



PROGRAMACIÓN ORIENTADA A  
OBJETOS  
**Taller 04**  
II Semestre – 2025  
ITI - ICCI



Docentes: **Alejandro Paolini Godoy**  
**Cristhian Rabi Reyes**

Ayudantes: **Daniel Durán García**  
**Nicolás Rojas Bustos**

## Academic System of Professional Certifications – "AcademiCore"

La Universidad Católica del Norte ha lanzado un ambicioso programa de Certificaciones Profesionales en Tecnología, diseñado para complementar la formación de grado con competencias especializadas demandadas por la industria tecnológica.

Actualmente, más de 400 estudiantes de las carreras de Ingeniería en TI e Ingeniería Civil en Computación participan en tres líneas de certificación:

- Desarrollo de Software (DevOps, Arquitectura, Full-Stack)
- Sistemas Inteligentes (IA, Machine Learning, Big Data)
- Ciberseguridad (Ethical Hacking, Criptografía, Análisis Forense)

La gestión manual ha resultado en:

- Estudiantes perdiendo oportunidades de certificación por desconocimiento
- Coordinadores sin herramientas para identificar cuellos de botella
- Proceso de generación de certificados lento (hasta 3 semanas)
- Falta de métricas para evaluar efectividad de certificaciones

Un sistema académico integral que digitaliza y automatiza la gestión de certificaciones profesionales, proporcionando:

- **Transparencia:** Los estudiantes ven en tiempo real su progreso, requisitos pendientes y proyección de completación.
- **Eficiencia:** Los coordinadores acceden a dashboards analíticos que identifican patrones de rendimiento y asignaturas críticas.
- **Automatización:** Generación instantánea de certificados oficiales cuando se cumplen los requisitos académicos.

## Objetivos:

- Diseñar un modelo de dominio y un diagrama UML coherentes con la implementación en Java.
- Implementar herencia, polimorfismo, interfaces y arquitectura en un sistema orientado a objetos.
- Aplicar patrones de diseño (Singleton, Factory, Strategy, Visitor).
- Implementar una interfaz gráfica completa utilizando Java Swing.
- Utilizar colecciones avanzadas (LinkedList, List, ArrayList) para la gestión de datos.

# Files:

**usuarios.txt:** Contiene información sobre coordinadores y administradores.

```
admin;admin123;Administrador
coord.si;coord123;Coordinador;Sistemas Inteligentes
coord.cyber;cyber456;Coordinador;Ciberseguridad
coord.sw;software789;Coordinador;Desarrollo de Software
sysadmin;sysadmin2024;Administrador|
```

El archivo de texto se divide de la siguiente manera:

- Nombre de usuario: nombre de usuario.
- Contraseña: contraseña.
- Rol: rol del usuario.
- Información adicional: Área de coordinación (si es coordinador).

**estudiantes.txt:** Contiene información sobre los estudiantes del sistema.

```
12345678-9;Juan Carlos Pérez López;Ingeniería en Tecnologías de Información;5;juan.perez@alumnos.ucn.cl;password123
98765432-1;María José González Silva;Ingeniería en Tecnologías de Información;6;maria.gonzalez@alumnos.ucn.cl;mipassword456
11223344-5;Carlos Antonio Rodríguez Muñoz;Ingeniería en Tecnologías de Información;4;carlos.rodriguez@alumnos.ucn.cl;carlos789
22334455-6;Ana Beatriz Sánchez Torres;Ingeniería en Tecnologías de Información;7;ana.sanchez@alumnos.ucn.cl;anabeatriz321
```

El archivo de texto se divide de la siguiente manera:

- RUT: Identificador único del estudiante.
- Nombre: Nombre completo del estudiante.
- Carrera: Carrera del estudiante.
- Semestre: Semestre actual del estudiante.
- Correo electrónico: Dirección de correo electrónico del estudiante.
- Contraseña: Contraseña de acceso al sistema.

**cursos.txt:** Contiene información sobre las asignaturas.

```
DCCB-00246;Álgebra II;2;6;General;
SSed-00102;Comunicación Efectiva;2;5;General;
SSed-02184;Inglés II;2;4;General;
DCTE-00002;Diálogo, Fe y Cultura;2;2;General;
ECIN-00360;Programación Avanzada;3;6;Desarrollo de Software;ECIN-00361
```

El archivo de texto se divide de la siguiente manera:

- NRC: Código único de la asignatura.
- Nombre: Nombre de la asignatura.
- Semestre: Semestre en el que se imparte.
- Créditos: Número de créditos SCT.
- Área: Línea de certificación o departamento al que pertenece.
- Prerrequisitos: Códigos de los cursos prerrequisito separados por comas. (Si es que tiene).

