



UNIVERSIDAD NACIONAL DE INGENIERÍA

**Facultad de Ingeniería Económica, Estadística y CC. SS.
Escuela Profesional de Ingeniería Estadística**

SÍLABO

CURSO: FCC61 DISEÑO Y ANALISIS DE SISTEMAS

I. INFORMACIÓN GENERAL

CÓDIGO	: FCC61
NOMBRE	: DISEÑO Y ANALISIS DE SISTEMAS
CICLO	: VI
CRÉDITOS	: 4
HORAS POR SEMANA	: 5 (2 Teoría – 3 Práctica)
CONDICION	: Obligatorio
DEPARTAMENTO ACADEMICO	: Estadística
PROFESOR	: Dr. Eric Gustavo Coronel Castillo
E-MAIL	: gcoronel@uni.edu.pe

II. SUMILLA DEL CURSO

El curso de Análisis y Diseño de Sistemas proporciona las bases metodológicas para el desarrollo de sistemas de información utilizando el Rational Unified Process (RUP). Se enfoca en el análisis de requerimientos, modelado de procesos y diseño de arquitectura de sistemas con UML (Unified Modeling Language). Se aplican técnicas para el modelado de casos de uso, clases, secuencias y componentes, asegurando una transición efectiva a la implementación.

En la segunda mitad del curso, se desarrolla la implementación del sistema utilizando Java y SQL Server, abordando la conexión con bases de datos, desarrollo de interfaces y optimización de consultas. Se enfatiza la construcción de sistemas eficientes para el análisis y gestión de datos, integrando modelos estadísticos en soluciones de software.

Asimismo, el curso exige la documentación de los análisis y desarrollos conforme a la normativa APA (última edición), asegurando la correcta estructuración de informes, referencias y citaciones en el ámbito académico y profesional.

El curso combina sesiones teóricas con laboratorios prácticos, donde los estudiantes aplican los conceptos a través de un proyecto integrador, fortaleciendo sus habilidades en el diseño, desarrollo y prueba de sistemas de información.

III. COMPETENCIAS DE LA ASIGNATURA

Al finalizar la asignatura, el estudiante:

- Aplica RUP como proceso estructurado para el análisis, diseño e implementación de sistemas de información.
- Aplica técnicas como entrevistas, historias de usuario y casos de uso para analizar correctamente los requerimientos de negocio en el diseño de sistemas de información.
- Estructura y documenta requerimientos de un sistema utilizando artefactos de RUP, como el Modelo de Casos de Uso y la Especificación de Requerimientos.
- Utiliza UML para modelar la estructura, comportamiento e interacción de un sistema mediante diagramas de casos de uso, clases, secuencia y componentes.
- Desarrolla aplicaciones en lenguaje Java integradas con bases de datos SQL Server, aplicando principios de persistencia y optimización de consultas.

IV. UNIDADES DE APRENDIZAJE

Unidad I: Análisis y diseño de sistemas de información con RUP y UML

Esta unidad abarca las etapas iniciales del desarrollo de sistemas de información, con énfasis en el análisis de requerimientos y el diseño de la arquitectura del sistema utilizando el proceso unificado RUP (Rational Unified Process) y el lenguaje de modelado UML. A lo largo de las semanas 1 a 8, el estudiante explora los conceptos fundamentales de los sistemas de información y su ciclo de vida, comparando metodologías estructuradas frente a enfoques ágiles (por ejemplo, RUP vs. Scrum) para contextualizar el marco de trabajo. Se profundiza en la elicitación y documentación de requerimientos, aplicando técnicas como entrevistas, historias de usuario y casos de uso, y formalizando los resultados con artefactos estándar de RUP (modelo de casos de uso, especificación de requerimientos, etc.). La unidad cubre el modelado detallado del sistema mediante UML en sus distintos diagramas: casos de uso (de negocio y del sistema), actividades, clases, secuencia, componentes y despliegue. Asimismo, se incluye el modelado de datos conceptual a través de diagramas entidad-relación y la incorporación de principios básicos de arquitectura de software y patrones de diseño. Se enfatiza la revisión y refinamiento del diseño antes de la construcción, asegurando una transición fluida hacia la implementación.

