

**INGENIERÍA DE SOFTWARE II**

Año 2019

**Carrera/Plan:**Licenciatura en Sistemas, Planes  
2003-2007-2012-2015Licenciatura en Informática, Planes  
2003-2007-2012-2015Analista Programador Universitario,  
Planes 2007-2015

Analista en TIC Plan 2017

**Año:** 3º**Régimen de Cursada:** Semestral**Carácter:** Obligatoria**Correlativas:** Ingeniería de Software I**Profesores:** Lic. Patricia Pesado

Mg. Silvia Esponda

Mg. Alejandro González

Lic. Marcos Boracchia

**Hs. semanales:** 6 hs.**OBJETIVOS GENERALES**

Continuar con los temas desarrollados en Ingeniería de Software I, a partir del diseño de sistemas de software. Introducir los conceptos de re-ingeniería e ingeniería inversa. Estudiar los temas de gestión, planificación y evaluación de proyectos de software, incluyendo el análisis de riesgo. El alumno deberá desarrollar sistemas concretos utilizando las metodologías/herramientas estudiadas.

**COMPETENCIAS**

- CGS1- Desempeñarse de manera efectiva en equipos de trabajo, con capacidad para organizarlos y liderarlos.
- CGS2- Comunicarse con efectividad en forma oral y escrita.
- CGS4- Aprender en forma continua y autónoma, con capacidad de planificar este aprendizaje.
- CGS6- Capacidad para interpretar la evolución de la Informática con una visión de las tendencias tecnológicas futuras.
- CGT1- Identificar, formular y resolver problemas de Informática.
- CGT3- Gestionar, planificar, ejecutar y controlar proyectos de Informática
- CGT8 Capacidad de interpretación y resolución de problemas multidisciplinares, desde los conocimientos de la disciplina informática.
- CGT9 Capacidad de organización de equipos de trabajo en proyectos de Informática, definiendo los roles en los mismos.
- LI - CE4 – Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real, especificación formal de los mismos, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software/sistemas de información que se ejecuten sobre equipos de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.
- LI - CE6 – Controlar las normas de calidad en el software o software integrado a otros componentes. Capacidad de evaluación de performance de sistemas de software y sistemas que integren hardware y software.

- 
- LI - CE7 – Planificar, dirigir, realizar y/o evaluar proyectos de sistemas de seguridad en el almacenamiento y procesamiento de la Información. Especificación, diseño, desarrollo, implementación y mantenimiento de los componentes de seguridad de información embebidos en los sistemas físicos y en los sistemas de software de aplicación. Establecimiento y control de metodología de procesamiento de datos que mejoren la seguridad y privacidad incluyendo datawarehousing.
  - LI- CE8 – Efectuar las tareas de Auditorías de los Sistemas Informáticos. Realizar arbitrajes, peritajes y tasaciones relacionados con los Sistemas Informáticos.
  - LS - CE1 – Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real. Especificación formal, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software que se ejecuten sobre sistemas de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.
  - LS - CE5 – Establecer métricas y normas de calidad y seguridad de software, contralando las mismas a fin de tener un producto industrial que respete las normas nacionales e internacionales. Control de la especificación formal del producto, del proceso de diseño, desarrollo, implementación y mantenimiento. Establecimiento de métricas de validación y certificación de calidad. Capacidad de evaluación de performance de sistemas de software y sistemas que integren hardware y software.
  - LS - CE6 – Planificar, dirigir, realizar y/o evaluar los sistemas de seguridad en el almacenamiento y procesamiento de la información. Realizar la especificación, diseño, desarrollo, implementación y mantenimiento de los componentes de seguridad de información embebidos en los sistemas físicos y en los sistemas de software aplicados. Establecer y controlar las metodologías de procesamiento de datos orientadas a seguridad, incluyendo data-warehousing.
  - LS - CE7 – Efectuar las tareas de auditoría de los sistemas informáticos. Realizar arbitrajes, pericias y tasaciones relacionados con los sistemas informáticos

### **CONTENIDOS MINIMOS (de acuerdo al Plan de Estudios)**

- Diseño e Implementación.
- Verificación y validación.
- Mantenimiento.
- Interacción hombre-máquina.
- Reingeniería e ingeniería inversa.
- Gestión de proyectos. Planificación. Métricas.
- Estimaciones. Análisis y gestión del riesgo.
- Conceptos de Auditoría y Peritaje.

