



PROGRAMA DEL CURSO
IF6100 - Análisis y Diseño de Sistemas
II CICLO 2021

1 Datos Generales

- Sigla: IF6100
- Nombre: Análisis y Diseño de Sistemas
- Tipo de curso: Teórico-Práctico
- Créditos: 4
- Horas lectivas: 6
- Requisitos: IF-5100 Administración de bases de datos
- Correquisitos: Ninguno
- Ubicación en el plan de estudio: VI Ciclo
- Suficiencia: No
- Modalidad: Virtual

2 Descripción

En este curso el estudiantado asimilará los conceptos, herramientas, técnicas y metodologías necesarias para el análisis y diseño de sistemas de información, con el fin de que pueda determinar los requerimientos de un sistema, definir las estrategias de su desarrollo y administración del proceso de desarrollo.

3 Objetivo General

Al finalizar el curso el estudiantado deberá dominar los conceptos, herramientas, técnicas y metodologías necesarias para el análisis y diseño de sistemas de información, así como ser capaz de desarrollar un sistema de información.



4 Objetivos Específicos

Al finalizar el curso la persona estudiante estará en capacidad de:

- Conocer los roles y responsabilidades que asume una persona informática en el área de análisis y diseño de sistemas de información.
- Comprender la teoría de sistemas y sus componentes así como la teoría de comunicar su análisis mediante diferentes herramientas de diagramación.
- Ejecutar procesos de elicitación y especificación de requerimientos de software mediante técnicas actualizadas.
- Modelar las distintas vistas arquitectónicas que constituyen la solución del sistema de información que se desea construir.
- Implementar el marco de trabajo Proceso Unificado en proyectos de software.
- Analizar requerimientos, utilizar metodologías para extraer la información requerida de los usuarios y realizar un diseño adecuado de los sistemas por crear.
- Conceptualizar la metodología de análisis y diseño orientada a objetos.
- Formular modelos de diseño basados en el lenguaje de modelado unificado (UML).
- Implementar sistemas en función del diseño determinado.
- Implementar un plan de pruebas de aceptación.
- Emplear eficazmente herramientas de software modernas para apoyar el análisis y diseño de sistemas de información, así como para ejecutar la administración eficiente del proceso.
- Adoptar una actitud crítica ante la diversidad de temáticas asociadas con el análisis y diseño.

5 Contenidos

5.1. Introducción al Análisis y Diseño de Sistemas

5.1.1 La necesidad del análisis y diseño de sistemas.



5.1.2 Ciclo de vida del desarrollo de sistemas.

5.1.3 Modelos de desarrollo del software: cascada, espiral, iterativo-incremental, evolutivo, basado en componentes, RAD

5.2. Métodos de análisis de sistemas

5.2.1 Desarrollo de la visión de un sistema.

5.2.2 Análisis de sistemas.

5.2.3 Fases del análisis de sistemas.

5.2.4 Factibilidad del sistema.

5.2.5 Modelado de negocio.

5.3. Ingeniería de requerimientos

5.3.1 Qué es la ingeniería de requerimientos.

5.3.2 Clasificación de requerimientos.

5.3.3 Características de un buen requerimiento.

5.3.4 Guías de apoyo: Entrevistas, Cuestionarios, Sesiones de trabajo, Lluvia y reducción de ideas, Storyboarding, Representación de roles, Diagramas de Espina de Pescado, Diagramas de Pareto, Juego de roles, Diagramas de afinidad.

5.3.5 Artefactos de la ingeniería de requerimientos: plan de administración de requerimientos, peticiones de los afectados, visión, especificación de casos de uso y requerimientos suplementarios.

5.3.6 Los prototipos como medio de validación de requerimientos.

5.3.7 Modelado de Casos de Uso.

5.3.8 Diagrama de Casos de Uso.

5.3.9 Especificación de Casos de Uso.

5.3.10 Especificaciones suplementarias (requerimientos no funcionales)

5.3.11 Reglas de negocio

5.3.12 Validación de requerimientos.

5.3.13 Atributos de los requerimientos.

5.3.14 Trazabilidad.



5.4. Lenguaje de modelado unificado

5.4.1 El análisis y diseño orientado a objetos y el UML.

5.4.2 Introducción a UML.

5.4.3 Diagramas UML.

- i. Diagrama de paquetes.
- ii. Diagramas de clase: notación de clase, asociación, roles, multiplicidad, herencia, interfaces, agregación simple y compuesta.
- iii. Diagramas de interacción: secuencia y de comunicación.
- iv. Diagramas de despliegue.
- v. Diagramas de componentes.
- vi. Diagramas de actividad y de máquinas de estado.

