



FAC. DE INGENIERIA Y C. AGROPECUARIAS
INGENIERÍA EN SISTEMAS DE COMPUTACIÓN E INFORMÁTICA
ACI120/ INTRODUCCIÓN A LOS SISTEMAS INFORMACIÓN
Periodo 2015 – 2
MARZO - JULIO 2015

1. Identificación

Número de sesiones: 64
Número total de hora de aprendizaje: 147h
Créditos – malla actual: 4
Profesor: Anita Yáñez
Correo electrónico del docente (Udlanet): a.yanez@udlanet.ec
Coordinador: Marco Galarza
Campus: Queri
Pre-requisito: Co-requisito:
Paralelo: 1

Tipo de asignatura:

Optativa	
Obligatoria	X
Práctica	

Organización curricular:

Unidad 1: Formación Básica	X
Unidad 2: Formación Profesional	
Unidad 3: Titulación	

Campo de formación:

Campo				
Fundamentos teóricos	Praxis profesional	Epistemología y metodología de la investigación	Integración de saberes, contextos y cultura	Comunicación y lenguajes
		X		

2. Descripción del curso.-

Es la primera materia dentro de la malla curricular en la línea de desarrollo de software que ofrece una visión introductoria para el desarrollo de un sistema de información con una connotación teórico – práctica de la funcionalidad cliente/servidor. Se revisará los paradigmas de la programación orientación a objetos y estructurada, programación web. Y además se revisará los nuevos lenguajes y tecnología de desarrollo, como son PHP, Python, Ruby, y programación en bloques.

3. Objetivo del curso.-

Identificar las principales características de los sistemas de información, los lenguajes de programación, las técnicas de programación y la arquitectura cliente/servidor para el desarrollo de software.

4. Resultados de aprendizaje deseados al finalizar el curso:

Resultados de aprendizaje (RdA)	RdA perfil de egreso de carrera	Nivel de dominio (carrera)
1. Identifica las	Aplica el enfoque sistémico en el	I__X__ M_____ F_____

características de los sistemas de información, y las consideraciones éticas del manejo de la información.	análisis y resolución de problemas relacionados con las TIC.	
2. Identifica los paradigmas de programación y las nuevas tecnologías de desarrollo de aplicaciones.	Aplica metodologías de investigación, pensamiento lógico, fundamentos matemáticos, principios algorítmicos y teorías de Ciencias de la Computación en la fundamentación, modelación y diseño de soluciones informáticas.	I__X__ M_____ F____
3. Identifica la arquitectura cliente/servidor para el desarrollo de aplicaciones de software.	Aplica metodologías de investigación, pensamiento lógico, fundamentos matemáticos, principios algorítmicos y teorías de Ciencias de la Computación en la fundamentación, modelación y diseño de soluciones informáticas.	I__X__ M_____ F____

5. Sistema de evaluación.-

De acuerdo al Modelo Educativo de la UDLA la evaluación busca evidenciar el logro de los resultados de aprendizaje (RdA) enunciados en cada carrera y asignatura, a través de mecanismos de evaluación (MdE). Por lo tanto la evaluación debe ser continua, formativa y sumativa.

La evaluación es continua, formativa y sumativa.

Reporte de progreso 1: 35%
 Reporte de progreso 2: 35%
 Evaluación final: 30%

Al finalizar el curso habrá un examen de recuperación para los estudiantes que deseen reemplazar la nota de un examen anterior (ningún otro tipo de evaluación). Este examen es de carácter complejo y de alta exigencia, por lo que el estudiante necesita prepararse con rigurosidad. La nota de este examen reemplazará a la del examen que sustituye. 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.

6. Metodología del curso y de mecanismos de evaluación.-

Las metodologías que se aplicarán para el desarrollo del curso son:

1. Lectura Previa: Para cada unidad el docente facilitará fuentes de información que el estudiante deberá leer previa la clase.
2. Prácticas individuales en laboratorio: Explicado el tema el docente, propone a los alumnos una actividad de aplicación de dichos contenidos.
3. Clase magistral: El docente presentará el tema a tratar de manera resumida, usando medios tecnológicos, (diapositivas), se usará la técnica pregunta – respuesta.
4. Trabajos Individuales: Que serán desarrollados dentro y fuera del aula para reforzar lo aprendido en clase.
5. Trabajo colaborativo: Los estudiantes realizarán trabajos en grupo dentro y fuera del aula de clases.



Componentes de la libreta de calificaciones

Progreso 1: 35%

El progreso 1 consta de las siguientes tareas con su porcentaje de evaluación:

- a) Portafolio de Ejercicios 20%, se adjuntará rúbrica.
- b) Prueba online 40%.
- c) Examen 40%

Progreso 2: 35%

El progreso 2 consta de las siguientes tareas con su porcentaje de evaluación:

- a) Portafolio de Ejercicios 10% se adjuntará rúbrica.
- b) Informe salida de campo 20%.
- c) Prueba online 30%.
- d) Examen 40%

Evaluación Final 30%

La evaluación final consta de las siguientes tareas con su porcentaje de evaluación.

