

Facultad de Ingeniería y Ciencias Agropecuarias
Ingeniería en Sistemas de Computación e Informática
ACI-760 – Desarrollo de Software II
Período 2018-1

A. Identificación

Número de sesiones:	48 horas
Número total de horas de aprendizaje:	48 horas presenciales + 72 horas de trabajo autónomo = 120 horas.
Docente:	Santiago Ramiro Villarreal Narváez
Correo electrónico del docente:	
Director:	Marco Galarza
Campus:	Queri
Pre-requisito:	Análisis de requerimientos (ACI480)
Co-requisito:	N/A
Paralelo:	1

B. Descripción del curso

Se estudiará los conceptos de desarrollo tradicional de software utilizando una metodología orientada a objetos y las fases para desarrollar sistemas de este tipo, partiendo del análisis de requerimientos hasta el diseño con los diagramas de casos, actividad, estado, clases, interacción, componentes y despliegue. Durante el curso se planteará la aplicación de toda la metodología hasta llegar a la fase de diseño de un proyecto de software.

C. Resultados de aprendizaje (RdA) del curso

1. Explica los conceptos de una metodología de desarrollo de software, en las fases de análisis y diseño.
2. Aplica las características de la metodología de desarrollo, los diagramas de modelamiento y los patrones de diseño en las fases de análisis y diseño del proceso de desarrollo de software.

D. Sistema y mecanismos de evaluación

De acuerdo con el Modelo Educativo de la UDLA la evaluación busca evidenciar el logro de los resultados de aprendizaje institucionales, de cada carrera y de cada asignatura, a través de mecanismos de evaluación (MdE). Por lo tanto, la evaluación debe ser continua, formativa y sumativa. La UDLA estipula la siguiente distribución porcentual para los reportes de evaluaciones previstas en cada semestre de acuerdo con el calendario académico:

Progreso 1: 25%

- a) Portafolio de Ejercicios Autónomo (5%): Recopilación de trabajos individuales autónomos de los estudiantes subidos al apoyo virtual

- b) Actividades en clase (10%): Recopilación de trabajos, talleres, laboratorios individuales o grupales realizados y subidos al apoyo virtual.
- c) Evaluación continua (10%): Examen (6%): Es una prueba multi-opción de los temas tratados durante el progreso para la teoría (3%), se desarrolla el análisis y diseño de un caso de estudio para la práctica (3%). Avance de Proyecto (4%): Para la evaluación final se considera la realización de un proyecto en el que los estudiantes desarrollen un tema asignado en el cual puedan demostrar el aprendizaje de los conceptos revisados durante la materia. El proyecto final será evaluado por medio de una rúbrica

Progreso 2: 35%

- a) Portafolio de Ejercicios Autónomo (10%): Recopilación de trabajos individuales autónomos de los estudiantes subidos al apoyo virtual.
- b) Actividades en clase (12.5%): Recopilación de trabajos, talleres. Laboratorios, individuales o grupales realizados y subidos al apoyo virtual.
- c) Evaluación continua (12.5%): Examen (5%), Es una prueba multi-opción de los temas tratados durante el progreso para la teoría (2%), se desarrolla el análisis y diseño de un caso de estudio para la práctica (3%). Avance de Proyecto (5%): Para la evaluación final se considera la realización de un proyecto en el que los estudiantes desarrollen un tema asignado en el cual puedan demostrar el aprendizaje de los conceptos revisados durante la materia. El proyecto final será evaluado por medio de una rúbrica. Presentaciones (2.5%): El estudiante debe realizar presentaciones de un tema asignado, se debe realizar en grupos de 2 estudiantes. Se calificará de acuerdo con la rúbrica de presentaciones.

Progreso 3: 40%

- a) Portafolio de Ejercicios Autónomo (10%): Recopilación de trabajos individuales autónomos de los estudiantes subidos al apoyo virtual.
- b) Actividades en clase (15%): Recopilación de trabajos individuales o grupales realizados y subidos al apoyo virtual.
- c) Evaluación continua (15%): Proyecto Final (8%): Para la evaluación final se considera la realización de un proyecto en el que los estudiantes desarrollen un tema asignado en el cual puedan demostrar el aprendizaje de los conceptos revisados durante la materia. El proyecto final será evaluado por medio de una rúbrica. Los Mejores Proyectos

se presentarán en UDLA Tech Day 2018 (23 enero 2018). Examen (7%), Es una prueba multi-opción de los temas tratados durante el progreso para la teoría (2%), se desarrolla el análisis y diseño de un caso de estudio para la práctica (5%).

