

Facultad de Ingeniería y Ciencias Agropecuarias
Ingeniería en Sistemas de Computación e Informática
ACI-120 - Introducción a los Sistemas de Información
Período 2017-1

1. Identificación (*Sílabo maestro*)

Número de sesiones:	64
Número total de hora de aprendizaje:	160h
Créditos – malla actual:	4
Profesor:	Santiago Villarreal
Correo electrónico del docente (Udlanet):	s.villarreal@udlanet.ec
Coordinador:	Marco Galarza
Campus:	Queri
Pre-requisito:	Co-requisito:
Paralelo:	3, 70,71

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

2. Descripción del curso

La materia abordará temas muy importantes y generales de la carrera, sin abarcar de manera profunda las temáticas. La asignatura 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 las nuevas herramientas de programación como la programación en bloques, para desarrollar la lógica de los estudiantes, los paradigmas de la programación orientación a objetos y estructurada, y además de una breve revisión de la programación web.

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, a través de nuevas herramientas de programación para que el estudiante desarrolle la lógica de la programación.

4. Resultados de aprendizaje deseados al finalizar el curso

Resultado de Aprendizaje Carrera De Ingeniería de Sistemas

Resultado de Aprendizaje Carrera De Ingeniería de Sistemas		
Resultados de aprendizaje (RdA)	RdA perfil de egreso de carrera	Nivel de desarrollo (carrera)
1. Identifica 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.	1. Aplica el enfoque sistémico en el análisis y resolución de problemas relacionados con las TIC.	Inicial (X) Medio () Final ()
Resultado de Aprendizaje Carrera De Ingeniería de Electrónica		
Resultados de aprendizaje (RdA)	RdA perfil de egreso de carrera	Nivel de desarrollo (carrera)
1. Identifica 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.	Identifica oportunidades para mejorar el desempeño de las comunicaciones en las organizaciones a través de la incorporación y uso eficiente de plataformas de servicios de redes.	Inicial () Medio (X) Final ()

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 UDLA estipula la siguiente distribución porcentual para los reportes de evaluaciones previstas en cada semestre de acuerdo al calendario académico:

Reporte de progreso 1: 35%

- a) Portafolio de Ejercicios 30%, se adjuntará rúbrica.
- b) Pruebas de evaluación continua 30%.
- c) Examen 40%

Reporte de progreso 2: 35%

- a) Portafolio de Ejercicios 30% se adjuntará rúbrica.
- b) Pruebas de evaluación continua 30%.
- c) Examen 40%

Evaluación final: 30%

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

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 un examen anterior (ningún otro tipo de evaluación). 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 del examen 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.

Proyecto Final: El proyecto final consiste en realizar un aplicativo usando programación en bloques (Scrath o AppInventor), la aplicativo debe ser innovador y creativo, puede ser un juego, animación o aplicación para Android. Este debe ser presentado al final del semestre.

6. Metodología del curso y de mecanismos de evaluación.

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

6.1. Escenario de aprendizaje presencial.

Para el aprendizaje presencial en el curso se realizará:

1. *Instrucción directa*: Los estudiantes recibirán explicación directa de los temas de la clase.
2. *Trabajo grupal*: Los estudiantes realizarán trabajos en grupo dentro del aula de clases. Todos los trabajos deben presentar su bibliografía académica que sustente su contenido, la evaluación será de manera individual de acuerdo a la rúbrica respectiva. Todos los trabajos formarán parte del portafolio de ejercicios.
3. *Trabajo individual*: Los estudiantes realizarán trabajos en el laboratorio de PC, los cuales van a ser dirigidos por el docente, su entrega va a ser al final de la clase y su evaluación de acuerdo a la rúbrica respectiva. Todos los trabajos formarán parte del portafolio de ejercicios.
4. *Evaluaciones*: Los estudiantes deberán rendir una evaluación por cada tema expuesto en clase, estos exámenes serán en el aula virtual y no tomarán más de 10 min. Todos los trabajos formarán parte del portafolio de evaluaciones. Además deberán rendir un examen al final de cada período (cátedras).

6.2. Escenario de aprendizaje virtual.

Para el aprendizaje virtual en el curso se realizará:

1. *Portafolio de ejercicios*. Recopilación de Trabajos individuales de los estudiantes subidos al apoyo virtual.
2. *Foros*. El estudiante debe aportar con ideas a foros virtuales en el apoyo virtual. Todos los foros formarán parte del portafolio de ejercicios.