- a) Portafolio de Ejercicios 10% se adjuntará rúbrica.
- b) Proyecto final 40% se adjuntará rúbrica.
- c) Evaluación final 50%

Cada ítem de evaluación contendrá la rúbrica respectiva, publicada en el apoyo virtual junto con tarea enviada.

7. Temas y subtemas del curso.-

RdA	Temas	Subtemas
1. Identifica las características de los sistemas de información, y las consideraciones éticas del manejo de la información.	1. Sistemas de información (1 semana)	1.1 Sistemas e Información
		1.2 Manejo de la información.
		1.3 Sistemas de información, componentes.
		1.4 Características de los SI
		1.5 Clasificación de los SI
		1.6 Consideraciones éticas para el manejo de la información.
2. Identifica los paradigmas de programación y las nuevas tecnologías de desarrollo de aplicaciones.	2. Conceptos Generales (3 semana)	2.1 Programa
		2.2 Historia de la programación
		2.3. Algoritmia, pseudocódigo básico
		2.4. Proceso de compilación
		2.5 Paradigmas de programación
		2.6. Lenguajes de programación
		2.7 Conocimientos básicos de algebra booleana.
		3.1 Visión general del paradigma de la programación estructurada.

	3. Introducción a la programación estructurada (3 semana)	3.2 Enfoque algorítmico en la solución de problemas de programación estructurada.
		3.3. Revisión de algoritmos
		3.4. Conocimiento básico de las estructuras de control del lenguaje de programación estructurado.
	4. Introducción a la programación orientada a objetos. (3 semana)	4.1 Visión general del paradigma de la programación orientada a objetos.
		4.2 Conocimiento básico de las estructuras de control del lenguaje de programación java para implementar programas orientados a objetos.
	5 Nuevos lenguajes y tecnología de desarrollo de aplicaciones (2semana)	5.1 Programación en bloques
		5.2 Programación Ruby
		5.3 Programación Python
		5.4 Programación PHP.
3. Identifica la arquitectura cliente/servidor para el desarrollo de aplicaciones de software.	5. Aplicaciones Cliente Servidor. (3 semana)	6.1 Programación HTML5
		6.2 Arquitectura cliente/ servidor
		6.2 Conocimiento de las herramientas para trabajar con formularios y servlets.
		6.3 Integración de formas con Servlets por medio de simulador del ambiente Cliente / Servidor, ICARNEGIE WORKBENCH.
		6.4 Conocimiento y manipulación (pruebas) básica de la parte conceptual para la escritura de los Servlets y programas en java en general.

8. Planificación secuencial del curso.-

SEMANA 1 (9-03 al 13-03)					
RdA	Tema	Sub tema	Actividad/ metodología/clase	Tarea/autónoma	MdE/Producto/
1	1.Sistemas de información	1.1 Sistemas e Información	Instrucción Directa	Lectura del libro Sistema de Información (Stair, Reynolds, 2010, pp. 3 a 79)	Cuestionario teórico: Sistemas de información Fecha: 3 de Abril.
		1.2 Manejo de la información.	Evaluación	Portafolio de ejercicios	Portafolio de ejercicios: Trabajo en grupo. Mapas conceptuales SI Fecha: 20 de Marzo
		1.3 Sistemas de información, componentes.	Trabajo colaborativo		Portafolio de ejercicios: Resolución del caso, Banca Móvil (Stair, Reynolds, 2010) Fecha: 27 de Marzo
		1.4 Características delos SI			
		1.5 Clasificación de los SI			
		1.6 Consideraciones éticas para el manejo de la información.			
SEMANA 2 - 4 (16-03 al 03-03)					
2.	2. Conceptos Generales	Programa	Instrucción Directa		Cuestionario teórico: Conceptos Básicos. Fecha: 30 de Marzo
		Historia de la programación	Trabajo colaborativo	Lectura del libro Programación Estructurada y Orientada a Objetos(López, 2011, Capítulo 1)	Portafolio de ejercicios Fecha: 03 de Abril

