



<b>CARRERA</b>	<b>Tecnicatura Superior en Análisis de Sistemas con Orientación en Redes Informáticas</b>			
<b>PLAN DE ESTUDIOS</b>	<b>Resolución 2561/06</b>			
<b>DOCENTE</b>	<b>Daniel Reyes</b>			
<b>UNIDAD CURRICULAR</b>	<b>Período</b>	<b>Año</b>	<b>Régimen</b>	<b>Hs. Cátedra</b>
Análisis de Sistemas II	2024	3ro.	Cuatrimstral (1er. cuat.)	4 (cuatro)

## 1. Fundamentación

La Cátedra de Análisis de Sistemas II constituye un espacio central en la formación de los estudiantes de la Tecnicatura Superior en Análisis de Sistemas con Orientación en Redes Informáticas, ya que proporciona los conocimientos y habilidades necesarios para diseñar soluciones de software eficaces.

El diseño de soluciones software es un proceso fundamental en la fase de Requerimientos de Software, también conocida como la fase de análisis en el ciclo de vida de los Estándares de Ingeniería de Software. En esta etapa, se debe emplear un método reconocido o una combinación de métodos que se adecuen a las necesidades del proyecto, definiendo técnicas para desarrollar una solución que satisfaga los requisitos del cliente.

En este contexto, la Unidad Curricular (UC) se enfoca inicialmente en la recuperación y consolidación de conocimientos previos sobre abstracción, modelado y organización de la información, adquiridos en etapas formativas anteriores. Esta recuperación es esencial para crear un espacio de integración de diversos saberes, conceptos y tecnologías relacionados con el diseño de sistemas, con el objetivo de elaborar un Modelo del Sistema y su Arquitectura de manera eficiente.

Por consiguiente, se abordan los conceptos necesarios para formar estudiantes capaces de desempeñarse eficazmente en su campo profesional, comprendiendo y dominando las aplicaciones de la Ingeniería del Software, especialmente en el Diseño de Software. Se hace hincapié en el “Paradigma Orientado a Objetos” y su relación con la fase de Análisis del Proceso de Desarrollo de Software.

La apropiación de tales conocimientos, permitirá presentar a los alumnos conceptos nucleares del Diseño de Software, tales como la importancia del diseño en el proceso de desarrollo software, su aplicación en los distintos modelos de proceso y su adecuación en las metodologías ágiles, junto con la necesidad de aplicar un proceso para el mismo; las distintas fases del Diseño y su relación con el modelado arquitectónico del Sistema, y el rol que cumple el diseñador de sistemas a lo largo del proceso de desarrollo de software.

Además, dentro de la UC se incorporarán las habilidades de modelado y diseño que se integrarán con las capacidades adquiridas en los trayectos formativos de las cátedras “Análisis de Sistemas I” y “Bases de Datos”, completando las competencias de los alumnos para el desarrollo de sistemas.

El desarrollo de un taller de modelado, a lo largo de todo el Espacio Curricular, enfocado en el Diseño Orientado a Objetos, permitirá a los alumnos afianzar, revisar y aplicar los conceptos aprehendidos en clase.

Asimismo, y conforme con lo expuesto anteriormente, sería recomendable que los alumnos cursen la asignatura “Programación IV”, simultáneamente al cursado de presente UC, completando así las competencias necesarias para el desarrollo de software.

## 2. Objetivos

- Conocer las distintas etapas del proceso de desarrollo de software
- Dominar los elementos conceptuales necesarios para diseñar software de calidad.
- Diferenciar conceptualmente las principales metodologías para el diseño de software.
- Elaborar modelos para el diseño de sistemas de información.
- Diseñar y construir artefactos de software aplicando herramientas de soporte de diseño.
- Comprender la importancia de mantener la calidad en un producto de software.
- Valorar la importancia del diseño como punto de partida para el desarrollo de sistemas de información.
- Reconocer la importancia de utilizar la metodología de Orientada a Objetos en el desarrollo de software.

## 3. Contenidos

### **Unidad 1: Introducción al Diseño de Software (SW)**

El Diseño en el proceso de desarrollo de SW. ¿Qué es diseñar? Definición de Diseño. El Proceso de Diseño. Diseño de la Solución. Evolución del Proceso de Diseño. Criterios para un buen diseño. Principios de Diseño. Importancia del diseño de SW. Conceptos de Diseño. Diseño Orientado a Objetos. Conceptos de DOO. Clases de diseño. Clases de diseño bien formadas.