E. Asistencia

Al finalizar el curso habrá un examen de recuperación para los estudiantes que, habiendo cumplido con más del 80% de asistencia presencial a clases, deseen reemplazar la nota de una evaluación anterior (el de mayor peso dentro de los componentes). Este examen debe integrar todos los conocimientos estudiados durante el periodo académico, por lo que será de alta exigencia y el estudiante necesitará prepararse con rigurosidad. La nota de este examen reemplazará a la evaluación que sustituye. Recordar que, para rendir el EXAMEN DE RECUPERACIÓN, es requisito que el estudiante haya asistido por lo menos al 80% del total de las sesiones programadas de la materia. No se podrá sustituir la nota de un examen previo en el que el estudiante haya sido sancionado por una falta grave, como copia o deshonestidad académica.

F. Metodología del curso.

Las metodologías y mecanismos de evaluación deben explicarse en los siguientes escenarios de aprendizaje:

F.1. Escenario de aprendizaje presencial.

- **Lecciones Escritas:** Son preguntas de elección múltiple que implican el estudio de cada tema de la materia. Un elemento de este mecanismo de evaluación podrá ser recuperada con el examen de recuperación
- **Portafolio:** Ejercicios y tareas prácticas semanales juntamente con el profesor: Conforman el portafolio que se desarrollará a lo largo del curso y deben ser enviados al apoyo virtual al finalizar cada semana. Las participaciones serán evaluadas de acuerdo con la rúbrica de ejercicios.
- **Examen final:** Son preguntas de elección múltiple que implican el estudio de toda la asignatura. Esta evaluación podrá ser recuperada con el examen de recuperación

F.2. Escenario de aprendizaje virtual.

- **Portafolio de Ejercicios:** El estudiante deberá resolver los ejercicios e investigaciones indicadas por el docente, ejercicios prácticos rendidos en clases y subirlos a la plataforma virtual.

F.3. Escenario de aprendizaje autónomo.

- **Portafolio:** Ejercicios y tareas prácticas semanales de trabajo autónomo: Conforman el portafolio que se desarrollará a lo largo del curso y deben ser enviados al apoyo virtual al finalizar cada semana. Las participaciones serán evaluadas de acuerdo con la rúbrica de ejercicios.
- **Proyecto Final:** Análisis y diseño de una aplicación definida en un caso de negocio presentado al inicio del semestre. Las actividades relacionadas serán desarrolladas utilizando una herramienta CASE seleccionada por cada grupo de manera tal que como resultado final se

obtenga código generado automáticamente que permita tener un punto de partida para el desarrollo posterior con el que se pueda atender las necesidades planteadas en el caso de negocio.

El proyecto será realizado en grupos de 2 estudiantes. Se calificará de acuerdo con la rúbrica de cada componente del proyecto integrador.

- **Presentaciones:** El estudiante debe realizar presentaciones de un tema asignado, se debe realizar en grupos de 2 estudiantes. Se calificará de acuerdo con la rúbrica de presentaciones.