SEMANA	UNIDAD	TEMAS PRINCIPALES
1	Introducción a los Sistemas de Información y el Ciclo de Vida con RUP	Conceptos básicos de sistemas de información. Clasificación de los sistemas de información. Ciclo de vida del desarrollo de sistemas con enfoque en RUP.
2	Metodología RUP y UML en el Desarrollo de Sistemas	Introducción a RUP. Características y fases. Comparación con metodologías ágiles (Scrum). Introducción a UML.

3	Análisis de Requerimientos y Modelado UML con RUP	Fundamentos del análisis de requerimientos. Técnicas de elicitación. Documentación de requerimientos con plantillas RUP (Business Case, SRS). Diagramas de Casos de Uso de Negocio. Diagramas de Actividad de Negocio.
4	Diseño de Sistemas - Parte I: Modelado de la Aplicación	Diagramas de Casos de Uso del Sistema. Modelado de procesos de negocio. Modelado de datos con Diagramas Entidad-Relación.
5	Diseño de Sistemas - Parte II: Arquitectura de Software	Diagramas de Secuencia. Diagramas de Clases. Mapeo del Modelo de Clases a Entidades en Bases de Datos.
6	Diseño de Sistemas - Parte III: Patrones de Diseño y Diagramas UML	Introducción a Patrones de Diseño en software. Diagramas de Componentes. Diagramas de Despliegue.
7	Revisión y Ajustes del Diseño	Evaluación de modelos UML diseñados. Refinamiento de la estructura del sistema previo a la implementación.
8	EXAMEN PARCIAL	Evaluación de los temas desarrollados hasta el momento.

Unidad II: Implementación de sistemas de información con Java y SQL Server

La segunda unidad comprende la fase de construcción e implementación del sistema de información diseñado en la unidad previa, empleando tecnologías Java para el desarrollo de la aplicación y SQL Server para la gestión de datos. Abarcando las semanas 9 a 16, se aborda la arquitectura multicapa del software, estableciendo la interacción entre la capa de lógica de negocio en Java y la base de datos relacional en SQL Server (incluyendo la conexión mediante JDBC). El estudiante implementa la persistencia de datos, realizando operaciones CRUD y consultas SQL optimizadas desde Java, a la vez que desarrolla la lógica de negocio del sistema siguiendo buenas prácticas de programación orientada a objetos (organización en controladores, servicios, etc.). Paralelamente, se diseña e implementa la capa de presentación (interfaz gráfica de usuario), integrándola con la lógica de negocio para asegurar la coherencia entre todas las capas de la aplicación. A lo largo de esta unidad se enfatiza la realización de pruebas rigurosas: se llevan a cabo pruebas unitarias y funcionales tanto de la capa lógica como de la interfaz, con el fin de depurar errores y garantizar el correcto funcionamiento del sistema. Asimismo, la unidad incorpora el uso de herramientas de generación de reportes (como JasperReports y JFreeChart) para crear reportes dinámicos y visualizaciones gráficas basadas en los datos del sistema. Finalmente, se integran estos reportes en la aplicación Java y se realizan ajustes finales y optimizaciones, consolidando todos los componentes desarrollados.

SEMANA	UNIDAD	TEMAS PRINCIPALES
9	Introducción al Desarrollo de Software con Java y SQL Server	Arquitectura en capas. Introducción al desarrollo con Java y SQL Server. Conexión de Java con SQL Server mediante JDBC.

10	Persistencia de Datos en Java con SQL Server	Persistencia de datos en Java con consultas SQL directas. Optimización y manipulación de datos desde Java.
11	Desarrollo de la Capa Lógica del Sistema y Pruebas	Implementación de la lógica de negocio en Java. Creación de controladores y servicios. Pruebas unitarias de la capa lógica.
12	Desarrollo de la Capa Lógica del Sistema y Pruebas (continuación)	Finalización de la implementación de la lógica de negocio. Pruebas funcionales y depuración de errores.
13	Desarrollo de la Capa de Presentación (View Layer) y Pruebas	Diseño de interfaces gráficas. Integración con la capa de negocio.
14	Generación de Reportes Estadísticos con JasperReports y JFreeChart	Introducción a JasperReports y JFreeChart. Generación de reportes dinámicos basados en SQL Server.
15	Integración de Reportes en la Aplicación y Ajustes Finales	Integración de reportes en la aplicación Java. Ajustes finales y optimización de reportes estadísticos.
16	EXAMEN FINAL	Evaluación integral del curso.

