



Programa de la asignatura: ANÁLISIS Y DISEÑO DE SISTEMAS		Código: IF010
Departamento: Informática	Sede: Esquel	

Profesor Responsable: YAÑEZ, PATRICIA SILVANA DEL ROSA

Periodo de cursado: Primer Cuatrimestre

Carga Horaria:

Total	Sem. Teoría	Total Teoría	Sem. Prácticos	Total Práctica	Sem. Teórico-Práctico	Total Teórico-Práctico
135	4	60.00	5	75.00	0	0.00

Horarios de cursada:

Día	Horario de cursada	Tipo de cursada
Lunes	15:00 a 18:00	Teoría
Martes	15:00 a 18:00	Teoría
Miércoles	15:00 a 18:00	Práctica

Asignaturas Correlativas:

Correlativa	Para la/s carrera/s
IF007 - BASE DE DATOS I	APU, Lic. en Informática
IF004 - SISTEMAS Y ORGANIZACIONES	APU, Lic. en Informática

I. Objetivos de la Asignatura:

Comprender los conceptos básicos de la Ingeniería de Software, reconociendo las diferencias entre metodologías, aplicando los métodos pertinentes e identificando las buenas prácticas de la disciplina, relacionadas a las etapas de requerimientos, análisis y diseño de sistemas, con el fin de desarrollar soluciones informáticas eficientes, siguiendo un proceso definido, integrando habilidades y conocimientos adquiridos en las otras materias de la carrera.

II.1 - Contenidos mínimos:

- * Conceptos y objetivos de la Ingeniería de Software.
- * El proceso de software. Modelos de ciclos de vida del software.
- * Conceptos de ingeniería de requerimientos.
- * Especificación de requerimientos.
- * Diseño de sistemas. Diseño centrado en el usuario.
- * Pruebas de software.



Programa de la asignatura: ANÁLISIS Y DISEÑO DE SISTEMAS	Código: IF010
Departamento: Informática	Sede: Esquel

II.2 Programa Analítico:

1 - Introducción a la Ingeniería de Software:

Costo y complejidad del Software. Problemas en el desarrollo de software. Cuerpo del conocimiento de la Ingeniería de Software. Perfil profesional: roles, habilidades y ética. La esencia de la práctica de la Ingeniería de Software. Personas, Proyecto, Producto y Proceso, en el desarrollo de software. Características y atributos de calidad del producto Software. Tipos de Desarrollo. Dominios de aplicación. Tipos de proyectos.

2 - El proceso de software:

Principios de la Ingeniería de Software. Capas de la Ingeniería de Software. Métodos y metodologías. Visión general de la IS: Fases genéricas. Modelo de proceso y ciclo de vida. Diferentes modelos de proceso. El proceso unificado: EDUP. Metodologías ágiles: Scrum.

3 - Ingeniería de Requerimientos:

Conceptos y terminología. El Proceso de Requerimientos. Captura, análisis y especificación de los Requerimientos. Determinación de requisitos. Características. Clasificación de los requerimientos. Calidad de los requisitos. El plan de relevamiento. Elicitación de requerimientos. Conceptos y técnicas de Elicitación. Criterios de Aceptación. El modelo del negocio. Documentación de requerimientos: Especificación de requerimientos del software. Estándar IEEE/ANSI 830/1998. Gestión de los requerimientos. Proceso Volere.

4 - Modelado de requerimientos:

Principios y Metodologías del análisis. Abstracción y modelado de sistemas. Técnicas de modelado. El Lenguaje Unificado de Modelado (UML). Diagrama de contexto. Casos de uso de negocio. Casos de uso de producto. Modelo de Casos de uso. Escenarios. Modelo de dominio. Modelado Dinámico. Diagramas de Colaboración. Diagramas de Actividades. Diagramas de secuencia. Contratos y operaciones. Diagramas de estado. Documentación.

