



Universidad Nacional de Río Cuarto  
Facultad de Ciencias Exactas, Físico-Químicas y Naturales  
Departamento de Computación

**UNIVERSIDAD NACIONAL DE RÍO CUARTO**  
**FACULTAD DE CIENCIAS EXACTAS, FÍSICO-QUÍMICAS Y NATURALES**  
**DEPARTAMENTO DE COMPUTACIÓN**

**CARRERA/S: LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN (14) -  
ANALISTA EN COMPUTACIÓN (12)**

**PLAN DE ESTUDIOS: 1999. V1**

**ASIGNATURA: Ingeniería de Software**

**CÓDIGO: 3304**

**MODALIDAD DE CURSADO: Presencial**

**DOCENTE RESPONSABLE: Mg. Marcela Daniele**

**EQUIPO DOCENTE: Dr. Marcelo Uva, Lic. Ariel Arsaute, Prof. Daniela Solivellas**

**AÑO ACADÉMICO: 2024**

**RÉGIMEN DE LA ASIGNATURA: Cuatrimestral (2do cuatrimestre)**

**UBICACIÓN EN EL PLAN DE ESTUDIO: Tercer año**

**RÉGIMEN DE CORRELATIVIDADES: (para cursado, según plan de estudio vigente)**

Asignaturas aprobadas: --. Asignaturas regulares: Análisis y Diseño de Sistemas (3303)

**CARÁCTER DE LA ASIGNATURA: Obligatoria**

**CARGA HORARIA TOTAL: 170 horas**

<b>Teóricas:</b>	<b>60 hs</b>	<b>Prácticas:</b>	<b>60 hs</b>	<b>Teóricas -Prácticas:</b>	<b>.... hs</b>	<b>Laboratorio:</b>	<b>50 hs</b>
------------------	--------------	-------------------	--------------	---------------------------------	----------------	---------------------	--------------

**CARGA HORARIA SEMANAL: 12 horas (según el plan de estudio vigente)**

<b>Teóricas:</b>	<b>4 hs</b>	<b>Prácticas:</b>	<b>4 hs</b>	<b>Teóricas -Prácticas:</b>	<b>.... hs</b>	<b>Laboratorio:</b>	<b>3 hs</b>
------------------	-------------	-------------------	-------------	---------------------------------	----------------	---------------------	-------------

## **1. CONTEXTUALIZACIÓN DE LA ASIGNATURA**

Los temas abordados en Ingeniería de Software amplían y enriquecen las competencias adquiridas por los estudiantes a lo largo de su formación, capacitándose para desarrollar e implementar soluciones informáticas con amplia gama de desafíos en diversos entornos y campos. Se refuerzan habilidades fundamentales en programación, diseño y modelado de software, consolidando la capacidad de diseñar soluciones genéricas que pueden aplicarse a diferentes tipos de problemas. Además, con esta asignatura el estudiante se prepara con un sólido bagaje de conocimientos que le permite adaptarse y evolucionar ante los cambios tecnológicos y de entorno, lo que constituye una valiosa contribución a su futuro profesional.



Universidad Nacional de Río Cuarto  
Facultad de Ciencias Exactas, Físico-Químicas y Naturales  
Departamento de Computación

## 2. OBJETIVOS PROPUESTOS

- Introducir conceptos fundamentales de Ingeniería de Software y transversales al ciclo de vida de desarrollo de un producto de software, que permiten su adecuado gerenciamiento, como: planificación y gestión de proyectos, medición, estimación, calidad y modelos de mejora de procesos de desarrollo, tecnologías de software distribuidas, gestión de la configuración de software.
- Modelar soluciones a problemas computacionales utilizando lenguajes de especificación, como: Redes de Petri y Object Constraint Language (OCL).
- Promover el uso de herramientas de sistemas colaborativos, y fortalecer la vinculación entre estudiantes y el trabajo en equipo, de manera presencial o remota, para el desarrollo de software.
- Abordar conceptos sobre aspectos legales y auditoría de sistemas. Brindar fundamentos sobre propiedad Intelectual, licenciamiento de software y contratos informáticos.