**Certificaciones.txt:** Contiene información sobre líneas de certificación.

```
CERT-001;Sistemas Inteligentes;Especialización en inteligencia artificial, machine learning, redes neuronales y procesamiento de datos. Prepara a los estudiantes para desarrollar soluciones inteligentes y sistemas autónomos.;30;3
CERT-002;Ciberseguridad;Especialización en seguridad informática, criptografía, ethical hacking y análisis forense digital. Forma profesionales capaces de proteger sistemas y detectar vulnerabilidades.;25;3
CERT-003;Desarrollo de Software;Especialización en arquitectura de software, metodologías ágiles, DevOps y desarrollo full-stack. Capacita para liderar proyectos de desarrollo tecnológico.;35;3
```

El archivo de texto se divide de la siguiente manera:

- ID: Identificador único de la certificación.
- Nombre: Nombre de la certificación.
- Descripción: Descripción detallada de la certificación.
- Requisitos: Número mínimo de créditos requeridos. ~~
- Validez: Años de validez de la certificación.

**Registros.txt:** Contiene los registros de estudiantes para certificaciones.

```
12345678-9;CERT-003;2024-03-15;Activa;75
98765432-1;CERT-001;2024-02-20;Activa;85
11223344-5;CERT-002;2024-04-10;Activa;40
22334455-6;CERT-003;2023-08-25;Completada;100
```

El archivo de texto se divide de la siguiente manera:

- RUT: RUT del estudiante.
- ID de certificación: ID de certificación.
- Fecha de registro: Fecha de registro.
- Estado: Estado del registro (Activa, Completada, Suspendida).
- Progreso: Porcentaje de finalización de la certificación.

**Notas.txt:** Contiene las calificaciones de los estudiantes.

```
12345678-9;ECIN-00562;0.0;Cursando;2024-1
12345678-9;ECIN-00563;0.0;Cursando;2024-1
98765432-1;DCCB-00141;6.8;Aprobada;2021-1
98765432-1;DCCB-00142;6.5;Aprobada;2021-1
```

El texto se divide de la siguiente manera:

- RUT: RUT del estudiante.
- Código de la asignatura: Código de la asignatura.
- Calificación: Calificación obtenida.
- Estado: Estado de la asignatura (Aprobada, Reprobada, Cursando).
- Semestre: Semestre en el que se cursó la asignatura.

**Asignaturas\_certificaciones.txt:** Indica las asignaturas ligadas a sus líneas de certificación.

```
CERT-001;ECIN-00460
CERT-001;ECIN-00462
CERT-001;ECIN-00664
CERT-001;IA-00001
CERT-001;IA-00002
```

- ID: Identificador único de la certificación.
- NRC: Código único del curso que pertenece a la línea de certificación.

## **Requerimientos:**

Deberán crear tres menús accesibles mediante un inicio de sesión capaz de hacer lo siguiente:

- **Menú administrador (30 pts):**

- Gestión de Usuarios (**30 pts**):
  - Crear cuentas de estudiante o coordinador (6 pts).
  - Modificar cuentas de estudiante o coordinador (10 pts).
  - Eliminar cuentas de estudiante o coordinador (8 pts).
  - Restablecer contraseñas (6 pts).

- **Menú coordinador (50 pts):**

- Gestión de Certificaciones (**15 pts**):
  - Modificar línea de certificación (5 pts).
  - Generar certificados de estudiantes que hayan completado la línea de certificación (10 pts).
- Panel de Métricas y Análisis (**18 pts**):
  - Mostrar estadísticas de inscripciones de línea de certificación (9 pts).
  - Análisis de asignaturas críticas en la línea de certificación (9 pts).
- Gestión de Estudiantes (**17 pts**):
  - Consultar perfiles completos de estudiantes de la línea de certificación (10 pts).
  - Revisar y validar avances académicos (7 pts).

- **Menú estudiante (85 pts):**

- Perfil del Estudiante (**25 pts**):
  - Visualizar información personal completa (5 pts).
  - Mostrar malla curricular con asignaturas aprobadas, reprobadas y pendientes (12 pts).
  - Calcular y mostrar promedio general y por semestre (8 pts).
- Malla Curricular Interactiva (**27 pts**):
  - Visualización gráfica de la malla curricular por semestres (12 pts).
  - Indicadores visuales de estado de asignaturas (colores diferentes) (10 pts).
  - Información detallada al hacer clic en cada asignatura (5 pts).
- Inscripción a Certificaciones (**18 pts**):
  - Listar líneas de certificación disponibles (4 pts).
  - Mostrar requisitos y descripción de cada línea (4 pts).
  - Proceso de inscripción con validaciones (7 pts).
  - Verificar prerrequisitos académicos (3 pts).
- Seguimiento de Progreso (**15 pts**):
  - Dashboard personal con progreso en certificaciones inscritas (5 pts).
  - Aplicar Visitor para diferentes acciones según tipo de certificación (7 pts).
  - Mostrar asignaturas pendientes para completar certificaciones (3 pts).