---

## **PROGRAMA ANALÍTICO**

### **1. Gestión de Proyectos**

- Conceptos. El problema de las 4 “P” (personal, producto, proceso, proyecto). Actividades de gestión, planificación del proyecto, hitos y entregas. El plan de proyecto.
- Métricas y Estimaciones.
  - Clasificación de las métricas. Métricas del proceso y del proyecto. Métricas orientadas al tamaño, a la función, a casos de uso. Recopilación, cálculo y evaluación de métricas.
  - Estimación de proyectos. Técnicas de descomposición. Modelos empíricos (COCOMO). Decisión de desarrollar-comprar.
- Planificación Temporal: calendarización del proyecto, distribución del esfuerzo, redes de tareas, seguimiento de la planificación. Métodos PERT, Gantt.
- Planificación Organizativa: del equipo y del proyecto.
- Gestión del Riesgo: identificación de riesgos, proyección, impacto, reducción, supervisión y gestión. Planes de contingencia. El plan de RSGR.
- Gestión de la configuración del software: Línea base, gestión del cambio, control de versiones, auditoría.

### **2. Diseño**

- Conceptos. Abstracción, arquitectura, patrones, modularidad, ocultamiento de la información, independencia funcional, cohesión, acoplamiento, refinamiento.
- El modelo de diseño: diseño de datos, diseño arquitectónico, diseño de interfaz, diseño al nivel de componentes.
- Diseño Arquitectónico.
  - Organización del sistema: modelo de repositorio, modelo cliente-servidor, Modelo de capas. Arquitecturas de Sistemas Distribuidos: multiprocesador, c-s, objetos distribuidos, interorganizacional (peer-to-peer, sistemas orientados a servicios).
  - Descomposición modular: orientada a objetos, orientada a flujos de funciones.
  - Control: centralizado, dirigido por eventos.
- Diseño de interfaces de usuario: interacción del usuario, presentación de la información, análisis del usuario, prototipo de la interfaz, evaluación de la interfaz. Concepto de Diseño de Experiencias de Usuario
- Diseño a nivel de componentes: notaciones gráficas, notaciones tabulares, lenguajes de diseño.
- Características de un buen diseño. Técnicas para la mejora del diseño. Evaluación y validación del diseño. Documentando el diseño.

### **3. Implementación**

- Estándares de programación y procedimientos
- Pautas para la programación
- Documentación

---

#### 4. Verificación y Validación

- Técnicas de Prueba
  - Pruebas de Caja blanca: camino básico, bucles.
  - Pruebas de Caja negra: partición equivalente, análisis de valores límites.
- Estrategias de Prueba
  - Defectos y fallas. Planificación. Diseño de casos de prueba. Resultados. Documentación de las pruebas. Automatización.
  - Pruebas de unidad (arquitecturas convencionales y arquitecturas orientadas a objetos)
  - Pruebas de integración (arquitecturas convencionales y arquitecturas orientadas a objetos)
  - Pruebas de validación: alfa y beta.
  - Pruebas del sistema: de recuperación, de seguridad, de resistencia, de desempeño.
  - Pruebas de regresión.
  - La depuración: proceso, estrategia, corrección del error.

#### 5. Entrega

- Entrenamiento
- Documentación

#### 6. Mantenimiento

- Evolución del software. Tipos de mantenimiento: correctivo, adaptativo, perfectivo, preventivo.
- Sistemas heredados.
- Métricas, técnicas y herramientas para el mantenimiento.
- Rejuvenecimiento del software: redocumentación, reestructuración, ingeniería inversa, reingeniería.

#### 7. Auditoría y Peritaje

- Conceptos
- Objetivos
- Planeamiento de Auditoría

#### **BIBLIOGRAFÍA OBLIGATORIA**

- ✓ Ingeniería de Software. 9na Edición. Ian Sommerville. Pearson. 2011.
- ✓ Software Engineering: Theory and Practice. 4th Edition. Shari Pfleeger. Prentice Hall. 2010s. (Edición en castellano: Ingeniería de Software. Teoría y Práctica. Shari Pfleeger. Pearson Education. 2002)



- 
- ✓ Ingeniería de Software. Un enfoque práctico. 7ma Edición. Roger Pressman. McGraw-Hill. 2010.
  - ✓ Auditoria en Informática 2da Edición. José Antonio Echenique García. McGraw-Hill. 2001.

