

UNIVERSIDAD NACIONAL DEL SUR BAHÍA BLANCA		1 6
DEPARTAMENTO DE CIENCIAS E INGENIERÍA DE LA COMPUTACIÓN		
ANÁLISIS Y DISEÑO DE SISTEMAS	CÓDIGO: 5534	
	ÁREA N°: III	

CARRERAS Licenciatura en Ciencias de la Computación					
PROFESOR RESPONSABLE: Mg. MARÍA MERCEDES VITTURINI – Profesor Adjunto con Dedicación Semi-Exclusiva					
CARGA HORARIA	Teoría 64	Práctica 54	Laboratorio 10	CANTIDAD DE SEMANAS	16
CORRELATIVAS					
PARA CURSAR LA MATERIA			PARA APROBAR LA MATERIA		
APROBADAS	CURSADAS	APROBADAS	CURSADAS		
-Estructuras de Datos	-Tecnología de Programación	-Tecnología de Programación			
DESCRIPCIÓN					
Análisis y Diseño de Sistemas es la primera materia del área de Desarrollo de Sistemas que cursa el alumno de la Licenciatura en Ciencias de la Computación. En forma coordinada con las otras materias del área de Desarrollo de Sistemas busca formar a los alumnos en los fundamentos de Ingeniería de Software. Se espera que los egresados de la Licenciatura en Ciencias de la Computación tengan las habilidades necesarias para planificar, auditar, dirigir y administrar equipos de desarrollo de software de mediana y gran escala. En la materia Análisis y Diseño de Sistemas se plantean los principios y calidades de la ingeniería de software, tipos de sistemas y modelos de procesos de desarrollo. Los contenidos enfatizan temas referentes a las actividades iniciales del proceso de construcción de software: requerimientos, análisis y especificación, y principios generales de diseño.					
COMPETENCIAS PREVIAS Y A DESARROLLAR					
El alumno que inicia el cursado de Análisis y Diseño de Sistemas debe poseer conocimientos y capacidades para diseñar e implementar algoritmos que utilicen estructuras de datos, poseer conceptos generales del proceso de desarrollo de software y debe haber adquirido experiencia en el modelado estático de pequeñas aplicaciones. Son objetivos de la materia que los alumnos desarrollen capacidades para:					

- 1) Ejercitar el poder de abstracción a fin de desarrollar habilidades de modelización de sistemas,
- 2) Comprender las vistas de un sistema y determinar la importancia de cada una de ellas dependiendo del tipo de sistema y las necesidades de comunicación,
- 3) Aplicar las distintas técnicas que le permiten realizar la extracción y especificación y validación de requerimientos,
- 4) Aplicar conceptos de diseño que le permitan desarrollar modelos de diseño de calidad,
- 5) Evaluar las distintas metodologías de desarrollo de software y poder determinar cuándo es conveniente aplicar cada una de ellas,
- 6) Evaluar y usar herramientas de software para desarrollar sistemas,
- 7) Trabajar en equipo y en forma individual de manera eficiente,
- 8) Realizar presentaciones orales e informes de trabajo.

METODOLOGÍA DE ENSEÑANZA

Las materias del área de Desarrollo de Software se caracterizan porque la **construcción** de las soluciones a los problemas planteados requiere cierto nivel de **creatividad**. Como en toda actividad creativa, en la **valoración** de las soluciones, aún cuando se apliquen criterios establecidos, siempre existe un margen de **subjetividad**. El desafío en el diseño de las clases teóricas y prácticas consiste justamente en establecer pautas que guíen el proceso constructivo de un modelo o una vista y criterios que permitan evaluar y comparar soluciones.

Las clases teóricas están enfocadas a presentar conceptos generales de Ingeniería de Software y problemas que permitan ilustrar aspectos referidos al modelado y especificación de sistemas. Se enfatiza sobre el poder de abstracción y análisis del problema, comunicación y evaluación de soluciones alternativas (generalidad, representatividad del problema, etc.), buscando desarrollar en el alumno la capacidad crítica sobre la solución propuesta. Se analizan interpretaciones sintácticas y semánticas de un modelo.

Las actividades prácticas están diseñadas para acompañar el desarrollo de los contenidos teóricos con ejercicios y problemas prácticos que refuerzan las estrategias de modelado para las distintas vistas de una misma situación: del usuario, de datos, funcional y dinámica. Las soluciones a los ejercicios son supervisadas en consulta y aleatoriamente resueltas en clases prácticas. Se incentiva la discusión de soluciones alternativas entre pares y el uso de herramientas de modelado. Se trabaja también con ejercicios en el laboratorio.