## Aclaraciones:

- Menú Administrador:
  - En caso de eliminar una cuenta, se deberá eliminar todas las referencias a esta.
- El menú debe ser fluido, sin interrupciones, con manejo de errores

## Consideraciones:

- Comunicación y Dudas
  - Las consultas deben realizarse por correo institucional o WhatsApp oficial
  - Las dudas serán respondidas hasta las **17:00 del jueves 11 de diciembre**
  - Situaciones especiales deben comunicarse con anticipación
  - Puede haber excepciones según la situación (previa justificación)
- Trabajo en Equipo
  - Talleres en pareja (2 personas por grupo)
  - **Ambos integrantes deben tener commits verificables en el repositorio**
  - Si un integrante no realiza aportes visibles, puede recibir nota diferenciada
  - Los commits deben ser descriptivos y frecuentes (no todo en un solo commit)
- Requisitos Técnicos Obligatorios (Nota 1.0 si no se cumple)
  - Uso de Programación Orientada a Objetos (herencia, polimorfismo, interfaces)
  - Arquitectura de 3 capas (dominio, lógica, presentación)
  - Implementación de 4 patrones de diseño (Singleton, Factory, Strategy, Visitor)
  - Interfaz gráfica completa con Java Swing, Java AWT y Java FX
  - Nombres completos, RUT y carrera en primeras 5 líneas de TODAS las clases principales
  - Directorio con formato:  
nombreApellidoIntegrante1\_nombreApellidoIntegrante2\_Taller4POO
  - El programa debe ejecutarse sin errores inmediatos (incluye paths)
- Entrega
  - **Formato:** Archivo .txt con link al repositorio GitHub
  - **Nombre archivo:**  
nombreApellidoIntegrante1\_nombreApellidoIntegrante2\_Taller4POO.txt
  - **Contenido del .txt:**  
Integrante 1: Nombre Completo - RUT - Usuario GitHub  
Integrante 2: Nombre Completo - RUT - Usuario GitHub  
Link repositorio: <https://github.com/usuario/repositorio>
- Estructura del repositorio
  - Proyecto Eclipse en la raíz (NO dentro de carpetas adicionales)
  - README.md en formato Markdown
  - Archivo PDF con diagrama de clases
  - .gitignore configurado
- Revisión
  - Se revisará **el último commit antes de las 12:59:59 del 12 de diciembre de 2025**
  - No se aceptarán entregas atrasadas (ni 1 minuto después)
  - **Solo una persona del grupo debe subir el archivo de entrega**
- Librerías Externas
  - Se pueden usar librerías previo visto bueno de los ayudantes
  - Solicitar autorización por correo con al menos 5 días de anticipación
  - Justificar el uso de la librería
- Documentación Obligatoria
  - **JavaDoc:** Todas las clases y métodos públicos
  - **README.md:** Debe incluir:
    - Descripción del proyecto
    - Integrantes con RUT y usuario GitHub
    - Estructura del proyecto (paquetes y clases principales)
    - Patrones de diseño implementados

- Instrucciones de ejecución
- **Diagrama de Clases:** PDF generado con herramienta digital (no a mano)
  - Debe incluir TODAS las clases, interfaces y patrones
  - Notación UML correcta
  - Relaciones claramente identificadas
- Penalizaciones (Nota 1.0)
  - Copia total o parcial de otro grupo o fuente externa
  - No cumplir requisitos técnicos obligatorios mencionados arriba
  - Código que no compile o se cierre inmediatamente por error no manejado
  - Falta de nombres/RUT/carrera en las primeras 5 líneas
  - Directorio o archivo de entrega con nombre incorrecto
  - Proyecto Eclipse no en la raíz del repositorio

## Entregables:

- Diagrama de clases.
- Código fuente del programa.
- Archivos txt que se ocuparon.
- Documentación.

## Criterios de evaluación:

Criterio	Subcategoría	Puntos
<b>Código (50 pts)</b>		
	Nombres de variables	12
	Estructura y organización	18
	Buenas prácticas	10
	Patrones de diseño	10
<b>Diseño (30 pts)</b>		
	Diagrama de clases UML	30
<b>Funcionalidad (165 pts)</b>		
	Menú Administrador	30
	Menú Coordinador	50
	Menú Estudiante	85
<b>Interfaz Gráfica (30 pts)</b>		
	Funcionalidad y usabilidad	20
	Diseño y componentes	10
<b>Documentación (25 pts)</b>		
	JavaDoc	10
	README	10
	Comentarios	5
<b>TOTAL</b>		<b>300</b>

**Puntaje total:** 300  
**Puntaje nota mínima:** 177  
**Fecha de Inicio:** lunes 24 de noviembre.  
**Fecha máxima de entrega:** viernes 12 de diciembre (12:59:59).

Correos: [daniel.duran02@alumnos.ucn.cl](mailto:daniel.duran02@alumnos.ucn.cl), [nicolas.rojas11@alumnos.ucn.cl](mailto:nicolas.rojas11@alumnos.ucn.cl)  
 Grupo de WhatsApp: [https://chat.whatsapp.com/GvnkfE5RUViJ0AgVtyusbs?mode=ac\\_t](https://chat.whatsapp.com/GvnkfE5RUViJ0AgVtyusbs?mode=ac_t)