Objetivos de la fase de diseño. El modelo de diseño. Dimensiones del modelo de diseño. Elementos del modelo de diseño. Punto de vista de gestión: Diseño preliminar y Diseño detallado. Punto de vista técnico: Diseño de datos, Diseño arquitectónico, Diseño a nivel de componentes (diseño procedimental), Diseño de la interfaz. Elementos del diseño del despliegue. Estrategias para el diseño modular.

Documentación del diseño del SW. Notaciones para el diseño de SW: Notaciones estructurales, Notaciones estáticas, Notaciones dinámicas o de comportamiento: Diagramas de Secuencia, Notaciones híbridas: Diagramas de clase. Especificación de Diseño (Documento de diseño de SW).

<b>Bibliografía específica de la Unidad</b>			
<b>Autor</b>	<b>Título</b>	<b>Editorial</b>	<b>Lugar y año de edición</b>
Kendall, K. & Kendall, J.	Análisis y Diseño de Sistemas, 8va. ed.	Prentice Hall	España, 2007
Pressman, R.	Ingeniería del Software: Un Enfoque Práctico, 5ta. ed.	Mc Graw Hill	España, 2007
Sommerville, I.	Ingeniería del Software, 7ma. Ed.	Prentice Hall	España, 2008
Reyes, D.	Apuntes de Cátedra.	Del autor	Argentina, 2024

### **Unidad 2: Introducción a la Arquitectura del Software**

Fundamentos de la Arquitectura de Software. Qué es la arquitectura de software. Importancia de la arquitectura de software. Elementos del Diseño Arquitectónico. Diferencias entre Diseño y Arquitectura. Vistas Arquitectónicas. Qué son las vistas arquitectónicas. Cómo documentar las arquitecturas de un sistema.

Estilos y Patrones Arquitectónicos. Clasificación de estilos de arquitectónicos. Clasificación de patrones arquitectónicos. Patrones de diseño.

Arquitectura en Metodologías Ágiles. Principios de arquitectura ágil. Arquitectura evolutiva y emergente. Integración continua y entrega continua.

<b>Bibliografía específica de la Unidad</b>			
<b>Autor</b>	<b>Título</b>	<b>Editorial</b>	<b>Lugar y año de edición</b>
Bass, Len; Clements, P. & Kazman, R.	Software Architecture in Practice, Third Edition	Addison Wesley	EEUU, 2012
Arias, A. y Durango, A.	Arquitectura del Software	IT Campus Academy	España, 2001
Booch, G.; Rumbaugh, J. y Jacobson, I.	El Lenguaje Unificado de Modelado	Addison Wesley	España, 1999
Gamma, E.; Helm, R.; Johnson, R. y Vlissides, J.	Patrones de Diseño	Addison Wesley	España, 2002
Larman, C.	UML y Patrones. Una Introducción al Análisis y Diseño Orientado a Objetos y al Proceso Unificado	Prentice Hall	España, 2003
Pressman, R.	Ingeniería del Software: Un Enfoque Práctico, 5ta. ed.	Mc Graw Hill	España, 2007
Sommerville, I.	Ingeniería del Software, 7ma. Ed.	Prentice Hall	España, 2008
Reyes, D.	Apuntes de Cátedra.	Del autor	Argentina, 2024

### **Unidad 3: Diseño en el nivel de componentes y de la interfaz de usuario**

Diseño en el nivel de componentes. ¿Qué es un componente? Distintas visiones del diseño en el nivel de componentes: Orientada a Objetos, Tradicional y Relacionada con el Proceso. Diseño de componentes basados en clases. Principios básicos del diseño en el nivel de componentes. Lineamientos de diseño en el nivel de componentes. Cohesión. Acoplamiento. Realización del diseño en el nivel de componentes.

Diseño de la Interfaz de Usuario. Las reglas doradas para el diseño de interfaces. Análisis y Diseño de la Interfaz de usuario. Patrones de Diseño de la Interfaz de Usuario. Algunas consideraciones para el diseño de interfaz del usuario.