V. LABORATORIOS Y EXPERIENCIAS PRÁCTICAS

Se ha programado realizar 2 laboratorios que son desarrollados en forma grupal, los cuales son complementados con el informe técnico elaborado por los estudiantes, según la guía de laboratorios, se utilizan casos que tengan relación con el mundo real:

- **Laboratorio 1**

Implementación de procesos aplicando la arquitectura en capas y el enfoque de servicios.

- **Laboratorio 2**

Implementación de reportes estadísticos utilizando JasperReport y gráficos utilizando JFreeChart.

VI. METODOLOGÍA

El curso se desarrolla en sesiones de teoría, practica y laboratorio de cómputo. En las sesiones de teoría, el docente desarrolla los conceptos teóricos, presenta ejemplo y resuelve problemas. En las sesiones prácticas, se implementan casos concretos utilizando lenguaje Java y base de datos SQL Server. En las sesiones de laboratorio, el estudiante debe resolver el caso propuesto para cada laboratorio.

Durante el curso los estudiantes en grupo de 3 participantes presentan un proyecto para desarrollar una solución de un caso específico, el cual se inicia en la 2da. semana de clase. El proyecto será evaluado en dos entregas correspondientes a las PC3 y PC4. Cada entrega está conformada por un informe en formato APA, una presentación en PowerPoint, la solución desarrollada y una exposición

según las especificaciones que se darán en clase. Los 2 mejores trabajos serán presentados en la feria de proyectos.

VII. FÓRMULA DE EVALUACIÓN

El Sistema de evaluación que se adoptara es el sistema “I”, el cual propone evaluar los exámenes y prácticas del siguiente modo:

- | | |
|-------------------------|-----|
| ▪ Prácticas calificadas | 50% |
| ▪ Examen Parcial | 25% |
| ▪ Examen Final | 25% |

Es decir, mediante la siguiente fórmula se calcula el Promedio Final:

$$PF = (2*PP + EP + EF) / 4$$

Donde:

PF: Promedio Final

PP: Promedio de prácticas

EP: Examen Parcial

EF: Examen Final

VIII. RECURSOS VIRTUALES

1. PLATAFORMA VIRTUAL

Se utilizará la plataforma virtual de la UNI para dar soporte al curso, la aplicación de esta plataforma servirá para:

- Canal de comunicación con los estudiantes.
- Acceder al material de cursos debidamente organizado, este material puede ser: archivos PDF, enlaces a videos, artículos, etc.
- Plataforma de evaluación preliminar.
- Realizar debate mediante foros.

Para acceder a la plataforma el estudiante debe utilizar el siguiente enlace:

<https://virtual.fieecs.uni.edu.pe/login/index.php>

2. REPOSITORIO DE CODIGO FUENTE

También se utilizará un repositorio de código fuente, en este repositorio se compartirá el código de todos los ejemplos desarrollados en clase, para acceder a este repositorio el estudiante debe hacer uso de la siguiente URL:

<https://github.com/gcoronelc/UNI-EPIES-DAS-2025-2>

IX. BIBLIOGRAFÍA

- Kendall, K y Kendall, J. (2011). Análisis y Diseño de Sistemas (8a. ed.). México: Prentice Hall.

- Pender TA. (2002). UML WEEKEND CRASH COURSE. 1st ed. Indianapolis, Indiana: Wiley Publishing, Inc.
- Dennis A, Wixom BH, Tegarden DP. (2015). SYSTEMS ANALYSIS AND DESIGN WITH UML. 5th ed. Hoboken, NJ: John Wiley & Sons.
- Arlow J, Neustadt I. (2005). UML 2 AND THE UNIFIED PROCESS: PRACTICAL OBJECT-ORIENTED ANALYSIS AND DESIGN. 2nd ed. Upper Saddle River, NJ: Addison-Wesley.
- Dennis A, Wixom BH, Roth RM. (2012). SYSTEMS ANALYSIS AND DESIGN. 6th ed. Hoboken, NJ: John Wiley & Sons.
- Bruegge B, Dutoit AH. (2010). OBJECT-ORIENTED SOFTWARE ENGINEERING: USING UML, PATTERNS, AND JAVA. 3rd ed. Boston: Prentice Hall.
- Gomaa H. (2011). SOFTWARE MODELING AND DESIGN: UML, USE CASES, PATTERNS, AND SOFTWARE ARCHITECTURES. Cambridge; New York: Cambridge University Press.

Enlaces virtuales:

- <https://www.youtube.com/DesarrollaSoftware>
- <https://gcoronelc.blogspot.com/>