5.5. Diseño de Sistemas

5.5.1 Fundamentos y propósitos de la disciplina.

5.5.2 Análisis y diseño del sistema desde los casos de uso: realización de los casos de uso

5.5.3 Arquitectura del software.

5.5.4 Diseño de la base de datos.

5.5.5 Artefactos de la disciplina: Modelo de Análisis, Documento de Arquitectura de Software, Modelo de Diseño de la Base de Datos, Modelo de Diseño, Modelo de Diseño de Interfaz Gráfica.

5.5.6 Subsistemas e interfaces.

5.5.7 Diseño de Componentes.

5.5.8 Diagramación del diseño.

5.5.9 Diagramas de flujos de datos.

5.5.10 Diagramas de interacción y clase.

5.5.11 Clases de diseño: límite, control y entidad.

5.5.12 Patrones del diseño.

5.6. Construcción

5.6.1 Fundamentos de construcción.



- 5.6.2 Prototipos en la etapa de diseño.
- 5.6.3 Transición del diseño a la implementación.
- 5.6.4 Versiones funcionales del sistema.
- 5.6.5 Integración de subsistemas y del sistema.

5.7. Pruebas

- 5.7.1 Propósitos de la disciplina.
- 5.7.2 Pruebas de caja negra y caja blanca.
- 5.7.3 Tipos de pruebas: pruebas de unidad, de integración, alfa, beta, de casos de uso, de componentes, de instalación, de aceptación, de seguridad entre otros.
- 5.7.4 Creación de las pruebas desde los casos de uso y los requerimientos suplementarios.
- 5.7.5 Estrategias de pruebas: desarrollo, fase de pruebas, liberación.
- 5.7.6 Automatización de las pruebas.
- 5.7.7 Artefactos: Plan de Calidad del Software, Plan de Pruebas, Elementos de Prueba, Resultados de Prueba.

5.8. Metodologías básicas del desarrollo de sistemas

- 5.8.1 Proceso unificado (RUP)
- 5.8.2 Orientado a objetos
- 5.8.3 Fundamentos de metodologías ágiles

6 Metodología

- 6.1. El curso presenta un eje de desarrollo teórico-práctico.
- 6.2. La persona docente desarrolla las clases soportada en diapositivas, modelos, plantillas de artefactos y código fuente.
- 6.3. Los materiales didácticos y las tareas del curso se gestionarán a través de Mediación Virtual.
- 6.4. El estudiantado desarrolla las siguientes actividades de aprendizaje:



- 6.4.1. Realizan lecturas sobre temáticas que se vayan a abordar en el curso.
 - 6.4.2. Realizan investigaciones aplicadas sobre temas relacionados con el análisis y diseño de sistemas.
 - 6.4.3. Desarrollan un proyecto de software en el cual efectuarán el análisis, diseño e implementación. El mismo lo gestionarán mediante el marco de trabajo denominado Proceso Unificado. Los requisitos técnicos, políticas de evaluación y fechas de los puntos de control del proyecto estarán consignadas en un documento de proyecto que se entregará de forma oportuna. El proyecto tendrá puntos de control, en los cuales deberán entregarse los elementos o artefactos definidos en el enunciado del proyecto. Cada punto de avance deberá ser discutido, con antelación a la entrega, en consultoría con el profesor o profesora.
 - 6.4.4. Realizan casos de estudio y tareas en donde ponen en práctica las temáticas estudiadas en el curso.
- 6.5. Las sesiones de clase se desarrollarán de forma sincrónica y asincrónica según la estrategia didáctica acorde con los contenidos del curso. Con respecto a las actividades asincrónicas, entre ellas laboratorios, casos de estudio y proyectos, el profesorado dará el seguimiento correspondiente vía correo institucional y mediante videollamadas de la plataforma Zoom.

7 Evaluación

DESCRIPCIÓN	PORCENTAJE
Examen parcial I	20%
Examen parcial II	20%
Casos de estudio y tareas	20%
Proyecto práctico	30%
Investigación	10%

7.1 Consideraciones sobre la evaluación

- Según lo establecido en las resoluciones VD-R-8458-2009 y VD-R-9374-2016, se utilizará el entorno virtual de aprendizaje institucional Mediación Virtual (<https://mediacionvirtual.ucr.ac.cr>). El mismo se empleará para la entrega del programa del curso, material, enunciados de evaluaciones, entre otros, por



parte del profesorado. En el caso del estudiantado, para el envío de entregables y/o realización de evaluaciones asociadas al curso.