G. Planificación alineada a los RdA

Planificación	Fechas	RdA 1	RdA 2
Unidad 1	Semana 1-4		
1. Análisis de sistemas de software 1.1 Modelos de proceso de software. 1.2 Metodología Rational Unified Process (RUP)			
Lecturas			
Campderrich, F. B. (2003). Ingeniería del software. Barcelona, ES: Editorial UOC. Retrieved from http://www.ebrary.com Capítulo 1		X	
Actividades			
(P)Investigación de opciones de Herramientas CASE Open Source (P)Taller: documento en base a las plantillas RUP (Rubrica de Portafolio de ejercicios) (P)Presentación del tema Análisis de sistemas de software por parte del docente. (A)Portafolio de ejercicios: Elaboración de documento de Modelo de proceso de software (A)Avance del Proyecto: Propuesta de una aplicación		X	
Evaluaciones			
Cuestionario RUP Documento de Avance del Proyecto de una propuesta de una aplicación – Elaboración del caso de negocio Documento en base a las plantillas RUP Examen Teórico-Practico		X	
Unidad 2	Semana 5-6		
2. Modelamiento en la fase de análisis de sistemas de software 2.1 Modelamiento del dominio del problema. 2.2 Modelamiento de requerimientos y creación del modelo de requerimientos del sistema. 2.3 Modelamientos de casos de uso y de las especificaciones casos de uso de un sistema. 2.4 Modelamiento de las reglas de negocio de un sistema			
Lecturas			
Campderrich, F. B. (2003). Ingeniería del software. Barcelona, ES: Editorial UOC. Retrieved from http://www.ebrary.com Capítulo 2			X
Actividades			
(P)Taller: Elaboración de documento de modelo Requerimientos (P)Taller: Elaboración de diagramas de casos de uso			X

(P)Taller: Elaboración de especificaciones de casos de uso basado en un problema (P)Presentación del tema Fase de análisis de requerimientos por parte del docente. (A)Portafolio ejercicios: diagrama de modelo Requerimientos-Problema propuesto (A)Portafolio ejercicios: diagramas de casos de uso-Problema propuesto (A)Portafolio ejercicios: especificaciones de casos de uso basado en un problema-Problema propuesto			
Evaluaciones			
Cuestionario multi-opción sobre análisis de requerimientos Portafolio ejercicios: diagrama de modelo Requerimientos-Problema propuesto (P y A) Portafolio ejercicios: diagramas de casos de uso (P y A) Portafolio ejercicios: especificaciones de casos de uso basado en un problema (P y A) Avance del Proyecto: Propuesta de una aplicación – Elaboración de documento de Análisis de Requerimientos Examen Teórico-Practico			X
Unidad 3 3. Modelamiento en la fase de diseño de sistemas de software 3.1 Modelamiento de actividades y estado 3.2 Modelamiento del diagrama de clases del sistema. 3.3 Estrategias de abstracción del diagrama de clases en función del modelo de entidad relación de un sistema. 3.4 Modelamiento interacción (diagramas de secuencia y colaborativo) 3.5 Modelamiento de componentes 3.6 Modelamiento de despliegue.	Semana 7-11		
Lecturas			
Campderrich, F. B. (2003). Ingeniería del software. Barcelona, ES: Editorial UOC. Retrieved from http://www.ebrary.com Capítulo 3			X
Actividades			
(P) Taller de elaboración de diagramas de actividades y de estado (P) Taller de elaboración de diagramas de clases basado en el problema propuesto (P) Taller de elaboración de diagramas de componentes y despliegue de un sistema (P) Taller de elaboración de diagramas de secuencia y colaboración de un sistema (P)Presentación del tema Fase de diseño de sistemas de software por parte del docente. (A)Portafolio ejercicios: diagramas de actividades y de estado-Problema propuesto (A)Portafolio ejercicios: diagramas de clases basado en el problema propuesto-Problema propuesto (A)Portafolio ejercicios: diagramas de componentes y despliegue de un sistema-Problema propuesto (A)Portafolio ejercicios: diagramas de secuencia y colaboración de un sistema-Problema propuesto			X
Evaluaciones			
Cuestionario multi-opción sobre diagramas de diseño UML Documento de diagramas de actividades y de estado (P y A)			X