## 3. EJES TEMÁTICOS ESTRUCTURANTES DE LA ASIGNATURA Y ESPECIFICACIÓN DE CONTENIDOS

### 3.1. Contenidos mínimos (según plan de estudio vigente)

Gestión de Proyectos. Sistemas Colaborativos. Métricas del Proceso, del Proyecto y del Producto. Métricas Orientadas a Objetos. Planificación de Proyectos. Estimación. Técnicas de Descomposición. Métodos empíricos de estimación. Puntos de Función. Puntos de Casos de Uso. Planificación Temporal. Gestión de Configuración de Software. Elementos de Configuración de Software. Control de versiones y cambios. Auditoría de la Configuración. Herramientas. Aseguramiento de la Calidad del Software. Revisiones Técnicas Formales. Estándares de Calidad ISO 9000. Especificación de Software. Redes de Petri. Object Constraint Language (OCL). Arquitectura de Software y Patrones Arquitecturales. Introducción a Sistemas Distribuidos. Introducción a Tecnologías Web. Propiedad Intelectual, licenciamiento de software y contratos informáticos. Aspectos legales. Auditoría y Peritaje. Los contenidos se detallan en el programa analítico.

### 3.2. Ejes temáticos o unidades

#### Unidad 1: Gestión de Proyectos

Personas. Organización. Coordinación y Comunicación. Producto. Proceso. Proyecto. Herramientas. El Proyecto construye el Producto. Artefactos de un Producto. El Proyecto dirige Proyectos. Herramientas en el Proceso. Prácticas Críticas. Conceptos fundamentales de Sistemas Colaborativos.

#### Unidad 2: Planificación temporal y seguimiento de Proyectos.

Planificación Temporal. Principios básicos. Distribución de esfuerzo. Conjunto de tareas, red de interdependencias. Función de la gerencia en la planificación temporal. Métodos. PERT. Gráfico de Gantt. Seguimiento de Proyectos. Herramientas.

#### Unidad 3: Métricas y Estimación de Proyectos de Software.

Medidas. Métricas. Indicadores. Métricas del Proceso. Métricas del Proyecto. Medición del Software. Métricas orientadas al Tamaño. Métricas orientadas a la función. Puntos de Función, Puntos de Casos de Uso, Puntos de Historia. Modelos Empíricos de Estimación. Modelo COCOMO y COCOMO II. Métricas para Sistemas Orientados a Objetos: Objetivo. Métricas orientadas a Clases. Métricas CK.



Universidad Nacional de Río Cuarto  
 Facultad de Ciencias Exactas, Físico-Químicas y Naturales  
 Departamento de Computación

Métricas de Lorenz y Kidd. Métricas orientadas a Operaciones. Métricas para Proyectos orientados a Objetos.

#### **Unidad 4: Especificación de Software**

Uso. Calidad. Verificación. Declaración de tipos, valores, axiomas. Tipos base. Redes de Petri. Modelado, Aplicación y Propiedades. Grafo de Alcanzabilidad. Especificaciones, definición de reglas, usando Object Constraint Language (OCL).

#### **Unidad 5: Gestión de la Configuración del Software**

Introducción. Línea Base. Elementos de Configuración de Software. El Proceso de Gestión de la Configuración del Software. Control de versiones. Control de Cambios. Auditoría de la Configuración. Herramientas.

#### **Unidad 6: Arquitectura de Software y Patrones Arquitecturales**

Arquitectura. Conceptos. Descripción. Un proceso de desarrollo centrado en la Arquitectura. Vistas de la arquitectura de los diferentes modelos del proceso. Patrones. Categoría de Patrones. Patrones Arquitecturales: Layers. Pipes y Filter. Backboard. Broker. Model-View-Controller. Mikrokernel.

#### **Unidad 7: Aseguramiento de la Calidad del Software**

Conceptos. Calidad de Software. Factores de Calidad de McCall. Costos. Actividades SQA. Revisiones de Software. Revisiones Técnicas Formales. Calidad Estadística. Medidas. Fiabilidad. Disponibilidad. Seguridad. Plan de SQA. Estándares de Calidad ISO 9000.

#### **Unidad 8: Propiedad Intelectual, licenciamiento de software y contratos informáticos. Aspectos legales. Auditoría y Peritaje.**

Introducción. Conceptos Básicos. Fundamentos. Derecho de autor. Patentes y Marcas. Licencias de software libre. Software Libre y Open Source. Códigos de Ética. Leyes de la Propiedad Intelectual. Definición y Objetivos de la Auditoría Informática. Tipos y clases de Auditoría.