6.3. Escenario de aprendizaje autónomo.

Para el aprendizaje autónomo en el curso se realizará:

1. *Prácticas individuales en laboratorio*: Explicado el tema el docente, propone a los alumnos una actividad de aplicación de dichos contenidos, las tareas deben ser subidos al apoyo virtual. Todos los trabajos formarán parte del portafolio de ejercicios.

2. *Portafolio de ejercicios*: El estudiante práctica los conocimientos y destrezas adquiridos. Todos los trabajos formarán parte del portafolio de ejercicios.
3. *Auto-evaluaciones y evaluaciones*: Se utilizará la herramienta de co-evaluación del aula virtual, la cual permite evidenciar el nivel de aprendizaje de los estudiantes, desarrollando en ellos responsabilidad y autonomía en las tareas enviadas.
4. *Proyecto final*: 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.

1. Temas y subtemas del curso

RdA	Temas	Subtemas
Identifica 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.	1. Introducción a los Sistemas de información (1 semana) (12/09 - 16/09) (Libro: Análisis y diseño de sistemas de información)	1.1 Definición de Sistemas 1.2 Definición de Ingeniería de Sistemas 1.3 Rol de la Ingeniería de sistemas 1.4 Campo de acción de Ingeniería de sistemas 1.5 Sistemas de Información y niveles dentro de la organización 1.6 Tipos de Sistemas de información 1.7 Ciclo de vida en el desarrollo de un Sistema de Información
	2. Algebra booleana (2 semana) (19/09 - 30/09)	2.1. Proposiciones 2.2. Conocimientos básicos de algebra booleana. 2.3. Tablas de verdad.
	3. Programación en bloques (3 semana) (03/10 - 21/10)	4.1. Scratch 4.2. App Inventor 4.3. Programación EV3
	4. Introducción a la programación (3 semana) (07/11 - 25/11)	3.1. El software y sus aplicaciones 3.2. Programas, algoritmos, lenguaje. 3.3. Proceso de programación, programa fuente y programa compilado 3.4. Pseudocódigo 3.5. Diagramas de flujo. 3.6 Paradigmas de programación 3.7. Lenguajes de programación 3.8. Revisión de algoritmos. Métodos formales para verificación de programas
	5. Introducción a la programación estructurada (2 semana) (28/11 -09/12)	5.1 Visión general del paradigma de la programación estructurada. Lenguaje de programación C 5.2. Conocimiento básico de las estructuras de control del lenguaje de programación en C para implementar programas estructurados.

	<p>6. Introducción a la programación orientada a objetos.</p> <p>(1 semana)</p> <p>(02/01/2017 - 06/01/2017)</p>	<p>6.1 Visión general del paradigma de la programación orientada a objetos.</p> <p>6.2 Conocimiento básico de las estructuras de control del lenguaje de programación java para implementar programas orientados a objetos.</p>
	<p>7. Introducción a los nuevos lenguajes de programación</p> <p>(1 Semana)</p> <p>(09/01 - 13/01)</p>	<p>7.1. Ruby</p> <p>7.2. Phyton</p>
	<p>8. Introducción a aplicaciones Cliente-Servidor.</p> <p>(1 semana)</p> <p>(16/01 - 20/01)</p>	<p>8.1 Programación HTML5</p> <p>8.2 Arquitectura cliente/ servidor</p> <p>8.2 Conocimiento de las herramientas para trabajar con formularios y servlets.</p>

