

# INGENIERIA DEL SOFTWARE

## PRACTICO 1 B

Grupo 1: Lucas Palacio, Benjamin Bavera, Ian Gandini

Comisión 1

Análisis de la problemática:

Primero leímos varias veces el enunciado. El problema parecía sencillo: Una universidad quería un sistema para organizar la información de sus estudiantes y docentes, pero nos encontramos con algunos desafíos.

Anotamos las primeras y posibles clases que serían la base del diagrama, como estudiante, profesor, materia. Estos serán nuestros principales actores. Identificamos los problemas mas importantes y complicados de modelar: las materias correlativas, y las diferentes relaciones de los estudiantes con las materias, ¿cómo podemos modelar que un estudiante pueda cursar ciertas materias dependiendo si tiene las correspondientes correlativas aprobadas para hacerlo?

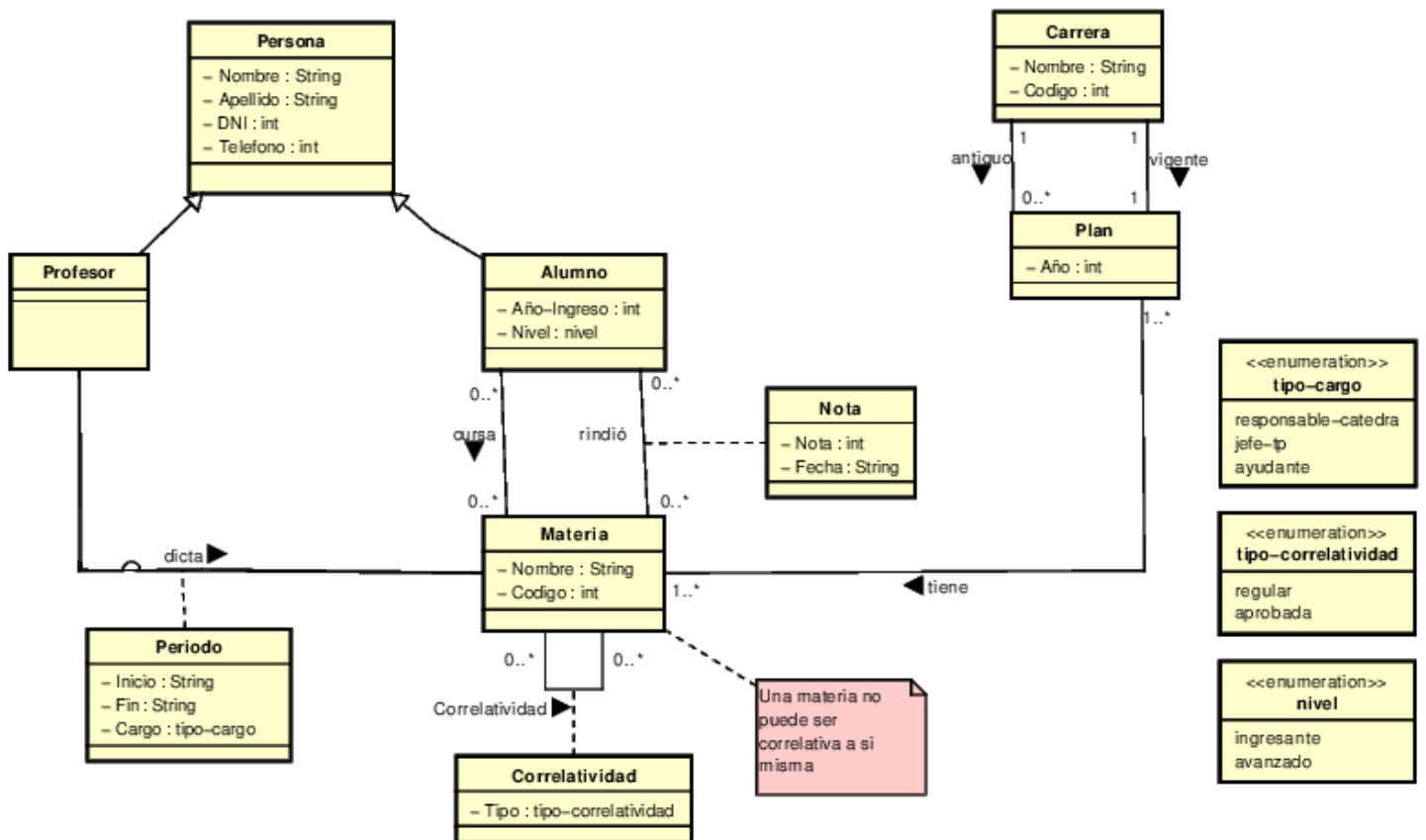
Anotamos los requerimientos funcionales y no funcionales. Luego, empezamos a plasmar estas ideas en una hoja, haciendo un boceto/borrador del diagrama.