5 - Diseño:

El proceso de Diseño. Principios y conceptos del diseño. Arquitectura del Software. Estilos arquitectónicos. El Diseño arquitectural y detallado. Fundamentación. Del modelo de dominio al diseño de clases del sistema. Especificación de restricciones: OCL. Patrones de diseño. Diseño centrado en el usuario, usabilidad. Diseño de la interfaz hombre máquina. El proceso de diseño de centrado en el usuario. Normas de diseño de interfaces visuales. Cualidades de una interfaz icónica.

6 - Validación y verificación del software:

Validación, verificación y pruebas: conceptos y diferencias. Técnicas. Metodologías de desarrollo y pruebas. Prueba vs. Depuración. Tipos de pruebas. Estrategias. Pruebas de caja blanca. Pruebas de caja negra. Diseño de casos de prueba. Inspecciones del software. Prueba de unidad. Prueba de integración. Pruebas de sistema. Pruebas de aceptación. Estrategias de prueba de software. Estructura del plan de pruebas.

III. Descripción de Actividades Teóricas y Prácticas:

Semana	Descripción
1	Sondeo de intereses de los estudiantes. Presentación de la asignatura. Dictado de contenidos teóricos. Lectura de la bibliografía sugerida y resolución de las actividades propuestas. Actividad Nro. 1. Guía de estudio.. Unidades a desarrollar: 1
2	Dictado de contenidos. Actividad Nro. 2. Guía de estudio. Practica de Laboratorio Nro. 1 Conociendo EDUP. Trabajo práctico Nro.1 El proceso de Software.. Unidades a desarrollar: 2
3	Dictado de contenidos. Actividad Nro. 3. Guía de estudio. Autoevaluación de conceptos unidades 1 y 2. Practica de Laboratorio Nro. 2 GIT. Unidades a desarrollar: 3
4	Dictado de contenidos teóricos. Trabajo práctico Nro.2 Plan de elicitación.. Unidades a desarrollar: 3



Programa de la asignatura: ANÁLISIS Y DISEÑO DE SISTEMAS		Código: IF010
Departamento: Informática	Sede: Esquel	

Semana	Descripción
5	Contenidos de la unidad. Practica de Laboratorio Nro. 3 Herramienta de Modelado. Actividad Nro. 4. Guía de estudio.. Unidades a desarrollar: 3
6	Contenidos teóricos. Trabajo práctico Nro.3 Modelo de Negocio.. Unidades a desarrollar: 4
7	Contenidos teóricos de la unidad. Autoevaluación unidad 3.. Unidades a desarrollar: 4
8	Repaso de conceptos y dudas, actividades adicionales. Primer evaluación parcial..
9	Conceptos de arquitectura. Actividad Nro. 5. Guía de estudio . Trabajo práctico Nro.4 Arquitecturas del sistema.. Unidades a desarrollar: 1,2,3,4
10	Coceptos de diseño. Trabajo práctico Nro.5 Modelo de diseño. Autoevaluaci{on unidad 4 y 5.. Unidades a desarrollar: 5
11	Diseño centrado en el usuario. Trabajo práctico Nro.6 Interfaces de usuario.. Unidades a desarrollar: 5
12	Conceptos de validación y verificación. Actividad Nro. 6. Guía de estudio. Trabajo práctico Nro.7 Actividades SQA.. Unidades a desarrollar: 5
13	Conceptos de pruebas. Test de unidad. Trabajo Final Integrador. Autoevaluaci{on unidad 6 y 7.. Unidades a desarrollar: 6
14	Consultas y dudas. Segunda evaluación parcial.. Unidades a desarrollar: 6
15	Defensa del trabajo final. Recuperatorio final.. Unidades a desarrollar: 5,6

IV. Bibliografía:

Título	Autor	Editorial	Año Publicación
Analisis y diseño OO con UML y el proceso Unificado. Básica para: 4,5	Schach, S..	McGrawHill.	2005.
Complementaria para: .			
Applying UML and patterns Básica para: 4,5	Larman, C..	Prentice Hall.	2004.
Complementaria para: .			
Casos de Uso 2.0 La guia definitiva. Básica para: 3	Jacobson, I..	Ivar Jacobson Intern.	2013.
Complementaria para: 4,5.			
El proceso unificado de desarrollo de software. Básica para: 2	Rumbaugh, J., Jacobson, I. Booch, G..	Addison Wesley.	2000.
Complementaria para: 1,2,3,4,5,6.			
Ingeniería de Software Básica para: 1,2	Sommerville I..	Pearson.	2011.
Complementaria para: 3.			