Además se exige la presentación un proyecto con entregas incrementales. Como proyecto se trabaja con un problema de dificultad media vinculado con la realidad. Consiste en la presentación de informe que incluya el diseño de modelos consistentes para las distintas vistas de un mismo problema. Para el proyecto los alumnos se organizan en grupos de entre 3 y 4 alumnos. Con el trabajo de proyecto los alumnos experimentan el uso de herramientas de modelado y entornos colaborativos de trabajo y confección de informes de

trabajo. Además se fortalece el trabajo en equipo, respetar estándares de trabajo y vocabulario normalizado para valorizar la comunicación.

MECANISMO DE EVALUACIÓN

Evaluación de Trabajos Prácticos

La condición de cursado de la materia consiste en aprobar cada uno de los dos parciales escritos teóricos-prácticos o el recuperatorio final. También se evalúan las entregas obligatorias del proyecto y la participación en los laboratorios. Las actividades y proyectos con entrega deben estar aprobadas. Existe una posibilidad de reentrega para cada entrega.

Evaluación Final

El examen final regular consta de un examen escrito teórico-práctico. El examen final libre consta de dos instancias escritas teórico-prácticas y un proyecto. Se brinda a los alumnos la posibilidad de rendir en las mesas del año académico de cursado acotando los temas a evaluar y promediando el examen con la nota general de cursado.

PROGRAMA SINTÉTICO

1. Sistemas e Ingeniería de Software. Teoría general de sistemas. Sistemas de información. Clasificación.
2. Calidad del software: del producto y del proceso. El Proceso de software. Ciclos de vida del software.
3. Ingeniería de requerimientos. Actividades: extracción, análisis, organización, validación y verificación. Documentación: modelos de negocio.
4. Modelos Estructurales. Modelado y calidad de datos. Modelo de Clases. Modelo Entidad Relación. Diccionario de Datos. Diseño de datos. Modelo Relacional.
5. Modelos Funcional y Dinámico. Modelado Funcional, Métodos de Especificación Funcional. Diagrama de Casos de Uso. Diagrama de Flujos de Datos. Modelos de comportamiento. Modelos de interacción. Modelos de estado
6. Métodos formales. Especificaciones algebraicas y orientadas a modelos. Descripción formal en modelos.
7. Conceptos de diseño. Abstracción, modularidad, patrones, ocultamiento de información. Decisiones de diseño.
8. Arquitecturas de Sistemas. Arquitectura y Diseño.

PROGRAMA ANALÍTICO

1) Sistemas e Ingeniería de Software.

Definición de sistemas. Teoría general de sistemas. Principios de sistemas. Sistemas de Software. Clasificación de Sistemas: batch, en línea, de tiempo real, embebidos, de soporte de decisión, inteligentes. Sistemas de información. Ingeniería de software. Evolución de la ingeniería de software. Principios de la ingeniería de software. Rigurosidad y formalismo. Separación de intereses. Modularización. Abstracción. Participantes. Tipos de Usuario. Perfiles. Equipo de Desarrollo. Roles.

2) El proceso de software y ciclos de Vida

Producto y Proceso. Calidad del software. Calidad interna y externa. Calidad del producto y del proceso. Correctitud. Confiabilidad. Robustez. Performance. Amigabilidad. Verificabilidad. Mantenibilidad. Reparabilidad. Evolutividad. Reusabilidad. Portabilidad. Comprensibilidad. Interoperabilidad. Productividad. Puntualidad. El proceso de desarrollo de software. Ciclo de vida del software. Actividades. Modelos de proceso: cascada, prototipos, iterativo incremental, transformacional, espiral. Desarrollo Rápido de Aplicaciones. Personalización del modelo de proceso. Criterios de Selección.

3) Ingeniería de requerimientos

Definición. Clasificación de requerimientos. Requerimientos funcionales y no funcionales. Requerimientos de usuario y de sistema. Ingeniería de requerimientos. Etapas: extracción, análisis, organización, validación y verificación. Técnicas de elicitación. Técnicas de documentación. Estudio de factibilidad. Modelos del Negocio. Diagrama de contexto. Modelo de casos de uso: casos de uso y actores. Modelo de conceptos del negocio. Escenarios y diagramas de secuencia del sistema. Trazabilidad de requerimientos.

4) Modelos Estructurales.

Modelado y calidad de datos. Diagrama de clases. Clases y objetos. Atributos. Operaciones y métodos. Links y asociaciones. Atributos de link. Roles. Multiplicidad. Calificación. Generalización. Generalización por extensión y restricción. Agregación. Diagramas de objetos. Diccionario de Datos. Diseño de datos. Modelo Entidad-Relación. Entidades fuertes y débiles. Atributos. Llaves. Relaciones. Relación "es_un". Mapeo al modelo relacional. Normalización. Formas normales: primera, segunda y tercera. Balanceo entre modelos.