<b>Bibliografía específica de la Unidad</b>			
<b>Autor</b>	<b>Título</b>	<b>Editorial</b>	<b>Lugar y año de edición</b>
Bass, Len; Clements, P. & Kazman, R.	Software Architecture in Practice, Third Edition	Addison Wesley	EEUU, 2012
Booch, G.	Análisis y Diseño Orientado a Objetos, 5ta. ed.	Addison Wesley	España, 2001
Booch, G.; Rumbaugh, J. y Jacobson, I.	El Lenguaje Unificado de Modelado	Addison Wesley	España, 1999
Gamma, E.; Helm, R.; Johnson, R. y Vlissides, J.	Patrones de Diseño	Addison Wesley	España, 2002
Larman, C.	UML y Patrones. Una Introducción al Análisis y Diseño Orientado a Objetos y al Proceso Unificado	Prentice Hall	España, 2003
Pressman, R.	Ingeniería del Software: Un Enfoque Práctico, 5ta. ed.	Mc Graw Hill	España, 2007
Sommerville, I.	Ingeniería del Software, 7ma. Ed.	Prentice Hall	España, 2008
Wood, D.	Diseño de Interfaces. Introducción a la Comunicación Visual en el diseño de Interfaces de Usuario.	Parramon	España, 2015
Reyes, D.	Apuntes de Cátedra.	Del autor	Argentina, 2024

## 4. Estrategias de Enseñanza

- Trabajos grupales e individuales.
- Elaboración de informes escritos.
- Exposiciones grupales e individuales
- Aprendizaje cooperativo.
- Aula invertida.

## 5. Evaluación

### 5.1. Criterios de Evaluación durante el Cursado

- Participación comprometida en los trabajos propuestos, ya sean individuales o grupales.
- Coherencia y adecuación en los trabajos presentados.
- Capacidad de análisis crítico y reflexivo de las lecturas articuladas con los marcos conceptuales.
- Compromiso y responsabilidad en las tareas a desarrollar
- Presentación de trabajos en tiempo y forma
- Manejo adecuado de la bibliografía
- Coherencia, claridad conceptual, integración y pertinencia tanto en oratoria como en el discurso escrito
- Presentación de los trabajos en tiempo y en forma.
- Dominio, comprensión y transposición de los contenidos del programa.
- Capacidad de resolución de problemáticas presentadas.
- Capacidad de organizar, coordinar y elaborar la tarea encomendada
- Aplicación de conocimientos y dominio de técnicas, mediante la respuesta a preguntas y la resolución de problema.
- Participación activa y pertinente en el desarrollo de las clases.

### 5.2. Instrumentos de Evaluación

Para valorar los aprendizajes de los alumnos y sistematizar dichas valoraciones, se utilizarán los siguientes instrumentos:

- Evaluaciones escritas
- Evaluaciones orales
- Trabajos prácticos
- Coloquios
- Informes

### 5.3. Criterios de Acreditación

#### 5.3.1. Para Regularizar la unidad curricular

Para regularizar la asignatura, el alumno deberá:

- Tener el 80% de asistencia a las clases o de participación en el aula virtual.
- Presentar a término y aprobar el 90% de los trabajos prácticos o actividades propuesto.
- Presentar y aprobar un portfolio de evidencias de las experiencias del alumno.
- Aprobar los exámenes parciales propuestos o sus respectivos exámenes recuperatorios. La nota mínima obtenida no deber ser menor de 5 (cinco), en una escala de 1 (uno) a diez (10).
- Obtener una calificación final mínima de 5 (cinco) puntos, en una escala de 1(uno) a 10(diez), al terminar el cursado de la asignatura.

Las condiciones de regularización descriptas serán expuestas a los alumnos durante la presentación de la Unidad Curricular y del plan de trabajo de la misma.

#### 5.3.2. Para Promocionar la Unidad Curricular

Según lo establece la normativa vigente (RAM), la Unidad Curricular NO ES promocionable.

### 5.4. Para Finalizar la Unidad Curricular como alumno Regular

En base a la normativa vigente (RAM), para rendir esta Unidad Curricular en calidad de alumno regular el alumno deberá rendir un primer examen final escrito sobre aspectos teóricos de la misma. Este examen versará sobre las temáticas comprendidas en el programa de la asignatura, desde la perspectiva de la bibliografía obligatoria. Los criterios de evaluación se basan en la acreditación de la lectura de la bibliografía, la argumentación sólida que utilice la terminología y los conceptos del campo de estudio y la apropiación crítica de los mismos.