Programa de la asignatura: ANÁLISIS Y DISEÑO DE SISTEMAS		Código: IF010
Departamento: Informática	Sede: Esquel	

Título	Autor	Editorial	Año Publicación
Ingeniería del Software Un enfoque práctico. Básica para: 1	Pressman, R..	MacGrawHill..	2010. Complementaria para: 1,2,3,4,5,6.
Introducción al Proceso Software Personal. Básica para:	Humphrey, W..	Addison Wesley.	2001. Complementaria para: 2.
Large Scale Software Architecture. A Practical Guide using UML. Básica para:	Garland ,J. Richard, A..	John Wiley and sons.	1999. Complementaria para: 5.
Mastering the requirements process. Básica para: 2	Robertson, S. Robertson J..	Adison Wesley Profes.	2006. Complementaria para: .
Patrones de Diseño Básica para: 5	Gamma, Helm, Johnson y vlissides.	AddisonWesley.	1995. Complementaria para: .
Requirements engineering. Básica para:	Hull E., Jackson K., Dick J..	Springer.	2011. Complementaria para: 2,3.
Software engineering theory and practice. Básica para:	Pfleeger, S., Atlee, J..	Pearson.	2010. Complementaria para: 1,4,5,6.
Software Requirements. Básica para: 3	Wiegers K., Beatty J..	Microsoft Press.	2013. Complementaria para: 2,4.
The Art of Software Testing. 2 edition. Básica para: 6	Myers G..	John Wiley and Sons.	2004. Complementaria para: .

IV.1. Sitios web recomendados:

Sitios

Url: <https://refactoring.guru/es/design-patterns>

Descripción: Presenta de manera divertida los patrones y anti patrones de diseño de software.

Complemento para unidades: 5

Url: <https://www.agilealliance.org/>

Descripción: Los practicantes ágiles comparten regularmente sus conocimientos, experiencias e ideas a través de charlas prácticas y talleres.

Complemento para unidades: 1,2,3,4,5,6

Url: <https://www.martinfowler.com/>

Descripción: Presenta artículos relevantes sobre patrones y prácticas que facilitan la creación de software útil.

Complemento para unidades: 1,2,3,4,5,6

Url: <https://www.uml.org/>

Descripción: Página oficial UML.

Complemento para unidades: 4,5



Programa de la asignatura: ANÁLISIS Y DISEÑO DE SISTEMAS		Código: IF010
Departamento: Informática	Sede: Esquel	

V - Metodologías de Enseñanza:

En el dictado de la asignatura se fomenta el aprendizaje activo, utilizando mayormente métodos de enseñanza constructivos. El profesor es un mediador que brinda las herramientas que le permitan al alumno la asimilación de los conocimientos, a través de un intercambio dialéctico entre los saberes de ambos, de tal forma que se pueda llegar a un aprendizaje significativo. Se transforma el rol docente de manera en conseguir que el alumno aprenda a aprender en lugar de seguir siendo meros transmisores de conocimiento. Para esto se propician espacios de retroalimentación, de discusión, de intercambio de ideas y de interacción activa entre los alumnos para que emerja el conflicto cognitivo, con el objetivo de estimular la capacidad de análisis, de autogestión del conocimiento y de fijación de criterios de manera que los nuevos conocimientos se integren a sus estructuras previas. Se promueven las capacidades para el trabajo colaborativo, la autonomía y la confianza en las propias posibilidades de aprender.