Pensamos a las correlativas como una relación recursiva de materia, ya que una materia tiene correlativas (relación), y las correlativas son a su vez materias. Después fuimos agregando clases que faltaron y pensamos todas las relaciones que habría.

Actores claves del sistema:

- El personal de la Oficina de Alumnos, como administradores.
- Los estudiantes, para consultar su información.
- Los profesores, para cargar notas o consultar listados de sus alumnos.

A partir de la narrativa, elaboramos un diagrama de clases que modela el dominio del problema planteado e identificamos una serie de requerimientos funcionales y no funcionales:



### Requerimientos Funcionales (RF):

- Gestionar estudiantes
- Gestionar Docentes
- Gestionar Materias: plan (vigente y antiguos), carreras, correlativas
- Gestionar Inscripciones: validar si un estudiante puede o no cursar una materia
- Gestionar Cursadas: que materias esta cursando, cuales ya aprobó, nota
- Gestionar seguimiento académico
- Gestionar comunicación
- Gestionar recordatorios (Adicional) : que el sistema notifique cuando publiquen las notas, o cuando hayan inscripciones

### Requerimientos No Funcionales (RNF) :

- El sistema debe poder soportar la concurrencia de cientos de estudiantes
- El sistema debe estar disponible las 24 horas
- Interfaz accesible y amigable para cualquier edad
- Ejecutable en cualquier navegador moderno
- Acceso protegido con usuario y contraseña
- Respuesta de menos de 5 segundos
- Fácil de actualizar/mantener

### Historias de Usuario (HU) : Listado de prioridad

A partir del diagrama realizado y los requerimientos que nosotros identificamos, definimos una lista priorizada de las historias de usuario que considerábamos mas necesarias para el sistema.

#### 1. Gestión de Estudiantes:

HU-001: Como administrador, quiero registrar estudiantes con sus datos personales y de contacto, para mantener actualizada la información académica.

2. Gestión de Docentes:

HU-002: Como administrador, quiero registrar docentes con sus datos personales y asignarlos a materias, para gestionar correctamente el plantel docente.

3. Gestión de Materias:

HU-ADM-001: Como administrador, quiero registrar carreras y planes de estudio vigentes y antiguos, para gestionar la oferta académica de la universidad.

HU-ADM-002: Como administrador, quiero registrar materias con sus correlatividades, para automatizar la validación de inscripciones.

HU-ALUM-003: Como estudiante, quiero consultar las materias disponibles en mi plan de estudios, para saber cuáles puedo cursar según mi avance.

4. Gestión de Inscripciones:

HU-ALUM-004: Como estudiante, quiero inscribirme en materias, para avanzar en mi carrera sin trámites presenciales.

HU-SYS-001: Como sistema, quiero validar automáticamente si el estudiante cumple las correlatividades, para evitar inscripciones incorrectas.

HU-ADM-003: Como administrador, quiero consultar el listado de inscriptos por materia, para gestionar cupos y organización de cursadas.

5. Gestión de Cursadas:

HU-DOC-003: Como docente, quiero cargar las notas finales de mis estudiantes, para formalizar su aprobación en el sistema.

HU-ALUM-005: Como estudiante, quiero consultar mis notas y materias aprobadas, para llevar un seguimiento claro de mi avance académico.

HU-ADM-004: Como administrador, quiero generar reportes de cursadas, aprobados y promedios, para obtener estadísticas académicas.

#### 6. Gestión de Seguimiento Académico:

HU-ADM-005: Como administrador, quiero ver indicadores del progreso de cada estudiante, para detectar a tiempo riesgos de abandono.

