).
- Profesor y Condicion:** 1 a +1 (Profesor tiene una Condicion).
- Carrera y Cursada:** 1 a +1 (Carrera tiene una Cursada).
- Carrera y Nota:** 1 a +1 (Carrera tiene una Nota).
- Carrera y Materia:** 1 a +1 (Carrera tiene una Materia).
- Cursada y Nota:** 1 a +1 (Cursada tiene una Nota).
- Cursada y Materia:** 1 a +1 (Cursada tiene una Materia).
- Nota y Materia:** 1 a +1 (Nota tiene una Materia).
- Plan\_De\_Estudio y Materia:** 1 a +1 (Plan\_De\_Estudio tiene una Materia).
- Condicion y Materia:** 1 a +1 (Condicion tiene una Materia).

Además, se definen enumeraciones para los tipos de Alumno, Profesor, Nota, Condicion y Tipo\_Periodo, así como para el Tipo\_Progreso de la Cursada.

**Gestión de estudiantes:**  
Registrar datos personales y de contacto.

Identificar si es ingresante o avanzado.

Consultar historial académico del estudiante.

### Gestión de docentes:

Registrar información de los docentes.

Asignar docentes a materias en un período académico.

Definir rol en la cátedra: responsable, JTP, ayudante.

### Gestión de materias y carreras:

Registrar materias con su plan de estudios.

Definir correlatividades entre materias.

Gestionar oferta académica por carrera y período

### Gestión de cursadas e inscripciones:

Permitir que un estudiante se inscriba a materias.

Validar automáticamente si cumple correlatividades antes de la inscripción.

Registrar qué materias está cursando cada estudiante.

### Gestión de notas y avance:

Registrar notas finales de cada materia cursada.

Consultar el avance del estudiante en el plan de estudios.

Soporte a usuarios específicos

Administradores (Oficina de alumnos): alta, baja y modificación de estudiantes, docentes, materias, correlativas.

Docentes: cargar notas, consultar lista de estudiantes.

Estudiantes: consultar su información académica, materias disponibles y correlatividades cumplidas.

### Seguimiento de estudiantes (futuro):

Analizar progreso de estudiantes.

Detectar riesgo de abandono.

Identificar estudiantes destacados para programas especiales.

### Requerimientos No Funcionales:

#### Usabilidad:

Interfaz intuitiva para que personal administrativo, docentes y estudiantes puedan usarlo sin dificultad.

#### Confiabilidad:

Validación automática de correlatividades para evitar inscripciones inválidas.

Registro seguro de notas y datos académicos.

#### Seguridad:

Autenticación diferenciada según el rol (administrador, docente, estudiante).

Permisos de acceso restringidos (ej: solo administradores pueden modificar datos).

#### Escalabilidad:

Capacidad de soportar crecimiento en cantidad de estudiantes, docentes y carreras.

#### Integración:

Posibilidad futura de integrarse con herramientas de análisis académico (para detectar riesgo de abandono o buen desempeño).

#### Disponibilidad:

El sistema debe estar disponible en horario académico y preferentemente accesible vía web para estudiantes y docentes.

### Mantenibilidad:

Diseño modular que facilite futuras mejoras como el seguimiento de estudiantes.

### Backlog:

♥Alta Prioridad (imprescindibles para el sistema)

Registrar y gestionar estudiantes (datos personales, contacto, estado ingresante/avanzado). → Fundamental para cualquier otra operación.

Registrar materias y planes de estudio. → Base de la oferta académica.

Definir y gestionar correlatividades. → Clave para la inscripción automática y evitar errores.

Inscribir estudiantes en materias validando correlatividades. → Función central que afecta directamente a los alumnos.

---

★ Prioridad Media (necesarios para operar bien el sistema)

Registrar y gestionar docentes (datos personales, materias que dictan, roles en cátedra). → Importante para organización académica.

Asignar docentes a materias por período académico con roles (responsable, JTP, ayudante). → Permite estructurar equipos docentes y mejorar administración.

Registrar notas finales de aprobación. → Necesario para consolidar el historial académico de los alumnos.

---

- Prioridad Baja (valor agregado, pero no crítico en una primera versión)

Consultar información académica de estudiantes (materias cursadas, aprobadas, estado de avance). → Muy útil para estudiantes, pero depende de que las notas y planes ya estén cargados.

Generar reportes de progreso estudiantil (avance, riesgo de abandono, buen desempeño). → Objetivo a futuro, pensado para analítica y gestión avanzada.

HU(Historias de Usuario): lista priorizada

### Historias de Usuario (HU):

<i>HU-ADM-001</i>
<i>Registro de estudiantes.</i>
<i>Como administrador, quiero registrar y consultar los datos personales y académicos de los estudiantes, para tener un historial centralizado y actualizado.</i>

<i>HU-ALUM-001</i>
<i>Consulta de información personal y académica.</i>
<i>Como estudiante, quiero consultar mis datos académicos y materias cursadas, para seguir mi progreso en la carrera.</i>

<i>HU-ALUM-002</i>
<i>Validación de correlatividades en inscripción.</i>
<i>Como estudiante, quiero que el sistema valide automáticamente las correlatividades antes de inscribirme en una materia, para evitar errores de inscripción.</i>

<i>HU-ALUM-003</i>
<i>Inscripción en materias.</i>

*Como estudiante, quiero inscribirme en las materias que me correspondan, para avanzar en mi carrera de acuerdo al plan de estudios.*

