



Facultad de Ingeniería
Departamento Académico de Ingeniería
Escuela Profesional de Ingeniería Informática

SÍLABO 2023-II

I. DATOS ADMINISTRATIVOS

1. Asignatura	:	Análisis de Sistemas y Diseño de Software
2. Código	:	IF0604
3. Naturaleza	:	Teórica-Práctica
4. Condición	:	Obligatorio
5. Requisitos	:	IF0504 Ingeniería de Requerimientos
6. Nro. Créditos	:	4.5
7. Nro. de horas	:	7 (Teoría=3 Taller=3)
8. Semestre Académico	:	2023-I
9. Docente	:	Mg. Ing. Edgard De Olazábal León
10. Correo Institucional	:	edgar.deolazabal@urp.edu.pe

II. SUMILLA

El curso de Análisis de Sistemas y de Diseño de Software corresponde al sexto semestre del Plan de Estudios de la Escuela Académico Profesional de Ingeniería Informática. Es de naturaleza teórico-práctico. Busca que el alumno consiga desarrollar capacidades en las técnicas modernas de Análisis de Sistemas y de Diseño de Software que le permitan construir un producto de software satisfactorio y eficiente. Esta asignatura propicia el trabajo grupal y de investigación mediante la elaboración de un proyecto que debe ser sustentado por los alumnos a lo largo de todo el ciclo académico. Tiene como propósito desarrollar en el estudiante las capacidades necesarias para realizar y especificar el Diseño del Software, que luego permita su implementación.

III. COMPETENCIAS GENÉRICAS A LAS QUE CONTRIBUYE LA ASIGNATURA

- Pensamiento crítico y creativo
- Resolución de problemas
- Investigación científica y tecnológica

IV. COMPETENCIAS ESPECÍFICAS A LAS QUE CONTRIBUYE LA ASIGNATURA

- Diseño en ingeniería
- Soluciona problemas de Ingeniería.
- Gestión de Proyectos
- Trabajo en equipo
- Comunicación

V. DESARROLLA EL COMPONENTE DE: INVESTIGACIÓN () RESPONSABILIDAD SOCIAL ()

VI. LOGRO DE LA ASIGNATURA

Al finalizar la asignatura, el estudiante desarrolla su capacidad para solucionar problemas de ingeniería, diseñando soluciones informáticas, construyendo la parte inicial de un proyecto de desarrollo de un producto software, trabajando en equipo y utilizando el proceso unificado de desarrollo y métodos y ágiles para su desarrollo.

VII. PROGRAMACIÓN DE CONTENIDOS

UNIDAD I: REVISIÓN DE REQUERIMIENTOS	
LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante sustenta el planteamiento inicial de la solución de un problema de gestión de información, identificando los requerimientos funcionales y no funcionales y, desarrollando el modelo de casos de uso del sistema a trabajar, priorizando y clasificando el desarrollo de estos.	
Semana	Contenido



Universidad Ricardo Palma
Rectorado
Oficina de Desarrollo Académico, Calidad y Acreditación

1	Revisión de la disciplina de Requerimientos en el ciclo de desarrollo iterativo del producto software. Pirámide de Requerimientos, Requerimientos Funcionales y No funcionales, atributos de los requerimientos. Aplicaciones.
2	Visión del Sistema. Modelo de Casos de uso del sistema. Identificación de actores y casos de uso. Especificación a alto nivel de los casos de uso. Priorización de Casos de uso. Aplicaciones.
3	Estructurar Casos de Uso del sistema. Generalización de actores. Relaciones Include Extend. Organización de paquetes. Diagramas de Casos de uso por Paquete. Matriz de trazabilidad RF versus CUS. Aplicaciones.
4	Especificación detallada de los casos de uso del sistema. Métodos ágiles, Historia de Usuarios, Épicas, Product Backlog y Sprints. Evaluación del Logro

UNIDAD II: GENERACIÓN DEL MODELO DE ANÁLISIS

LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante elabora el modelo de análisis, desarrollando los diagramas de interacción, así como el modelo conceptual de entidades del sistema. Identifica clases reactivas y elabora sus diagramas de estado. Elabora el modelo de datos partiendo de su modelo conceptual.

Semana	Contenido
5	Modelo de Análisis, Realización de los Casos de Uso. Identificar las Clases de análisis (Estereotipos: interfaz, control, entidad). Desarrollo de Diagramas de Interacción: Secuencia y Colaboración. Desarrollo de Prototipos de Interfaz de Usuario Aplicaciones.
6	El Modelo Conceptual de Entidades del Sistema, Atributos y Relaciones: Tipos de asociaciones, multiplicidad. Aplicaciones
7	Diagramas de Transición de Estados. Máquinas de estado. Aplicaciones.
8	Transformación del Modelo Conceptual al Modelo de Datos. Evaluación del Logro

UNIDAD III: GENERACIÓN DEL MODELO DE DISEÑO

LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante elabora el modelo de diseño, desarrollando los diagramas de interacción en el diseño, así como incorporando los patrones de diseño en su solución propuesta a la problemática grupal planteada.