### **4. ACTIVIDADES A DESARROLLAR**

Se dictan las clases teóricas a todos los estudiantes cursantes, con un total de entre 3 hs y 5 hs semanales. Además, los estudiantes asisten a las clases prácticas dos veces por semana de 2 hs de duración cada una. Y como actividad integradora, se desarrolla un proyecto/taller en equipos de 2 a 3 estudiantes.

**CLASES TEÓRICAS:** Presencial en aula/laboratorios, 60 Hs totales

**CLASES PRÁCTICAS:** Presencial en laboratorios, 60 Hs totales

**CLASES PROYECTO/TALLER:** Presencial en laboratorios, 50 Hs totales

### **5. PROGRAMAS Y/O PROYECTOS PEDAGÓGICOS INNOVADORES E INCLUSIVOS**

### **6. CRONOGRAMA TENTATIVO DE CLASES E INSTANCIAS EVALUATIVAS**

TEÓRICOS	TRABAJOS PRÁCTICOS	TALLER
1. Introducción. Gestión de Proyectos de Software.	1. Gestión de proyectos: Producto, Persona, Proceso, Proyecto (1 clase)	Inicio del Taller. Revisión y organización de equipos
2. Planificación Temporal. PERT. GANTT)	2. PERT. GANTT (2 clases)	Herramientas Refactoring
3. Planificación de Proyectos. Estimación. PF. PCU. COCOMO. COCOMO II	3. Puntos de Función, Puntos de Casos de Uso (1 clase) 4. COCOMO. COCOMO II (2 clases)	Ingeniería inversa Arquitectura



Universidad Nacional de Río Cuarto  
 Facultad de Ciencias Exactas, Físico-Químicas y Naturales  
 Departamento de Computación

4. Métricas para Sistemas Orientados a Objetos	5. MOO (1 clase)	Planificación y Estimación ágil
5. Gestión de Configuración del Software	6 GCS (2 clases)	Medición y análisis de SW
6. Especificación de Software. Redes de Petri.	7. PN (4 clases)	Mostrar proceso de GCS (PT, Git, Github)
7. Especificación de software. OCL	8. OCL (4 clases)	Presentación al cliente GCS implementación
8. Arquitectura de Software. Patrones Arquitecturales	9. Patrones Arquitecturales (4 clases)	Ajustes y preparación de la presentación final
9. Calidad de Software. SQA		Presentación Final
10. Auditoría. Propiedad Intelectual.		

#### Instancias Evaluativas. fechas tentativas.

- **1er Parcial.** (24/9) Temas: Planificación temporal. Estimación y Métricas
- **2do Parcial.** (15/10) Temas: modelado y especificación de software con Redes de Petri
- **3er Parcial.** (5/11) Temas: modelado y especificación de software con OCL
- Investigación y exposición grupal. Temas: patrones arquitecturales.
- **Recuperatorios** (14/11) del 1er, 2do y 3er parcial
- **Taller/Proyecto grupal**, cada grupo sigue la planificación acordada y realiza la defensa final de su proyecto.
- **Coloquio final teórico:** Al final de la asignatura, los estudiantes tienen la posibilidad de presentarse a un coloquio, fundamentalmente teórico, incluyendo todos los temas del programa de la asignatura. El coloquio final es escrito y opcional para el estudiante que quiere acceder a la promoción.