2. Planificación secuencial del curso (Docente)

SEMANA 1: 12- 09-2016 al 16-09-2016)					
RdA	Tema	Sub tema	Actividad/ estrategia de clase	Tarea/autónoma	MdE/Producto/
1	1. Introducción a los Sistemas de información (1 semana)	<p>1.1 Definición de Sistemas</p> <p>1.2 Definición de Ingeniería de Sistemas</p> <p>1.3 Rol de la Ingeniería de sistemas</p> <p>1.4 Campo de acción de Ingeniería de sistemas</p> <p>1.5 Sistemas de Información y niveles dentro de la organización</p> <p>1.6 Tipos de Sistemas de información</p> <p>1.7 Ciclo de vida en el desarrollo de un Sistema de Información</p>	<p>Sociabilización del perfil de egreso y de las actividades del sílabo.</p> <p>(1) Instrucción Directa</p> <p>Video: Ingeniería de sistemas. https://www.youtube.com/watch?v=LD1020ceufQ</p> <p>Video: Todo el mundo debería saber programar https://www.youtube.com/watch?v=8lp20JFiB4s</p> <p>(1) Portafolio del Estudiante</p> <p>(1) Evaluación</p>	<p>(2) Foro Virtual, "Historia : Ingeniería de sistemas e informática" , "https://www.youtube.com/watch?v=_ttwVnbeZSQ"</p> <p>(2) Foro Virtual: "Como Steve Jobs Cambió el Mundo Documental IGenius Apple" https://www.youtube.com/watch?v=SV7Hg23gF-s</p> <p>Lectura del libro: Sistema de Información (Stair, Reynolds, 2010. Capítulo 1)</p> <p>Lectura del Paper: Introducción a los Sistemas de Información: Fundamentos (Francisco M. Gonzalez- Longatt)</p>	<p>Foro virtual: 1 intervenciones fundamentadas, sobre el Ingeniería de sistemas e informática. Tema: Porque seguimos Ingeniería de Sistemas, como nos vemos en 5 años. (Rúbrica para foros) Fecha: 20-09-2016</p> <p>Foro virtual: 1 intervenciones fundamentadas sobre: Como Steve Jobs Cambió el Mundo Documental IGenius Apple (tiempo 45 min) Plantear 5 proyectos científicos informáticos revolucionarios que cambiarían el mundo. (Rúbrica para foros) Fecha: 20-09-2016</p> <p>Evaluación: Resolución de juegos en el aula virtual Fecha: 20-09-2016</p>
SEMANA 2 y 3 : 19- 09-2016 al 30-09-2016)					
1	2. Algebra booleana (2 semana)	<p>2.1. Proposiciones</p> <p>2.2 Conocimientos básicos de algebra booleana.</p>	<p>(1) Instrucción Directa</p> <p>(1) Portafolio del Estudiante</p> <p>(1) Evaluación</p>	<p>Evaluaciones: Lectura del material publicado en el apoyo.</p> <p>Resolución de ejercicios.</p>	<p>Evaluación: Resolución de cuestionario en el aula virtual Fecha: 30-09-2016</p> <p>Portafolio de ejercicios: Trabajo individual Resolución</p>

		2.3 Tablas de verdad.			del ejercicios de algebra booleana. Matemáticas discretas: aplicaciones y ejercicios (Villalpando, 2014, Capítulo 2 ejercicios 2.1 al 2.5) Fecha: 30-09-2016
SEMANA 4 - 6: 03-10-2016 al 21-10-2016					
1	3. Programación en bloques (3 semana) (03/10 - 26/10)	4.1. Scratch 4.2. App Inventor 4.3. Programación EV3	(1) Instrucción Directa (1) Portafolio del Estudiante	Lectura del material publicado en el apoyo. Curso programación en bloques: https://studio.code.org/s/20-hour	Evaluación: Curso programación en bloques: https://studio.code.org/s/20-hour Fecha: 21-04-2016 Portafolio de ejercicios: Resolución de ejercicios publicados en el apoyo.
EVALUACIÓN CATEDRA 1 25-10-2016 / 28-10-2016					
REALIMENTACIÓN CATEDRA I Y REGISTRO DE NOTAS 25-10-2016 / 28-10-2016					
SEMANA 8- 10 : 01-11-2016 - 25-11-2016 (feriado de noviembre)					
1	4. Introducción a la programación (3 semana) (01/11 - 25/11)	3.1. El software y sus aplicaciones 3.2. Programas, algoritmos, lenguaje. 3.3. Proceso de programación, programa fuente y programa compilado 3.4. Pseudocódigo 3.5. Diagramas de flujo. 3.6 Paradigmas de programación 3.7. Lenguajes de programación	(1) Instrucción Directa (1) Evaluación	Resumen video: "Bill Gates / El Sultán del Software - Informática [Documental]" Lectura del libro: Introducción a la Programación Lógica y Diseño (Farrell 2013, Capítulo 1)	Foro virtual: 1 intervenciones fundamentadas sobre: Bill Gates / El Sultán del Software – Informática. Documental (tiempo 44 min) Plantear 2 proyectos de software innovadores. (Rúbrica para foros) Fecha: 11-11-2016 Portafolio de ejercicios: Resolución de problemas algorítmicos. Fecha: 25-11-2016 Evaluación: Resolución de cuestionario en el aula virtual Fecha: 22-11-2016