### **BIBLIOGRAFÍA COMPLEMENTARIA**

- ✓ Systems Analysis and Design, 9/E. Kendall & Kendall. Pearson. 2013. (Edición en castellano: Análisis y diseño de sistemas. 8va Edición. Kendall & Kendall. Pearson. 2011)
- ✓ Análisis de Sistemas. Diseño y Métodos. 7ma Edición. Whitten y Bentley. 2008.

---

## **METODOLOGÍA DE ENSEÑANZA**

El curso consta de clases teóricas, explicaciones de práctica y clases prácticas. La asignatura utiliza la plataforma IDEAS para interactuar con los alumnos del curso. Durante la cursada, grupos de alumnos desarrollan un proyecto aplicable a un dominio particular, que es monitoreado por un docente de la cátedra a través de la plataforma y las consultas en las clases prácticas. Los proyectos tienen estipulados un conjunto de entregas y reentregas y un coloquio integrador fijado en el calendario.

La cátedra organiza actividades por equipos de trabajo, con un número variable de alumnos entre 3 y 4 los cuales tienen roles específicos. En la evaluación del trabajo en equipo, hay presentaciones donde se responden aspectos generales de las tareas asignadas y respuestas/explicaciones individuales que permiten calificar diferentes aptitudes de los miembros del equipo (conocimientos / modo de expresarse / predisposición al trabajo colaborativo).

La cátedra evalúa a los alumnos en coloquios en los cuales deben exponer la tarea realizada en forma individual y donde el docente evalúa no sólo los conocimientos sino la claridad de la presentación, su organización y la forma de expresión. Esto se refleja en planillas escritas que conforman documentación de evaluación del coloquio.

La cátedra organiza actividades planificadas para los alumnos, en los que se les propone investigar el dominio del proyecto presentado, buscando posibles alternativas de solución del problema, teniendo en cuenta la tecnología existente y la evolución de la misma; y generando la documentación necesaria para llevar a cabo la implementación del proyecto. Presentando el trabajo realizado en un coloquio oral.

## **EVALUACIÓN**

Los alumnos obtienen la cursada aprobando las entregas del proyecto y un coloquio integrador.

Para la aprobación final de la asignatura los alumnos tienen dos posibilidades:

### **Alumnos por promoción:**

Deben concurrir al 80% de las clases teóricas.

Deben rendir y aprobar dos evaluaciones teóricas con nota 6 o superior (cada una de ellas con una recuperación).

Deben rendir y aprobar un coloquio.

### **Alumnos regulares:**

Deben rendir y aprobar un examen final.

## CRONOGRAMA DE CLASES Y EVALUACIONES

Semana	Teoría	Práctica
11-mar	Presentación de la materia Repaso de Elicitación Requerimientos Documentos de Especificación de Sistema (1362) y de Requerimientos (830)	Consulta Entrega 1
18-mar	Planificación Temporal -GCS	Consulta Entrega 1
25-mar	Riesgos	Consulta Entrega 1
1-abr	Interfaces	Consulta Entrega 1
8-abr	Métricas	Entrega 1 + Consulta Entrega 2
15-abr		Consulta Entrega 2
22-abr	Diseño	Entrega 2 + Consulta Entrega 3
29-abr	Diseño Arquitectónico	Consulta Entrega 3
6-may	Consulta	Entrega 3 + Consulta Entrega 4
13-may		Consulta Entrega 4
20-may	Pruebas	Entrega + Planificación Sprint 1
27-may	Pruebas	Scrum diario 1
3-jun		Scrum diario 1
10-jun	Mantenimiento	Planificación Sprint 2
17-jun		Scrum diario 2
23-jun	Consulta	Scrum diario 2
30-jun		Planificación Sprint 3
1-jul	Consulta	Scrum diario 3
8-jul		Scrum diario 3
15-jul	Receso inercial	
22-jul	Receso inercial	

Evaluaciones Previstas	Semana
Examen Teórico 1	6-may
Demo 1	27-may
Recup Examen Teórico 1	10-jun
Demo 2	17-jun
Examen Teórico 2	24-jun
Demo 3	8-jul
Recup Examen Teórico 2	15-jul
Coloquio integrador + Manual de usuario	29-jul



**Contacto de la cátedra (mail, sitio WEB, plataforma virtual de gestión de cursos):**

Los alumnos pueden enviar consultas a la cuenta [is2@info.unlp.edu.ar](mailto:is2@info.unlp.edu.ar).

Se utiliza como plataforma virtual la plataforma provista por la Universidad: IDEAS (mensajería, página y gestión del curso) y Blogs (<http://blogs.unlp.edu.ar/ingenieria2/>).

Firma del/los profesor/es