Semana	Contenido
9	Revisión de los flujos del Diseño. Diseño de una clase, diseño de un Subsistema, diseño de un Caso de Uso. Aplicaciones.
10	Diseño de la Arquitectura. Arquitectura de capas. Desarrollo de Diagramas de Interacción (Colaboración, Secuencia) con clases de Diseño. Identificación de los métodos de las clases de diseño. Aplicaciones.
11	Patrones de diseño. Patrones Gang of Four. Patrones creacionales, estructurales y de comportamiento.
12	Aplicación de patrones de diseño en casos de uso del ciclo cero Evaluación del Logro

UNIDAD IV: MODELO DE IMPLEMENTACIÓN Y MODELO DE DESPLIEGUE

LOGRO DE APRENDIZAJE: Al finalizar la asignatura, el estudiante incorporará a su modelo del sistema los subsistemas y paquetes de componentes y distribución predefinidos, mostrando orden y rigurosidad en su procedimiento. presentación en formato digital.

Semana	Contenido
13	Representación de los componentes: Diagrama de Componentes e implementación. Aplicaciones.
14	Representación del ambiente físico: Diagrama de despliegue. Aplicaciones.
15	Desarrollo e incorporación de componentes, así como despliegue de casos de uso ciclo cero. Aplicaciones.
16	Evaluación del Logro
17	EVALUACIÓN SUSTITUTORIA CON PRODUCTO FINAL: RÚBRICA



VIII. ESTRATEGIAS DIDÁCTICAS

- Aula invertida
- Aprendizaje colaborativo
- Aprendizaje basado en proyectos
- Aprendizaje basado en problemas
- Disertación

IX. MOMENTOS DE LA SESIÓN DE APRENDIZAJE VIRTUAL

La modalidad no presencial desarrollará actividades sincrónicas (que los estudiantes realizarán al mismo tiempo con el docente) y asincrónicas (que los estudiantes realizarán independientemente fortaleciendo su aprendizaje autónomo. La metodología del aula invertida organizará las actividades de la siguiente manera:

Antes de la sesión

Exploración: preguntas de reflexión vinculada con el contexto, otros.

Problematización: conflicto cognitivo de la unidad, otros.

Durante la sesión

Motivación: bienvenida y presentación del curso, otros.

Presentación: PPT en forma colaborativa, otros.

Práctica: resolución individual de un problema, resolución colectiva de un problema, otros.

Después de la sesión

Evaluación de la unidad: presentación del producto.

Extensión / Transferencia: presentación en digital de la resolución individual de un problema.

IX. EVALUACIÓN

La modalidad no presencial se evaluará a través de productos que el estudiante presentará al final de cada unidad. Los productos son las evidencias del logro de los aprendizajes y serán evaluados a través de rúbricas cuyo objetivo es calificar el desempeño de los estudiantes de manera objetiva y precisa.

Retroalimentación. En esta modalidad no presencial, la retroalimentación se convierte en aspecto primordial para el logro de aprendizaje. El docente devolverá los productos de la unidad revisados y realizará la retroalimentación respectiva.

El promedio final del curso será calculado como un promedio ponderado según muestra el siguiente cuadro:

Concepto		Detalles	UNIDAD	Observación
Teoría	PRT1	Práctica Teórica 1	I	Se elimina la más baja
	PRT2	Práctica Teórica 2	II	
	PRT3	Práctica Teórica 3	III	
	PRT4	Práctica Teórica 4	IV	
Laboratorio	TLR1	Trabajo del Curso		No se elimina

FORMULA PARA CALCULAR NOTA FINAL DEL CURSO

$$(PRT1+PRT2+PRT3+PRT4+TLR1)/4$$

X. RECURSOS

- Equipos: computadora, laptop, Tablet, celular, con conexión a Internet.
- Software: Rational Rose, Rational Software Architect, StarUml
- Materiales: apuntes de clase del Docente, separatas de problemas, lecturas, videos.
- Plataformas: MLS Moodle, Collaborate, Kahoot.



XI. REFERENCIAS

Bibliografía Básica

Booch G., Jacobson I. y Rumbaugh J. (2000). *El proceso unificado de desarrollo de software*. Editorial Addison Wesley.

Booch G., Jacobson I. y Rumbaugh J. (2004). *The Unified Modeling Language - User Guide*. Addison-Wesley Educational Publishers Inc; 2da edición.

Rumbaugh J. (2004). *Object-Oriented Modeling and Design with UML*. Prentice Hall, Inc. 2da edición.

Larman C. (2003). *Aplicando Uml y Patrones: una introducción al análisis y diseño orientado a objetos y al proceso unificado*. Editorial: Prentice Hall

Bibliografía complementaria

Gamma E.; Helm R.; Jhonson R. y Vissides [J.](#) (2003) *Patrones de Diseño*. Editorial Addison Wesley.

.

Larman C. (2004). *Agile and iterative development: a managers guide*