		3.8. Revisión de algoritmos. Métodos formales para verificación de programas		Lectura del libro: Programación(Noguera – Riera, 2010, Capítulo 4)	
SEMANA 11 -12: 28-11-2016 - 09-12-2016					
1	5. Introducción a la programación estructurada (2 semana) (28/11 - 09/12)	5.1 Visión general del paradigma de la programación estructurada.	(1) Instrucción Directa	Lectura del libro: Programación Estructurada y Orientada a Objetos(López, 2011, Capítulo 1)	Portafolio de ejercicios: Resolución del ejercicios sobre Programación estructurada (López, 2011, Capítulo 3,4,5) Fecha: 12-05-2016
		5.2. Diagramas de flujo	(1) Evaluación		
		5.3. Pseudocódigo	(1) Portafolio del Estudiante	Lectura del libro: Fundamentos de programación Piensa e C(Cairo, 2006, Capítulo 1)	Evaluación: Resolución de cuestionario: Introducción a la programación estructurada Fecha: 12-05-2016
		5.4. Revisión de algoritmos. Métodos formales para verificación de programas			
EVALUACIÓN CATEDRA II 12-12-2016 / 16-12-2016					
REALIMENTACIÓN CATEDRA II Y REGISTRO DE NOTAS 12-12-2016 / 16-12-2016					
SEMANA 14: 02-01-2017 - 06-01-2017					
1	6. Introducción a la programación orientada a objetos. (1 semana) (02/01/2017 - 06/01/2017)	6.1 Visión general del paradigma de la programación orientada a objetos.	(1) Instrucción Directa	Lectura del libro Programación Estructurada y Orientada a Objetos(López, 2011, Capítulo 10)	Portafolio de ejercicios: Resolución del ejercicios sobre Programación Estructurada y Orientada a Objetos(López, 2011, Capítulo 10) Fecha: 06-01-2017
		6.2 Conocimiento básico de las estructuras de control del lenguaje de programación java para implementar programas orientados a objetos.	(1) Evaluación (1) Portafolio del Estudiante		
SEMANA 15: 09-01-2017 - 13-01-2017					
1	7. Introducción a los nuevos lenguajes de programación (1 semana) (09/01/2017 - 13/01/2017)	7.1 Programación Ruby	(1) Instrucción Directa	Lectura del material publicado en el aula virtual.	Exposición grupal sobre los nuevos lenguajes de programación. Fecha: 13-01-2017
		7.2 Programación Python	(1) Evaluación (1) Portafolio del Estudiante		
SEMANA 16: 16-01-2017 - 20-01-2017					
1	8. Introducción a	8.1 Programación HTML5	(1) Instrucción Directa	Lectura del material publicado	Portafolio de ejercicios: Solución de ejercicios enviados

aplicaciones Cliente Servidor. (16/01/2017 - 20/01/2017)	8.2Arquitectura cliente/ servidor	(1) Evaluación	en el aula virtual.	por el docente Fecha: 16-01-2017
	8.3 Conocimiento de las herramientas para trabajar con formularios y servlets.	(1) Portafolio del Estudiante	Resolución de ejercicios	

3. Normas y procedimientos para el aula (Docente)

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, y se calificará con la mínima calificación (cero).
5. Se acepta el uso de cualquier dispositivo electrónico (ipads, tablets, celulares, audifonos) 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.

4. Referencias bibliográficas (Docente)

a. Principales.

1. Stair, R. , Reynolds, G. (2010). *Principios de sistemas de información*. (9a ed). Mexico, Mexico: Cengage Learning Editores. ISBN 978-0-324-66528-4 007
2. Villalpando, J. Garcia, A. (2014) *Matemáticas Discretas: Aplicaciones y ejercicios*. (1ª ed). México, México D.F.. Larousse - Grupo Editorial Patria. ISBN ebook: 978-607-438-925-8.
3. López, L. (2011). *Programación Estructurada y Orientada a Objetos*. (3a ed) México, México: Alfaomega Grupo Editor. ISBN 978-607-707-211-9.

b. Referencias complementarias.

1. Noguera, F., Riviera, D. (2013). *Programación*. (1a ed.) Barcelona, España: Editorial UOC. ISBN: 978-84-9788-110-4

2. 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
3. Torres, C. Augusto S. (2010). *Lógica matemática para ingeniería de sistemas y computación. (1ª ed)* España, Madrid. Ediciones Elizcom. ISBN: 978-958-99325-9-9
4. Perez Aguila, R. (2013). *Una introducción a las matemáticas discretas y teoría de grafos. (1ª ed)*. México, México D.F. El Cid Editor. ISBN 978-1-4492-9128-0.

a. Perfil del docente

Nombre del docente: 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: s.villarreal@udlanet.ec teléfono: (02) 3981000 ext. 7931

Horarios atención al estudiante disponible en el Aula virtual