HU-ADM-006: Como administrador, quiero identificar estudiantes destacados, para ofrecerles programas especiales

#### Historias de Usuario:

ID de HU	HU-001
Título	Gestionar Estudiantes
Declaración	Como administrador, quiero registrar estudiantes con sus datos personales y de contacto, para mantener actualizada la información académica.
Descripción detallada	<p>El sistema debe permitir al administrador dar de alta un estudiante ingresando datos personales (nombre, apellido, DNI, fecha de nacimiento), información de contacto (email, teléfono, dirección) y estado académico inicial (ingresante/avanzado).</p> <p>El sistema debe validar que no existan dos estudiantes con el mismo DNI.</p> <p>También poder dar de baja estudiantes que abandonen la carrera o la materia.</p>
Criterios de Validación	<p>Si se registra un estudiante con un DNI nuevo, queda guardado en la base de datos y se visualiza en el listado.</p> <p>Si se intenta registrar un DNI existente, se informa y no se modifica nada.</p> <p>Si se modifica un estudiante, el cambio se guarda y aparece en el listado.</p> <p>Si se intenta borrar un estudiante con inscripciones activas, se rechaza y se informa.</p> <p>Si no tiene inscripciones, se elimina correctamente.</p>
Tareas asociadas a la implementación	<p>Crear la tabla de los estudiantes en la base de datos.</p> <p>Desarrollar la interfaz para el administrador.</p>

ID de HU	HU-002
Título	Gestionar Materias
Declaración	Como administrador, quiero registrar materias con sus correlatividades, para automatizar la validación de inscripciones.
Descripción detallada	<p>El sistema debe permitir ingresar el código, nombre, carga horaria y plan de estudio al que pertenece la materia. Al aceptar, si ya existe otra materia con el mismo código en ese plan, se debe informar y no modificar nada. Si no existe, se registra en la base de datos. Una vez registrada la materia, el administrador podrá asociar una o varias materias como correlativas. Una materia no puede ser correlativa de sí misma. Se debe mostrar un listado de materias, permitir seleccionar una y editar sus datos (nombre, carga horaria, correlatividades). Al intentar eliminar una materia, el sistema debe validar que no tenga inscripciones activas ni esté configurada como correlativa de otra.</p>
Criterios de Validación	<p>Se registra correctamente una materia con un código único y queda disponible en el listado.</p> <p>Si se intenta registrar un código duplicado en el mismo plan, se informa y no se guarda.</p> <p>Al consultar una materia, se muestran correctamente sus correlatividades.</p> <p>No se puede asignar como correlativa la misma materia.</p> <p>Una materia con inscripciones activas o usada como correlativa de otra no puede ser eliminada.</p>
Tareas asociadas a la implementación	<p>Crear la tabla de las Materias en la base de datos.</p> <p>Desarrollar la interfaz para administrar las materias.</p> <p>Controlar restricciones antes de eliminar materias.</p>

ID de HU	HU-003
Título	Gestionar Docentes
Declaración	Como administrador, quiero registrar docentes con sus datos personales y asignarlos a materias, para gestionar correctamente el plantel docente.
Descripción detallada	<p>El sistema debe permitir ingresar datos personales del docente (nombre, apellido, DNI, email, teléfono, categoría/rol: responsable de cátedra, JTP, ayudante). Al aceptar, si ya existe un docente con el mismo DNI en la base de datos, se debe informar y no guardar nada. Si no existe, se registra en la base de datos.</p> <p>Una vez creado el docente, el administrador debe poder asignarlo a una o varias materias en un período académico específico. Un docente puede ocupar diferentes roles según la materia.</p> <p>Se debe mostrar un listado de docentes, permitir seleccionar uno y editar sus datos personales o sus asignaciones a materias. Los cambios deben guardarse en la base de datos.</p> <p>Al intentar eliminar un docente, el sistema debe validar que no tenga asignaciones activas en materias del período en curso.</p>
Criterios de Validación	<p>Se registra correctamente un docente nuevo con DNI único y queda visible en el listado.</p> <p>Si se intenta registrar un docente con un DNI ya existente, se informa y no se guarda.</p> <p>Al consultar un docente, se deben mostrar tanto sus datos personales como sus materias asignadas.</p> <p>Se puede asignar un docente a múltiples materias y con distintos roles.</p> <p>Si se intenta eliminar un docente con asignaciones activas, se informa el error y no se elimina.</p>
Tareas asociadas a la implementación	<p>Crear la tabla Docente en la base de datos.</p> <p>Implementar una interfaz para administrar los docentes.</p> <p>Desarrollar interfaz para asignar docentes a materias y gestionar sus roles.</p>