## 7. BIBLIOGRAFÍA

### 7.1. Bibliografía obligatoria y de consulta

- Software Engineering: A Practitioner's Approach. Roger S Pressman. 8th Edition. McGraw-Hill Educ, 2014.
- Concise Guide to Software Engineering From Fundamentals to Application Methods, Gerard O'Regan., 2016.
- Web Engineering. A Practitioner's Approach. Roger S. Pressman, David Lowe, 2009.
- Agile Estimating and Planning - Mike Cohn, 2005.
- An Integrated Approach to Software Engineering. Springer. Pankaj Jalote. 2005.
- Advanced Systems Design with Java, UML and MDA. Kevin Lano, 2005
- Fundamentals of Software Engineering. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli - Prentice Hall, 1991.
- Software Engineering. Ian Sommerville. Addison Wesley, 9na Edición. 2011.
- Pattern-Oriented Software Architecture, Volume 1: A System of Patterns. R. Meunier, H. Rohnert, P. Sommerlad, M. Stal.
- Handbook of Software Quality Assurance- G. Gordon Schulmeyer, James I. McManus - Prentice Hall, 3ra Edición. 1999.
- El Proceso Unificado de Desarrollo de Software. Ivar Jacobson, Grady Booch, James Rumbaugh - Addison-Wesley, 2000.
- Software Testing: A Craftsman's Approach. Paul C. Jorgensen - CRC Press, 1995.
- Object Oriented Software Construction. Bertrand Meyer. Prentice Hall. 1997.
- Auditoría en Informática. 2da Edición. José Antonio Echenique García. Mc Graw Hill.
- Auditoria Informática: Un Enfoque Práctico. RAMA, Piattini, M., 2001

## 8. DÍA Y HORARIOS DE CLASES

### HORARIO DE CLASES TEÓRICAS

Lunes 13 a 16 hs (algunos martes 11 a 13 hs se completan clases teóricas)

### HORARIO DE CLASES PRÁCTICAS

Comisión 1: Martes 9 a 11 hs y Jueves 10 a 12 hs

Comisión 2: Martes 14 a 16 hs y Jueves 16 a 18 hs

**HORARIO DE PROYECTO/TALLER:** Martes de 11 a 13 hs

## 9. DÍA Y HORARIO DE CLASES DE CONSULTAS



Universidad Nacional de Río Cuarto  
Facultad de Ciencias Exactas, Físico-Químicas y Naturales  
Departamento de Computación

Todos los docentes de la asignatura están disponibles todas las semanas para clases de consultas. Los días y horarios no son fijos, se coordinan con la disponibilidad y demanda de los estudiantes.

## 10. REQUISITOS PARA OBTENER LA REGULARIDAD Y LA PROMOCIÓN

**CONDICIONES DE REGULARIDAD:** aprobar tres exámenes parciales, o su correspondiente recuperatorio, y aprobar el proyecto/taller final.

**CONDICIONES DE PROMOCIÓN:** la asignatura posee promoción. Los estudiantes que regularizan la asignatura, y cumplen y aprueban todas las instancias parciales evaluativas con el 70% o más, tienen la posibilidad de rendir un coloquio final teórico, y en caso de aprobarlo, obtienen la promoción de la asignatura.

## 11. CARACTERÍSTICAS, MODALIDAD Y CRITERIOS DE LAS INSTANCIAS EVALUATIVAS

**EVALUACIONES PARCIALES:** consiste de tres exámenes parciales escritos en los que se distribuyen todos los contenidos de la asignatura. Existe una instancia de examen recuperatorio, que se desarrolla en una fecha posterior a las instancias evaluativas de los tres parciales, donde cada estudiante tiene la posibilidad de recuperar los temas que no pudo alcanzar en los parciales.

**EVALUACIÓN del PROYECTO/TALLER INTEGRADOR:** Los estudiantes, en equipos de dos o tres, trabajan de manera colaborativa sobre una aplicación de software diseñada e implementada por ellos mismos, y aplican las actividades de gestión abordadas en esta asignatura. Se utiliza un repositorio (Github), los docentes siguen y evalúan individualmente el avance de cada equipo. Los equipos comparten presentaciones parciales del avance de su proyecto con la presencia de todos los estudiantes y docentes. Al final, cada equipo realiza una presentación final de su proyecto–taller como condición para la regularidad de la asignatura.

**EVALUACIÓN FINAL: “Regular”:** el estudiante que regulariza la asignatura debe rendir un examen final. oral o escrito, que incluye todo el contenido del programa de la asignatura.

**“Libre”:** El estudiante posea la condición de “Libre” en la asignatura, podrá rendir un examen final libre. En este caso el estudiante se deberá contactar con el docente responsable de la asignatura a fin de coordinar la realización de un proyecto, para posteriormente rendir un examen práctico y un examen teórico.

**Firma Profesor/a Responsable**

Mg. Marcela Daniele