- La resolución R-2664-2012 define el correo institucional con el dominio ucr.ac.cr como la herramienta oficial para las comunicaciones de toda la comunidad universitaria. Por lo tanto, se utilizará el correo institucional como medio oficial de comunicación entre docentes y estudiantes, por lo cual el estudiantado deberá tenerlo activo y revisarlo continuamente.
- Los criterios de calificación de cada evaluación serán especificados en el enunciado de la misma.
- Toda evaluación será comunicada al estudiantado del curso al menos 5 días hábiles antes de realizarse, de acuerdo con lo especificado en los artículos 15 y 18 del Reglamento de Régimen Académico Estudiantil.
- En caso de ausencia a alguna evaluación, se procederá según lo establecido en el Artículo 24 del Reglamento de Régimen Académico Estudiantil.
- Ante la detección de una posible copia o plagio, total o parcial, en cualquier evaluación, se procederá de acuerdo con lo establecido en el Reglamento de Orden y Disciplina Estudiantil.
- Como parte de las lecturas de apoyo a los temas que se desarrollarán en clase, se utilizará al menos dos lecturas en idioma inglés. El objetivo principal de este aspecto es impulsar la comprensión de lectura. Debido a que hay estudiantes con diferente nivel lingüístico, los reportes y presentaciones para revisar el material leído se deben realizar en idioma español.
- Las fechas del cronograma están sujetas a cambio dependiendo del avance en los contenidos.

8 Docentes del curso

GRUPO	AULA	DOCENTE	HORARIO	CONSULTA
SEDE DEL SUR				
01	Virtual	Lic Nury Leitón Baltodano nury.leitonbaltodano@ucr.ac.cr	K 17 a 20:50 J 08 a 09:50	K 13 a 16:50



GRUPO AULA		DOCENTE	HORARIO	CONSULTA
SEDE DEL ATLÁNTICO, RECINTO DE GUÁPILES				
31	Virtual	Mag. Carlos F. Escalante Solano carlosfelipe.escalante@ucr.ac.cr	M 08 a 09:50 V 08 a 11:50	K 13 a 14:00 M 10 a 12:00
SEDE DEL ATLÁNTICO, RECINTO DE PARAÍSO				
21	Virtual	MSI. Álvaro Mena Monge alvaro.mena@ucr.ac.cr	K 15 a 17:50 J 13 a 15:50	M 8 a 11:00
SEDE DEL ATLÁNTICO, RECINTO DE TURRIALBA				
01	Virtual	Mag. Juan José Quesada Sánchez juanjose.quesada@ucr.ac.cr	K 19 a 20:50 S 08 a 12:50	J 18 a 21:00
SEDE DEL CARIBE, RECINTO DE LIMÓN				
01	Virtual	Licda. Delia Smith Paul delia.smith@ucr.ac.cr	K 09 a 11:50 J 09 a 11:50	K & J 7:30 a 9:00
SEDE DEL CARIBE, AULA DE SIQUIRRES				
01	Virtual	Licda. Delia Smith Paul delia.smith@ucr.ac.cr	L 09 a 11:50 M 09 a 11:50	L & M 7:30 a 9:00
SEDE DE GUANACASTE				
01	Virtual	Lic. Alonso Chavarría Cubero ivan.chavarriacubero@ucr.ac.cr	M 13 a 16:50	M 17 a 18:50
SEDE DE OCCIDENTE, RECINTO DE GRECIA				
01	Virtual	Lic. Gabriel Quirós Ramírez gabriel.quiroramirez@ucr.ac.cr	L 18 a 20:50 M 18 a 20:50	S 8 a 10:50
SEDE DE OCCIDENTE, RECINTO DE SAN RAMÓN				
01	Virtual	Dra. Iyubanit Rodríguez Ramírez iyubanit.rodriguezramirez@ucr.ac.cr	K 13 a 14:50 V 08 a 11:50	K 08 a 10:50
SEDE DEL PACÍFICO				
01	Virtual	MCI Karol Sugeil Sánchez Pérez karol.sanchez@ucr.ac.cr	L 09 a 11:50 J 09 a 11:50	L 08 a 09:00 L 13 a 14:50
02	Virtual	Lic. Melber Dalorso Cruz melber.dalorso@ucr.ac.cr	L 08 a 11:00 J 08 a 11:00	J 13 a 15:50