*HU-DOC -001*

*Registro de notas.*

*Como docente, quiero cargar y actualizar las notas finales de los estudiantes, para mantener actualizado el rendimiento académico en el sistema.*

*HU-DOC -002*

*Consulta de listado de alumnos*

*Como profesor, quiero consultar el listado de estudiantes inscriptos en mis materias, para organizar las actividades de clase.*

*HU-ADM-001*

*Asignación de docentes a materias.*

*Como administrador, quiero asignar profesores a materias con su rol, para organizar los equipos docentes.*

*HU-ADM-002*

*Registro de materias y planes de estudio.*

*Como administrador, quiero cargar las materias con su plan de estudio y correlatividades, para facilitar la inscripción automática de los estudiantes.*

*HU-ADM-003*

*Seguimiento de progreso estudiantil.*

*Como administrador, quiero generar reportes sobre el avance de los estudiantes y detectar riesgos de abandono, para apoyar la gestión académica y programas de retención.*

<i>HU-ALUM-004</i>
<i>Consulta de materias cursadas</i>
<i>Como estudiante, quiero consultar las materias que estoy cursando y las que ya aprobé, para seguir mi progreso en la carrera.</i>

<i>ID de HU:</i>	<i>HU-ALUM-003</i>
<i>Título:</i>	<i>Inscripción en materias.</i>
<i>Declaración:</i>	<i>Como estudiante, quiero inscribirme en las materias que me correspondan, para avanzar en mi carrera de acuerdo al plan de estudios.</i>
<i>Descripción Detallada:</i>	<i>El Sistema debe permitir que los estudiantes se inscriban a materias de forma segura y eficiente, teniendo en cuenta su historial académico y el cumplimiento de correlativas. Identificando el perfil del estudiante si es ingresante (sin materias aprobadas) o avanzado (con materias cursadas y aprobadas). Antes de habilitar la inscripción, el sistema debe verificar si el estudiante cumple con los requisitos y correlativas definidos en el plan de estudios, solo debe</i>



	mostrar aquellas materias para las que el estudiante está habilitado, evitando errores administrativos o inscripciones inválidas, si una materia no está habilitada, el sistema debe informar el motivo (por ejemplo, “Falta aprobar Programación I”). Una vez finalizada la inscripción debe quedar registrado el estado (rechazado, pendiente, confirmado).
Criterios de Validación:	<p>El sistema debe mostrar únicamente las materias disponibles para el estudiante, de acuerdo a su plan de estudios y correlativas aprobadas.</p> <p>El sistema debe identificar automáticamente si el estudiante es ingresante (sin materias aprobadas) o avanzado (con materias aprobadas) y ajustar la oferta de materias según corresponda.</p> <p>La confirmación de inscripción debe estar disponible para consulta posterior por parte del estudiante.</p>
Tareas Asociadas a la Implementación:	Implementar lógica de correlativas.

ID de HU:	HU-ADM-001
Título:	Asignación de profesores a materias.
Declaración:	Como administrativo, quiero asignar profesores a materias

	con su rol, para organizar los equipos docentes.
Descripción Detallada:	<p>El sistema debe permitir al personal administrativo (oficina de alumnos) asignar docentes a las materias para un determinado período académico (año y/o cuatrimestre). Uno o varios profesores pueden ser asignados a una o varias materias especificando el rol del docente en esa asignatura: responsable de cátedra (solo puede haber un responsable de cátedra por materia), jefe de trabajos prácticos, ayudante.</p> <p>Cada docente a la hora de ser asignado a una materia debe incluir: período académico y carga horaria asignada.</p>
Criterios de Validación:	<p>Se debe especificar obligatoriamente el rol del docente.</p> <p>El sistema debe impedir la existencia de más de un responsable de cátedra.</p> <p>Validar que el profesor no tenga conflictos de horarios.</p>
Tareas Asociadas a la Implementación:	Implementar validación de roles (solo un Responsable de cátedra por comisión).

ID de HU:	HU-DOC -001
Título:	Registro de notas.
Declaración:	Como docente, quiero cargar y actualizar las notas finales de los estudiantes, para mantener actualizado el rendimiento académico en el sistema.
Descripción Detallada:	El sistema debe permitir que los docentes registren y actualicen las calificaciones finales de los estudiantes en cada materia. Las notas deben asociarse a una cursada específica y a un período académico (año/cuatrimestre). El sistema debe validar que el docente esté realmente asignado a la materia antes de habilitar la carga de notas. Además, debe garantizar que solo se pueda registrar una nota final por estudiante y materia, permitiendo su modificación únicamente bajo justificación y con registro de auditoría. Los estudiantes podrán visualizar sus notas confirmadas en su perfil académico.
Criterios de Validación:	<p>El sistema debe permitir al docente cargar notas únicamente en las materias donde esté asignado.</p> <p>El sistema debe validar que la nota ingresada esté en un rango permitido</p> <p>Los estudiantes deben poder consultar sus notas una vez confirmadas por el docente.</p> <p>Solo usuarios con rol Docente podrán realizar la carga o actualización de notas.</p>
Tareas Asociadas a la Implementación:	<p>Desarrollar validaciones de integridad</p> <p>Habilitar la consulta de notas confirmadas en el perfil académico del estudiante.</p> <p>Implementar pantalla de carga de notas para el docente (listado de alumnos y campo editable de nota).</p>