5) Modelo funcional y Modelo dinámico

Modelo funcional. Modelo de Casos de uso avanzado. Relaciones entre casos de uso: inclusión, extensión y herencia. Técnicas de especificación para casos de uso. Camino básico y alternativo. Condiciones pre-post. Puntos de extensión. Diagrama de flujo de datos: procesos, flujos de datos, almacenamientos y terminadores. Niveles. Descripción estructurada de procesos. Técnicas de

UNIVERSIDAD NACIONAL DEL SUR BAHÍA BLANCA		5 6
DEPARTAMENTO DE CIENCIAS E INGENIERÍA DE LA COMPUTACIÓN		
ANÁLISIS Y DISEÑO DE SISTEMAS	CÓDIGO: 5534	
	ÁREA N°: III	

descripción. Tablas de Decisión. Modelos de comportamiento. Diagramas de Interacción: secuencia y comunicación. Diagramas de Actividad. Diagramas de estado. Eventos y estados. Condiciones. Acciones. Actividades. Generalizaciones de estados y de eventos. Concurrencia. Balanceo con otros modelos del sistema.

6) Especificaciones formales

Métodos de Especificación Formal. Verificación y Validación. Grados de Formalismo. Lenguajes Formales. Especificaciones algebraicas y orientadas a modelos. Especificaciones en modelos: Object Constraint Language OCL. Invariantes. Precondiciones y postcondiciones. Inicializadores. Guardias. Visión general de las metodologías vistas: orientadas a objetos, estructuradas y formales. Comparación.

7) Conceptos de diseño de sistemas

Modularización. Diseño general y diseño detallado. Abstracción. Modularidad. Ocultamiento de información. Cohesión. Acoplamiento. Independencia funcional. Sistema, subsistemas y módulos. Patrones. Reutilización del diseño. Decisiones de diseño.

8) Arquitecturas sistemas

Diseño Arquitectónico. Vistas Arquitectónicas: modular, componentes y conectores y distribución. Requerimientos y arquitectura. Estilos arquitectónicos. Organización general del sistema. Repositorio de datos. Cliente-servidor. Máquinas abstractas (capas). Descomposición modular. Técnicas de descomposición orientadas a objetos y clases. Técnicas de descomposición funcional. Pipeline. Estilos de control. Centralizado. Basados en eventos. Tipos de aplicaciones y estilos arquitectónicos.

BIBLIOGRAFÍA

Bibliografía Básica

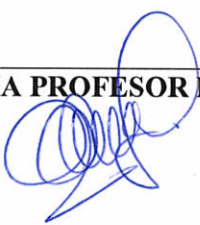
- *Software Engineering* (10th Edition) – Ian Sommerville. Pearson, 2015.
- *Fundamentals of Software Engineering*– Carlo Ghezzi, et al, 2nd. Edition Prentice Hall, 2002.
-
- Sitio oficial de UML <http://www.uml.org>
- *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design.* (2nd Edition), Craig Larman, Prentice Hall, 2002.
- *Documenting Software Architectures: Views and Beyond* – Paul Clements, Felix Bachmann, et all. Pearson Education, 2010

UNIVERSIDAD NACIONAL DEL SUR BAHÍA BLANCA		6 6
DEPARTAMENTO DE CIENCIAS E INGENIERÍA DE LA COMPUTACIÓN		
ANÁLISIS Y DISEÑO DE SISTEMAS	CÓDIGO: 5534	
	ÁREA N°: III	

- *Object-Oriented Modeling and Design with UML* (2nd Edition) – Michael Blaha and James Rumbaugh, Prentice Hall, 2005.
- *Análisis Estructurado Moderno*. Edward Yourdon, Prentice Hall, 1993.
- The Unified Modeling Language – User Guide, Reference Guide – Grady Booch et al., Addison Wesley.

Bibliografía Adicional

- A Concise Introduction to Software Engineering – Pankaj Jalote. Springer 2008
- The Unified Software Development Process – Ivar Jacobson, et al, Addison Wesley.
- Software Engineering: A Practitioner's Approach – Roger Pressman 8th Edition McGraw Hill, 2014
- Database System Concepts –Avi Silberschatz, Henry Korth, 6th Edition McGraw Hill. 2010,
- Software Engineering: Theory and Practice – Shari Lawrence Pfleeger, Prentice Hall, 2010.
- System Analysis and Design. 6th Ed. A. Dennis, B. Wixon and R. Roth. John Wiley & Sons Inc 2014.

AÑO 2018		FIRMA PROFESOR RESPONSABLE  Lic. MARIA MERCEDES VITTURINI Profesor Adjunto Dpto. Cs. e Ing. de la Computación	
VISADO			
COORDINADOR ÁREA	SECRETARIO ACADÉMICO	DIRECTOR DEPARTAMENTO	
	Dr. DIEGO C. MARTINEZ SECRETARIO ACADÉMICO DPTO. DE CS. E ING. DE LA COMP. UNIVERSIDAD NACIONAL DEL SUR	Dr. MARCELO A. FALAPPA DIRECTOR DECANO DPTO. DE CS. E ING. DE LA COMP. UNIVERSIDAD NACIONAL DEL SUR	