

Presidencia Roque Sáenz Peña, 03 de julio de 2025

RESOLUCIÓN N° 187/2025 - C.D.C.B. y A.

VISTO:

El Expediente N° 01-2025-02596 sobre propuesta de Programa de la asignatura Metodología de Construcción de Sistemas de Software de la carrera Ingeniería en Sistemas de Información, iniciado por la Directora de Carrera Ing. Zachman, Patricia; y

CONSIDERANDO:

Que la asignatura CO5-Metodología de Construcción de Sistemas de Software es un Curso Optativo y se dicta en 5° año – 1^{er} cuatrimestre de la Carrera Ingeniería en Sistemas de Información;

Que el Programa Analítico contempla los contenidos mínimos y la carga horaria propuestos en el Plan de Estudios de la Carrera aprobado por Resolución N°063/19-C.S.;

Que las asignaturas correlativas respetan lo establecido en el Sistema de Correlatividades de la Carrera aprobado por Resolución N° 088/19-C.S.;

Que los objetivos planteados guardan coherencia con los contenidos, los métodos pedagógicos y de evaluación propuestos, y la fundamentación refleja la relevancia de la asignatura en la formación de los futuros profesionales;

Que los Trabajos Prácticos planteados son pertinentes y adecuados y la bibliografía propuesta es actualizada;

Que se propone la modalidad de Aprobación mediante Exámenes Parciales (Promocional) según lo establece el Artículo 33° del Punto 3.5-Capítulo 3: Aprobación mediante Exámenes Parciales (PROMOCIONAL) de la Resolución N°080/12- C.S., Reglamento Académico de Alumnos;

Lo aprobado en sesión de la fecha.

POR ELLO:

**EL CONSEJO DEPARTAMENTAL
DEL DEPARTAMENTO DE CIENCIAS BÁSICAS Y APLICADAS DE LA
UNIVERSIDAD NACIONAL DEL CHACO AUSTRAL**

RESUELVE:

ARTÍCULO 1°: APROBAR el Programa de la asignatura Metodología de Construcción de Sistemas de Software de la carrera de Ingeniería en Sistemas de Información, que como Anexo Único forma parte de la presente Resolución.

ARTÍCULO 2°: Regístrese, comuníquese, y archívese.



Nora B. Okun
Dra. Nora B. Okun
Directora
Dpto. de Cs. Básicas y Apl.

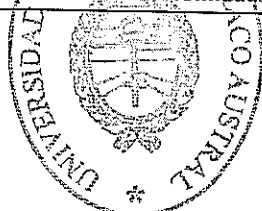
12.11.2019
12.11.2019
12.11.2019

12.11.2019
12.11.2019
12.11.2019

ANEXO: PROGRAMA DE ASIGNATURA

 UNCAUS UNIVERSIDAD NACIONAL DEL CHACO AUSTRAL		Curso Optativo CO 5-Metodología de Construcción de Sistemas de Software Plan de Estudios Resolución N°063/19-C.S.	
Carga Horaria: 50 horas Teóricas: 25 horas Prácticas: 25 horas		Programa vigente desde: 2025	
Carrera		Año	Cuatrimestre
Ingeniería en Sistemas de Información		Quinto	Primero
CORRELATIVAS PRECEDENTES		CORRELATIVAS SUBSIGUIENTES	
Asignaturas		Asignaturas	
Para cursar		Para rendir	
Regularizadas	Aprobadas	Aprobadas	
- Ingeniería de software	- Sistemas operativos	-----	
DOCENTES:		Esp. Lic. Rosenzvaig, Federico. Profesor Adjunto.	
FUNDAMENTACIÓN:		La asignatura Metodología para la Construcción de Sistemas de Software aporta al perfil del Ingeniero en Sistemas de Información las herramientas para planificar, diseñar y gestionar proyectos de desarrollo de software de manera eficiente y profesional. A través del estudio de metodologías tradicionales y ágiles, el estudiante adquiere competencias para seleccionar enfoques adecuados según el contexto, documentar procesos, trabajar en equipo y garantizar la calidad del producto final. Esta formación fortalece tanto las habilidades técnicas como las comunicativas, preparándolo para enfrentar desafíos reales en el desarrollo de soluciones informáticas integrales.	
OBJETIVOS:		Objetivo General: Desarrollar la capacidad de conceptualizar, analizar y diseñar sistemas de información desde el paradigma de la orientación a objetos, integrando una perspectiva teórica y práctica que les permita abordar problemas complejos de manera estructurada y eficiente. Objetivos Específicos: • Aplicar herramientas del análisis Orientado a Objetos, en particular UML y la escuela de Jacobson. • Utilizar la herramienta Rational Rose. • Mirar los sistemas de información desde la orientación a objetos. • Identificar tecnologías avanzadas en orientación a objetos.	
CONTENIDOS MÍNIMOS:		CASE Orientada a Objetos. Tutorial de Rational Rose. Caso práctico con UML. Metodología Métrica v.3. Caso práctico AGA 2000. Caso práctico CONCAB.	
MÉTODOS PEDAGÓGICOS:		Para este curso se propone desarrollar las clases en laboratorio de informática como un espacio de taller para la construcción de conocimientos mediante la práctica. En base a ello se realizarán exposiciones destinadas a la presentación de conceptos,	