9 Cronograma

SEM	FECHA	TEMA O ACTIVIDAD
01	16 - 21 AGO	Introducción al Análisis y Diseño de Sistemas
02	23 - 28 AGO	Métodos de análisis de sistemas
03	30 AGO - 04 SET	Ingeniería de requerimientos
04	06 - 11 SET	Ingeniería de requerimientos Lectura: Dathan y Ramnath (2015) <i>Analysing a System</i>
05	13 - 18 SET	Ingeniería de requerimientos
06	20 - 25 SET	Lenguaje de modelado unificado
07	27 SET - 02 OCT	Lenguaje de modelado unificado
08	04 - 09 OCT	Diseño de Sistemas Lectura: Dooley (2017) <i>Object-Oriented Analysis and Design</i>
09	11 - 16 OCT	Examen Parcial I -Diseño de Sistemas
10	18 - 23 OCT	Diseño de Sistemas
11	25 - 30 OCT	Diseño de Sistemas Lectura: Ingeno (2018) <i>The Meaning of Software Architecture</i>
12	01 - 06 NOV	Construcción
13	08 - 13 NOV	Construcción
14	15 - 20 NOV	Pruebas del Software
15	22 - 27 NOV	Pruebas del Software
16	29 NOV - 4 DIC	Métodologías Básicas de Desarrollo de Sistemas Examen Parcial II
17	6 - 11 DIC	Ampliación

Referencias obligatorias

Chemuturi, M. (2012). *Requirements engineering and management for software development projects*. Springer Science & Business Media. Recuperado de



- <https://doi-org.ezproxy.sibdi.ucr.ac.cr/10.1007/978-1-4614-5377-2>.
Dathan, B. and Ramnath, S. (2015). *Object-Oriented Analysis, Design and Implementation: An Integrated Approach*. Springer. Recuperado de <https://doi-org.ezproxy.sibdi.ucr.ac.cr/10.1007/978-3-319-24280-4>.
Dooley, J. F. (2017). *Software Development, Design and Coding: With Patterns, Debugging, Unit Testing, and Refactoring*. Springer. Recuperado de <https://doi-org.ezproxy.sibdi.ucr.ac.cr/10.1007/978-1-4842-3153-1>.
Joshi, B. (2016). *Beginning solid principles and design patterns for ASP.NET developers*. Springer. Recuperado de <https://doi-org.ezproxy.sibdi.ucr.ac.cr/10.1007/978-1-4842-1848-8>.
Rumbaugh, J., Booch, G., and Jacobson, I. (2000). *El lenguaje unificado de modelado: manual de referencia*. Addison Wesley.
Seidl, M., Brandsteidl, M., Huemer, C., and Kappel, G. (2012). *UML@ Classroom*. Springer. Recuperado de <https://doi-org.ezproxy.sibdi.ucr.ac.cr/10.1007/978-3-319-12742-2>.

Referencias secundarias

- Adams, K. M. (2015). *Nonfunctional requirements in systems analysis and design*. Springer. Recuperado de <https://doi-org.ezproxy.sibdi.ucr.ac.cr/10.1007/978-3-319-18344-2>.
Arlow, J. and Neustadt, I. (2005). *UML 2 and the unified process: practical object-oriented analysis and design*. Pearson Education.
Dennis, A., Haley Wixom, B., and Tegarden, D. (2015). *Systems Analysis & Design: An object-oriented approach with UML*. Wiley, 5 edition.
Ingeno, J. (2018). *Software Architect's Handbook: Become a successful software architect by implementing effective architecture concepts*. Packt Publishing Ltd.
Manassis, E. (2004). *Practical software engineering: analysis and design for the .NET platform*. Addison-Wesley.
Shelly, G. B. and Rosenblatt, H. J. (2011). *Systems analysis and design*. Cengage Learning.
Tilley, S. and Rosenblatt, H. (2017). *Systems Analysis and Design*. Cengage Learning, 11 edition.
Tockey, S. (2019). *How to engineer software: a model-based approach*. John Wiley & Sons, Inc.
Unhelkar, B. (2018). *Software Engineering with UML*. CRC Press.
Valacih, J. S. and George, J. F. (2017). *Modern Systems Analysis and Design*. Pearson, 8 edition.



- Walkinshaw, N. (2017). *Software Quality Assurance: Consistency in the Face of Complexity and Change*. Springer. Recuperado de <https://doi-org.ezproxy.sibdi.ucr.ac.cr/10.1007/978-3-319-64822-4>.
- Wieggers, K. (2003). *Software requirements*. Pearson Education.