### 5.5. Para finalizar la Unidad Curricular como alumno Libre

De acuerdo con la normativa actual (RAM), las condiciones establecidas para rendir esta Unidad Curricular en calidad de alumno libre comprenden:

- Presentación y aprobación de la totalidad de los trabajos prácticos o de un trabajo práctico integrador equivalente, en ambos casos, propuestos por la Cátedra. Para ello, el alumno deberá comunicarse con la Cátedra con treinta (30) días de anticipación a la fecha del examen final, a fin de obtener el conjunto de trabajos prácticos que deberá resolver. Los mismos deberán ser entregados al docente responsable siete (7) días antes de la fecha de examen final.
- Superada la instancia anterior, el alumno estará en condiciones de rendir un primer examen final escrito sobre aspectos teóricos (25%) y prácticos (75%) de la asignatura.
- En caso de aprobar el examen escrito, el alumno estará en condiciones de rendir un segundo examen final de características similares al que rinden los alumnos regulares. Este examen será tomado en forma oral o escrita y versará sobre las temáticas comprendidas en el programa de la asignatura, desde la perspectiva de la bibliografía obligatoria. Los criterios de evaluación se basan en la acreditación de la lectura de la bibliografía, la argumentación sólida que utilice la terminología y los conceptos del campo de estudio y la apropiación crítica de los mismos.

## 6. Materiales para el aprendizaje

- Gabinete de Informática, con la suficiente cantidad de computadoras con el siguiente software debidamente instalado: Windows 10+ y una herramienta CASE para modelado del Diseño (ArgoUML, StartUML o similar).
- Apuntes de Cátedra.
- Guía de trabajos prácticos
- Pizarra y fibrones.
- Notebook.
- Cañón proyector.

## 7. Cronograma tentativo de los Contenidos Programados

El siguiente cronograma representa el avance semanal de los contenidos desarrollados, considerando el calendario escolar en curso, incluyendo los días feriados y/o no laborables.

Semana	Fecha	Tema
1	11/03/2024 - 17/03/2024	El Diseño en el proceso de desarrollo de SW. ¿Qué es diseñar? Definición de Diseño. El Proceso de Diseño. Diseño de la Solución. Evolución del Proceso de Diseño. Criterios para un buen diseño.
2	18/03/2024 - 24/03/2024	Principios de Diseño. Importancia del diseño de SW. Conceptos de Diseño. Conceptos de DOO.
3	01/04/2024 - 07/04/2024	Clases de diseño. Clases de diseño bien formadas. Objetivos de la fase de diseño. El modelo de diseño. Dimensiones del modelo de diseño.
4	08/04/2024 - 14/04/2024	Elementos del modelo de diseño. Punto de vista de gestión. Punto de vista técnico. Elementos del diseño del despliegue. Estrategias para el diseño modular.
5	15/04/2024 - 21/04/2024	Documentación del diseño del SW. Notaciones para el diseño de SW. Especificación de Diseño.
6	22/04/2024 - 28/04/2024	Fundamentos de la Arquitectura de Software. Qué es la arquitectura de software. Importancia de la arquitectura de software. Elementos del Diseño Arquitectónico. Diferencias entre Diseño y Arquitectura.
7	29/04/2024 - 05/05/2024	Vistas Arquitectónicas. Qué son las vistas arquitectónicas. Cómo documentar las arquitecturas de un sistema. Estilos y Patrones Arquitectónicos. Clasificación de estilos de arquitectónicos.
8	06/05/2024 - 12/05/2024	Arquitectura en Metodologías Ágiles. Principios de arquitectura ágil. Arquitectura evolutiva y emergente. Integración continua y entrega continua.
9	13/05/2024 - 19/05/2024	Diseño en el nivel de componentes. ¿Qué es un componente? Distintas visiones del diseño en el nivel de componentes. Diseño de componentes basados en clases.
10	20/05/2024 - 26/05/2024	Principios básicos del diseño en el nivel de componentes. Lineamientos de diseño en el nivel de componentes. Cohesión. Acoplamiento. Realización del diseño en el nivel de componentes.
11	27/05/2024 - 02/06/2024	Diseño de la interfaz de usuario. Las reglas doradas para el diseño de interfaces.



Semana	Fecha	Tema
12	03/06/2024 - 09/06/2024	Análisis y Diseño de la Interfaz de usuario. Etapas del diseño de la Interfaz. Patrones de diseño de la interfaz de usuario. Algunas consideraciones para el diseño de interfaz del usuario.
13	10/06/2024 - 16/06/2024	<b>Examen parcial</b>
14	17/06/2024 - 23/06/2024	<b>Recuperatorio del examen parcial</b>

Firma del docente