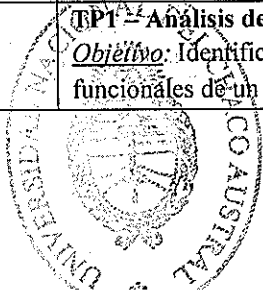
[Firma]





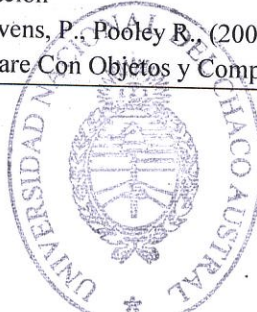
	metodologías, técnicas y se desarrollarán los conocimientos conceptuales metodológicos de la asignatura.
MÉTODOS DE EVALUACIÓN:	En la evaluación se tendrá en cuenta tanto los conocimientos adquiridos como también la participación en clase y la presentación en tiempo y forma de los trabajos prácticos. Forma de evaluación y controles: Se tomarán dos exámenes escritos, uno de medio término y otro, integrador, al finalizar el cursado. Instancias de aprobación: Se considerará aprobado quien apruebe ambos exámenes y presente los trabajos prácticos en tiempo y forma. Instancias de recuperación: Se recuperará el examen integrador, luego del examen de fin de curso, según corresponda.
PROGRAMA ANALÍTICO DE CONTENIDOS:	UNIDAD 1: PARADIGMA ORIENTADO A OBJETOS Paradigma Orientado a Objeto. Clase, Objeto, Identidad, Clasificación, Polimorfismo, Herencia, y Encapsulado. Atributos del sistema. Atributos en las funciones. UML. Historia. Tipos de diagramas. Herramientas CASE: Rational Rose: características. UNIDAD 2: CASOS DE USO, INTERACCIÓN ENTRE OBJETOS, DIAGRAMA DE CLASES Caso de Uso: de alto nivel y expandido. Curso normal y alterno de los eventos. Clasificación de los Casos de uso. Diagrama de Caso de Usos. Ejercitación utilizando Herramientas CASE. Diagramas de secuencia y de colaboración: mensajes, objetos y notas. Ejercitación utilizando Herramientas CASE. Clases, Instancias de una clase, Objetos. Definición. Atributos, Comportamiento, Especificación de Operaciones Modelando Asociaciones como Clases. Ejercitación utilizando Herramientas CASE utilizando Herramientas CASE. UNIDAD 3: TOPICOS AVANZADOS Clases Abstractas. Herencia Múltiple. Programación en tres capas. Escenarios y Trace de Eventos. Diagramas de Estados anidados. Diagramas de un ciclo y de Loop. Guardas de Transición. Ejercitación utilizando Herramientas CASE. Caso práctico AGA 2000. Caso práctico CONCAB. UNIDAD 4: TECNOLOGIAS DE DESARROLLO WEB Y PLATAFORMAS Modelo de Aplicaciones distribuidas. Contenedores. Web Services. Empaquetado de Aplicaciones. Despliegues. Roles en el equipo de sistemas. Estructura de un servidor de aplicaciones. Service Oriented Architecture. Tecnología JSF (Java Server Faces). Implementaciones. Configuración de aplicaciones JSF. Ventajas y desventajas del protocolo HTTP. Google Web Toolkit (GWT). Comparación con JSF, ventajas y desventajas. Introducción a Ext-JS. Comparación con GWT y JSF. Integración de GWT con Ext-JS. Plataforma Android. Conceptos, tecnologías, arquitecturas. Alternativas a diferentes tipos de dispositivos. Conexiones entre dispositivos mediante BlueTooth y servidores mediante HTTP. Librerías para aplicaciones basadas en ubicación.
	TPI - Análisis de Requisitos y Relevamiento <i>Objetivo:</i> Identificar y documentar los requerimientos funcionales y no funcionales de un sistema.