En este modelo se requiere que el rol del estudiante asuma un rol mucho más activo que lo haga partícipe del proceso formativo, teniendo que:

- Convertirse en el responsable de su aprendizaje, debiendo gestionar y controlar la forma en que se produce.
- Desarrollar un espíritu crítico y una actitud abierta ante los cambios científico-técnicos de su especialidad, con curiosidad intelectual y rigor científico.
- Fomentar actitudes y adquirir técnicas para trabajar en equipo con eficacia.
- Estimular el perfeccionamiento profesional y la formación continua a lo largo de la vida.

Atendiendo dichos aspectos, algunas de las metodologías que se utilizarán son:

- * Aprendizaje cooperativo
- * Aprendizaje orientado a proyectos
- * Aprendizaje inverso o flippedlearning
- * Contrato de aprendizaje
- * Aprendizaje basado en problemas
- * Investigación individual
- * Exposición / Lección magistral (profesor-alumnos; alumnos-alumnos)
- * Estudio de casos
- * Simulación y juego (Gamificación)
- * Técnicas de evaluación de baja y de alta repercusión

El profesor seleccionará un método o una combinación de métodos de enseñanza de acuerdo al perfil de los estudiantes y los objetivos de cada clase.

VI.1 Condiciones para la aprobación del cursado de la asignatura:

Para la aprobación del cursado de la asignatura se requiere:

- Asistencia al 80% de las clases prácticas.
- Aprobación de los trabajos prácticos obligatorios con un mínimo de seis (6) puntos.
- Aprobación de dos parciales prácticos con un mínimo de seis (6) puntos o sus respectivas instancias recuperatorias.
- Aprobación del trabajo final integrador de conocimientos con un mínimo de seis (6) puntos.

La calificación de cursado se determinará por promedio de las calificaciones obtenidas en la correspondiente instancia (ya sea inicial o recuperatoria) de aprobación de los parciales prácticos, los trabajos prácticos y el trabajo integrador.

CONDICIONES PARA ACCEDER A NUEVAS INSTANCIAS DE EVALUACIÓN EN EL CUATRIMESTRE SIGUIENTE AL CURSADO EN CASO DE QUE NO SE HAYA ALCANZADO LA REGULARIDAD (Dis. CAFI 001/01) El alumno se inscribirá en la Secretaría Administrativa de la Facultad antes del comienzo del cuatrimestre de cursado, cumplimentando los siguientes requisitos:



Programa de la asignatura:
ANÁLISIS Y DISEÑO DE SISTEMAS

Código:
IF010

Departamento: Informática

Sede: Esquel

- 1) Haber aprobado durante el cuatrimestre de cursado de la asignatura uno de los dos parciales prácticos, ya sea en la primera instancia o en recuperación.
- 2) Haberse presentado durante el cuatrimestre de cursado en ambos parciales a alguna de sus instancias.
- 3) Presentar carpeta de trabajos prácticos aprobada por parte de la cátedra, previamente a la inscripción.

VI.2 Condiciones para la aprobación de la asignatura:

Para la aprobación por promoción directa se requiere, además de las condiciones exigidas para la aprobación de la cursada:

- a) Aprobación de los trabajos prácticos obligatorio con un mínimo de siete (7) puntos.
- b) Aprobación de dos parciales teórico prácticos con un mínimo de ocho (8) puntos (sin haber accedido a instancias recuperatorias).
- c) Aprobación del trabajo final integrador de conocimientos con mínimo de siete (7) puntos.

En caso de no haber accedido a la promoción directa, el alumno una vez aprobada la cursada de la materia, deberá rendir examen final, con una Nota mínima de 4 pts.

Vigencia de este programa

Año	Firma	Profesor/a Responsable
2022		YAÑEZ, PATRICIA SILVANA DEL ROSA

Visado de este programa

Decano/a	Sec.Académico/a Facultad	Jefe de Departamento
Mg. Maria Elizabeth Flores Decana Facultad de Ingeniería	Lic. Miryam Acosta Secretaria Académica Facultad de Ingeniería	
Fecha:	Fecha:	Fecha: