



**Departamento de Computación**  
**Asignatura: Ingeniería de Software I (3385)**  
**Año 2025**

**PRÁCTICO 1B:**  
**EJERCICIO TRANSVERSAL INTEGRADOR:**  
**SISTEMA DE GESTIÓN ESTUDIANTIL**

Como futuros profesionales de la industria del software les proponemos el desafío de embarcarse en la resolución de un caso que les permita aplicar los conocimientos que vamos abordando en la asignatura, trabajando en equipos de desarrollo y simulando un entorno de desarrollo que se aproxime a lo que sucede en la Industria de Software. Este ejercicio transversal integrador los acompañará a lo largo de los tres prácticos, evolucionando desde la captura de requerimientos, pasando por el diseño e implementación, hasta llegar a la etapa de prueba

**Narrativa: Entrevista con la Oficina de Alumnos**

Imaginemos la siguiente situación: tu equipo de desarrollo ha sido contactado por la Oficina de Alumnos de la Universidad, que busca modernizar y optimizar la gestión de la información académica de sus estudiantes. Tras una primera entrevista informal, a continuación se presenta un resumen de lo conversado:

*"..Mirá, la verdad es que hoy estamos trabajando con planillas, algunos sistemas viejos que no se conectan entre sí, y además la comunicación con los docentes y los chicos (estudiantes) se vuelve un desafío constante. Por eso necesitamos algo que nos centralice la información y nos simplifique la vida."*

*" Lo principal es poder llevar un buen registro de nuestros estudiantes: sus datos personales, información de contacto, y si son ingresantes o ya avanzados en la carrera. También es clave gestionar a nuestros profesores: quiénes son y qué materias están dictando."*

*" Otro punto clave es la oferta académica. Necesitamos poder cargar todas las materias de cada carrera, con sus planes de estudio y, algo súper importante, las correlatividades. Esto último hoy es un dolor de cabeza: muchas veces un estudiante quiere inscribirse en una materia y no cumple los requisitos, y no tenemos forma automática de validar eso."*

*" Y en cuanto a los estudiantes, nos interesa saber qué materias están cursando en cada momento, cuáles pueden cursar según su avance en el plan y las correlativas que ya aprobaron. Además, necesitamos un registro claro de la nota final de aprobación de cada materia. "*

*" También queremos poder asignar a los docentes a las materias según el período académico, y saber si son responsables de cátedra, jefes de trabajos prácticos o ayudantes. Eso nos permitiría organizar mucho mejor los equipos docentes."*

*" Por último, y más como un objetivo a futuro, nos gustaría contar con una herramienta que nos permita hacer seguimiento de los estudiantes: ver su progreso, detectar si están en riesgo de abandonar o, por el contrario, si van muy bien y podríamos ofrecerles algún programa especial. No sabemos exactamente cómo sería, pero creemos que un sistema podría ayudarnos."*

*" Los principales usuarios de este sistema serían: el personal de la Oficina de Alumnos, como administradores; los estudiantes, para consultar su información; y los profesores, para cargar notas o consultar listados de sus alumnos."*

## **Análisis, modelado y planificación inicial**

En este práctico, tu equipo deberá realizar el análisis inicial del sistema a partir de la narrativa presentada. El objetivo es que, trabajando de manera colaborativa, elaboren la documentación correspondiente, la cual deberán almacenar en una carpeta llamada **documentación** dentro de su repositorio de GitHub.:

### **Actividades a realizar:**

- **Análisis del enunciado y actores:** Presenten un breve resumen de cómo interpretaron la narrativa y los principales desafíos que el sistema debe abordar. Además, identifiquen y describan los **actores** clave del sistema.
- **Diagrama de clases de contexto (UML):** Elaborar un diagrama de clases que modele el dominio del problema planteado.
- **Requerimientos Funcionales (RF):** A partir del análisis del problema, elaboren un listado de requerimientos funcionales que, a su criterio, debería contemplar el sistema. Además de los RF de gestión, este es el espacio para que cada equipo proponga funcionalidades adicionales que aporten un valor agregado y permitan diferenciar su sistema del de otros grupos.
- **Requerimientos No Funcionales (RNF):** A partir de la narrativa y de sus propias consideraciones, identifiquen y enumeren los requerimientos no funcionales (RNF) que debería cumplir el sistema (por ejemplo: rendimiento, seguridad, usabilidad, entre otros).
- **Historias de Usuario (HU) y Backlog Inicial:** A partir de la narrativa, el diagrama de clases y los RF identificados previamente, realicen un listado priorizado con todas las Historias de Usuario (HU) que ustedes consideren pertinentes para este tipo de sistema. Para realizar este listado utilicen sólo la declaración estándar ("Como [Rol del Usuario], quiero [Acción], para [Beneficio]."). De esas HU identificadas, seleccionen tres y describan en detalle utilizando la plantilla.

ID de HU	Identificador único de la Historia de Usuario (ej., HU-001, HU-ALUM-001).
Título	Un título breve y conciso que resuma la funcionalidad.
Declaración	Sigue el formato estándar: "Como [Rol del Usuario], quiero [Acción], para [Beneficio]."
Descripción Detallada	Amplía la declaración, explicando con más detalle la funcionalidad. Incluye posibles reglas de negocio, flujos alternativos o consideraciones especiales.
Criterios de Validación (Criterios de Aceptación)	Una lista de condiciones claras y verificables que deben cumplirse para que la Historia de Usuario se considere terminada y correcta.
Tareas Asociadas a la Implementación	Un listado de las tareas técnicas concretas que el equipo deberá realizar para implementar esta HU (ej., "Crear modelo de datos para Alumno", "Desarrollar endpoint REST para obtener materias", "Implementar lógica de correlatividades").

### Formato de Entrega (en GitHub):

Todo el trabajo de este práctico debe estar versionado en el repositorio de GitHub correspondiente a **cada equipo**. Para ello, asegúrense de crear un repositorio propio del grupo y mantener allí toda la producción. Dentro del repositorio, generen una carpeta llamada **documentación** y, en su interior, un archivo en formato PDF denominado **PRÁCTICO\_1B\_DOCUMENTACIÓN**. Este archivo debe contener la resolución completa de las actividades solicitadas en los puntos anteriores. Verifiquen que las imágenes de los diagramas estén correctamente incluidas y sean visibles.

Para la elaboración del diagrama de clases en formato digital, utilicen una herramienta que soporte el lenguaje UML, como **Astah**, **Jude** u otra similar que consideren adecuada.