[Handwritten signature]



///Res. N° 187/2025-DCByA.

PROGRAMA ANALÍTICO DE TRABAJOS PRÁCTICOS:	<u>Actividades:</u> • Análisis de un caso propuesto (empresa ficticia). • Relevamiento de requisitos mediante entrevistas simuladas y lectura de documentación. • Elaboración del Documento de Especificación de Requisitos (ERU / SRS). <u>Entregable:</u> Documento formal con requisitos detallados, modelo de casos de uso. TP2 – Modelado del Sistema <u>Objetivo:</u> Aplicar herramientas de modelado para representar la estructura y comportamiento del sistema. <u>Actividades:</u> • Diagramas UML: Casos de uso, clases, secuencia y actividades. • Definición de entidades clave y relaciones. • Introducción al modelado ágil (modelos livianos). <u>Entregable:</u> Conjunto de diagramas UML + glosario de entidades. TP3 – Diseño Arquitectónico <u>Objetivo:</u> Comprender y aplicar principios de diseño de software y arquitectura de sistemas. <u>Actividades:</u> • Selección de un estilo arquitectónico (cliente-servidor, MVC, servicio REST, microservicios, etc.). • Diseño de componentes y su interacción. • Identificación de patrones de diseño aplicables. <u>Entregable:</u> Diagrama de arquitectura + justificación técnica.
BIBLIOGRAFÍA:	Bibliografía Principal: • Eckel, B., (2006) Thinking in Java. Editorial Prentice-Hall • FONTELA C., (2008) Orientación a Objetos : Diseño y Programación. Editorial Nueva Liberia • Fontela C., (2011) UML: Modelado De Software. Editorial Alfaomega Grupo Editor • Hall M., (2000) Core Servlets and Java Server Pages. Editorial Prentice-Hall • Mclaughlin B., (2001) Java and XML. Second Edition. Editorial O'Reilly • Quatrani, T. (1998). Visual Modeling With Rational Rose And Uml Editorial Addison-Wesley • UML con Rational Rose, http://book/uml-con-rationalrose10201.html Bibliografía Complementaria: • Ribas Lequerica, J. (2013) Desarrollo De Aplicaciones Para Android. Editorial Anaya Multimedia • Booch, G., (2001) Análisis De Diseño Orientado A Objetos Con Aplicaciones (2ª Ed.) Editorial Alhambra Mexicana, S.A. • Gómez, C., (2003) Diseño De Sistemas Software En Uml, Ed.Upe • James Rumbaugh, J., (2007) El Lenguaje Unificado De Modelado. Manual De Referencia (2ª Ed.) Editorial Addison-Wesley • Larman J., (2002) Uml Y Patrones 2da Edic, Editorial Pearson Educación • Stevens, P., Pooley B., (2007) Utilización de Uml en Ingeniería Del Software Con Objetos y Componentes. Editorial Addison-Wesley



Nora B. Okun
Dra. Nora B. Okun
Directora
Dpto. de Cs. Básicas y Aplicadas