Documento de diagramas de clases basado en el problema propuesto (P y A) Documento de diagramas de componentes y despliegue de un sistema (P y A) Documento de diagramas de secuencia y colaboración de un sistema (P y A) Avance del Proyecto: Documento de Diagramas de actividades, estado, clases, componentes, despliegue, secuencia y colaboración referente a la aplicación propuesta Examen Teórico-Practico			
Unidad 4 4. Patrones de Diseño 4.1 Patrones de diseño 4.2 Patrones de diseño estructurales. 4.3 Patrones de diseño creacionales 4.4 Patrones de diseño de comportamiento	Semana 12-13		
Lecturas https://msdn.microsoft.com/es-es/library/bb972240.aspx Pressman, R. (2015). Software Engineering – A Practitioner’s Approach. (8th edition). USA: McGraw Hill.			X
Actividades (A y P) Presentación: Diapositivas y diseño de diagramas de clase de ejemplo de los patrones de diseño de creación y comportamiento. (P)Presentación del tema Patrones de diseño creación, comportamiento y estructurales por parte del docente.			X
Evaluaciones Presentación del patrón de diseño asignado: diapositivas, documento resumen. Avance del Proyecto: Propuesta de una aplicación usando un patrón de diseño – Defensa del proyecto, Diapositivas, ejecución del proyecto. Examen Teórico-Practico			X
Unidad 5 5. Herramientas para desarrollo de sistemas de software 5.1 Herramientas CASE 5.2 Herramientas de diseño de aplicaciones	Semana 14-16		
Lecturas Landeros, G. R. P. (2009). Herramientas case. Córdoba, AR: El Cid Editor apuntes. Retrieved from http://www.ebrary.com Campderrich, F. B. (2003). Ingeniería del software. Barcelona, ES: Editorial UOC. Retrieved from http://www.ebrary.com Capítulo 1			X
Actividades (P)Taller de elaboración de documento de Arquitectura del Software (diseño de la aplicación) y código generado por la herramienta CASE seleccionada (A)Documento de análisis y diseño de la aplicación del proyecto final			X
Evaluaciones Proyecto Final: Propuesta de una aplicación – Elaboración del caso de negocio Portafolio de ejercicios: Documento del taller Arquitectura del Software (diseño de la aplicación) y código generado por la herramienta CASE Examen Teórico-Practico			X

H. Normas y procedimientos para el aula

1. Solo se recibirán trabajos en el aula virtual y dentro del plazo establecido.
2. Se tomaré lista en los primeros 10 minutos iniciada la clase si el estudiante llega después, podrá ingresar de forma silenciosa, pero no se registrará la asistencia, sin excepción.
3. Se tomaré lista en los últimos 10 minutos de la clase si el estudiante sale antes de tomar lista no se registrará la asistencia, sin excepción.
4. Los estudiantes deberán practicar la “honestidad académica” para todas las actividades de esta asignatura. La copia de ejercicios, exámenes, proyectos, y todas las actividades de aprendizaje solicitadas por el docente, se calificará con la mínima calificación (1.0).
5. Se acepta el uso de cualquier dispositivo electrónico (iPads, tablets, celulares, audífonos) únicamente con fines académicos. El uso para fines no académicos equivaldrá a una inasistencia.
6. No se podrán ingresar alimentos al aula.
7. El estudiante tiene derechos a recibir tutoría en los horarios establecidos por el docente.
8. En el caso de inasistencia es responsabilidad del estudiante igualarse en los contenidos de la materia dictada en dicha clase.
9. En el caso de que un estudiante falte a una sesión en la que se realicen pruebas o prácticas de laboratorio, no se podrán recuperar las calificaciones. Las fechas de las evaluaciones serán publicadas en el apoyo virtual de la materia.

I. Referencias

1. Principales.

Pressman, R. (2015). Software Engineering – A Practitioner’s Approach. (8th edition). USA: McGraw Hill.

2. Complementarias.

Kimmel, P. (2010). Manual de UML. (1era edición). Madrid, España: MCGRAW-HILL INTERAMERICANA EDITORES.

Sommerville, I. (2011). Ingeniería de software. (9na edición). Madrid, España: Pearson Educación

J. Perfil del docente

MSc. Santiago Villarreal

Maestría Informática especializada en Inteligencia web, Licenciado en informática de la Universidad Jean Monnet. Experiencia en el campo de la educación en la Universidad de las Américas (UDLA) con sede en Quito y en la Escuela Politécnica Nacional del Ecuador (EPN). He realizado investigación en el campo de la algoritmia, inteligencia artificial, bienestar de multi-agentes.

Contacto: oficina: sede Queri, bloque 4, primer piso, oficina de docentes frente a direcciones de carrera, puesto 36, mail: santiago.villarreal@udla.edu.ec

Horarios atención al estudiante disponible en el Aula virtual