		Conocimientos básicos de álgebra booleana.	Evaluación	Portafolio de ejercicios Matemáticas discretas: aplicaciones y ejercicios (Villalpando, 2014, Capítulo 2)	
		Algoritmia, pseudocódigo básico		Portafolio de ejercicios Programación Estructurada y Orientada a Objetos(López, 2011, Capítulo 1)	Portafolio de ejercicios Fecha: 03 de Abril
		Proceso de compilación			
		Paradigmas de programación			
		Lenguajes de programación			
SEMANA 5 - 7 (30-04 al 17-04)					
2	2. Introducción a la programación estructurada	Visión general del paradigma de la programación estructurada.	Instrucción Directa	Lectura del libro Programación Estructurada y Orientada a Objetos(López, 2011, Capítulo 1)	Cuestionario teórico: Conceptos Básicos. Fecha: 30 de Marzo
		Enfoque algorítmico en la solución de problemas de programación estructurada.	Portafolio de Ejercicios	Portafolio de ejercicios Programación Estructurada y Orientada a Objetos(López, 2011, Capítulo 3,4,5)	Portafolio de ejercicios Fecha: 17 de Abril
		Revisión de algoritmos			
		Conocimiento básico de las estructuras de control del lenguaje del lenguaje de programación estructurado.			
EVALUACIÓN DEL PROGRESO 1 : 20 al 24 de Abril					
SEMANA 8-10 (27-04 al 15-05)					
	3. Introducción a la programación orientada	Visión general del paradigma de la programación orientada a objetos.	Instrucción Directa	Lectura del libro Programación Estructurada y Orientada a Objetos(López, 2011, Capítulo 10)	Cuestionario teórico: Conceptos Básicos. Fecha: 30 de Marzo

2	a objetos.	Conocimiento básico de las estructuras de control del lenguaje de programación java para implementar programas orientados a objetos.	Portafolio de Ejercicios	Portafolio de ejercicios Programación Estructurada y Orientada a Objetos(López, 2011, Capítulo 12)	Portafolio de ejercicios Fecha: 17 de Abril
			Visitas de Campo: Kruger Labs. (26 de Abril al 15 de Mayo)	Informe de Visita Fecha: 22 de Mayo	
SEMANA 11-13 (18-05 al 29-05)					
2	4. Nuevos lenguajes y tecnología de desarrollo de aplicaciones	Programación en bloques	Trabajo colaborativo	Investigación	Documento y exposición Fecha: 29 de Mayo
		Programación Ruby			
		Programación Python			
		Programación PHP.			
EVALUACIÓN DEL PROGRESO 2 : 1 al 5 de Junio					
SEMANA 14 a la 16 (8 de junio al 26 de Junio)					
3	6. Aplicaciones Cliente Servidor.	Programación HTML5	Instrucción Directa	Portafolio de ejercicios	Portafolio de ejercicios: Ejercicios enviados por el docente Fecha: 26 de Junio
		Arquitectura cliente/ servidor	Portafolio de Ejercicios		
		Conocimiento de las herramientas para trabajar con formas y servlets.	Trabajo colaborativo		
		Integración de formas con Servlets por medio de simulador del ambiente Cliente / Servidor, ICARNEGIE WORKBENCH.			



		Conocimiento y manipulación (pruebas) básica de la parte conceptual para la escritura de los Servlets y programas en java en general.			
EVALUACIÓN FINAL : 6 al 10 de Julio					
PRESENTACIÓN PROYECTO FINAL DEL 13 al 17 de Julio					

9. Observaciones generales.- Docente

1. No se permitirá entregar una tarea fuera del aula virtual y del plazo establecido.
2. Se tomará lista dentro de los primeros 5 minutos luego de iniciado cada módulo, si el estudiante llega después, podrá ingresar de forma silenciosa, pero no se registrará la asistencia
3. No se admitirá por ningún motivo la copia de ejercicios, exámenes, proyectos, y todas las actividades de aprendizaje solicitadas por el docente, y se calificará con la mínima calificación (cero).
4. Se restringe durante el desarrollo de clases el uso de celulares, redes sociales y audífonos
5. No se podrán ingresar alimentos al aula
6. El estudiante puede acceder a tutoría personal en los horarios establecidos por el docente
7. En el caso de inasistencia es responsabilidad del estudiante igualarse en los contenidos de la materia dictada en dicha clase
8. 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.

10. Referencias bibliográficas.- (sílabo maestro)

- Stair, R. , Reynolds, G. (2010). *Principios de sistemas de información*. (9a ed). Mexico, Mexico: Cengage Learning Editores. ISBN 978-0-324-66528-4 007
- López, L. (2011). *Programación Estructurada y Orientada a Objetos*. (3a ed) México, México: Alfaomega Grupo Editor. ISBN 978-607-707-211-9.
- Noguera, F., Riviera, D. (2013). *Programación*. (1a ed.) Barcelona, España: Editorial UOC. ISBN: 978-84-9788-110-4

10.1. Referencias complementarias.- Docente

- Villalpando, M. García, A., (2014). *Matemáticas discretas: aplicaciones y ejercicios*. México, Mexico. Larousse Grupo Editorial Patria. ISBN ebook: 978-607-438-920-3.
- Márquez, G. Osorio, S., & Olvera, N. (2010). *Introducción a la Programación Estructurada en C. (1a ed)* México, México: Pearson Educación. ISBN 978-607-32-0600-6