



DATOS DE IDENTIFICACIÓN DEL CURSO

DEPARTAMENTO:	Departamento de Ciencias Computacionales.			
ACADEMIA A LA QUE PERTENECE:	Ingeniería de Software			
NOMBRE DE LA MATERIA:	Taller de ingeniería de software II			
CLAVE DE LA MATERIA:	CC306			
CARÁCTER DEL CURSO:	Especializante Selectiva.			
TIPO DE CURSO:	Taller.			
No. DE CRÉDITOS:	4			
No. DE HORAS TOTALES:	60	Presencial	51	No Presencial 9
ANTECEDENTES:	CC304 – Ingeniería de Software I			
CONSECUENTES:	No Aplica			
CARRERAS EN QUE SE IMPARTE:	Licenciatura en Informática e Ingeniería en Computación.			
FECHA DE ULTIMA REVISIÓN:	12 de diciembre del 2016			

PROPÓSITO GENERAL

El propósito de este curso consiste en que el estudiante aplique los aspectos teóricos y prácticos de RUP resolviendo problemáticas de la vida real a través de los conceptos fundamentales y prácticas y/o proyectos. Como resultado del curso el alumno desarrollará habilidades que le permitan utilizar RUP como una metodología de desarrollo de proyectos de Software, además de que se familiarice con la notación de UML a través de uso de estándares y herramientas CASE para su implementación en los proyectos de software.

OBJETIVO TERMINAL

El alumno aplicará RUP, UML y herramientas CASE para modelar, analizar, diseñar, y documentar aplicaciones de Software.

CONOCIMIENTOS PREVIOS

Lógica, Bases de Programación, Paradigma Orientada a Objetos, UML, Requerimientos de Software.

HABILIDADES Y DESTREZAS A DESARROLLAR

Resolver, abstraer, sintetizar, aplicar, desarrollar, analizar, diseñar, modelar, documentar, manejar al menos una herramienta CASE.

ACTITUDES Y VALORES A FOMENTAR

Puntualidad, responsabilidad, compromiso, disciplina.

METODOLOGÍA DE ENSEÑANZA APRENDIZAJE

Método	Método tradicional de exposición	Método Audiovisual	Aula Interactiva	Multimedia	Desarrollo de proyecto	Dinámicas	Estudio de casos	Otros (Especificar)
%	20	15	10	0	30	10	15	0



UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E INGENIERÍAS
DIVISIÓN DE ELECTRÓNICA Y COMPUTACIÓN



RECURSOS METODOLOGICOS

RECURSOS METODOLOGICOS

En este curso-taller el alumno podrá hacer uso de las siguientes herramientas para el desarrollo de sus prácticas y proyecto:

- Office
- Project
- Visio 2003
- Smart Draw
- Rational Software Architect
- Día (Linux)
- Easy Case
- Sharp Develop
- Java
- C++ para Linux y para Windows

Esto a través del software instalado en el laboratorio de cómputo asignado al taller.

RECURSOS NECESARIOS

- Cañón
- Pintarrón
- Marcadores
- Borrador
- Labtop

RECURSOS NECESARIOS



CONTENIDO TEMÁTICO

MODULO 1. HERRAMIENTAS CASE.		3 HRS
El alumno comprenderá cómo la automatización de la ingeniería del software ha evolucionado y su impacto en el ciclo de vida del software.		
1.1	Conceptualización de CASE	1 HRS
	El alumno describirá a las herramientas CASE a través de sus conceptos y características fundamentales.	
1.1.1	Definición de CASE	15 MIN
	El alumno definirá el concepto de herramienta CASE a partir del significado de las siglas.	
1.1.2	Componentes de una herramienta CASE.	45 MIN
	El alumno distinguirá los componentes de las herramientas CASE.	
1.2	Tipos y entornos Case	1 HRS
1.2.1	Clasificación de las herramientas CASE	15 MIN
	El alumno comprenderá las características de los diferentes tipos de herramientas CASE.	
1.2.2	Entornos CASE integrados	15 MIN
	El alumno enumerará un conjunto de diferentes herramientas CASE.	
1.2.3	Evaluación y selección de herramientas CASE	30 MIN
	El alumno revisará las características de diferentes tipos de herramientas CASE mediante el análisis de las características planteadas en clase.	
1.3	Rational Rose Enterprise como herramienta CASE	1 HRS
	El alumno distinguirá las características de la herramienta CASE Rational Rose Enterprise.	
MODULO 2. Rational Unified Process (RUP)		3 HRS
El alumno identificará a RUP como una metodología de desarrollo de proyectos de Software. Además se familiarizará con la notación de UML y la relación entre RUP, UML y Rational Rose ENTERPRISE.		
2.1	Conceptualización de RUP	1 HRS
	El alumno comprenderá qué es RUP, cómo ayuda al desarrollo de Software de aplicaciones, y las fases y disciplinas que contiene.	



2.1.1	Definición y conceptos principales de RUP. El alumno definirá RUP a partir de sus siglas y comprenderá los conceptos principales.	30 MIN
2.1.2	Conceptualización de fases y disciplinas de RUP. El alumno debatirá las características de las fases y disciplinas de RUP con respecto a otras metodologías.	30 MIN
2.2	Conceptualización de UML	1 HRS
2.2.1	Definición El alumno definirá UML a partir de sus siglas.	10 MIN
2.2.2	Conceptos El alumno recordará los principales conceptos de UML tales como: actor, casos de uso, clases, etc.	20 MIN
2.2.3	Vistas El alumno distinguirá las definiciones y características de las diferentes vistas que pueden generarse con UML.	30 MIN
2.3	Relación entre Rational Rose Enterprise, UML y RUP El alumno comprenderá la relación entre Rational Rose Enterprise, UML y RUP durante el proceso de desarrollo de software.	1 HRS
MODULO 3. Fase de Concepción		12 HRS
El alumno comprenderá cómo se adquieren los requerimientos por parte de los usuarios para consolidar una visión única de los objetivos y alcances de un proyecto de software. También, definirá la visión del proyecto de software a desarrollar en el taller.		
3.1	Conceptos básicos El alumno entenderá la definición, objetivos y actividades involucradas en la fase de concepción.	1 HRS
3.1.1	Definición de la fase de concepción El alumno definirá la fase de concepción de acuerdo a la información proporcionada en clase.	10 MIN
3.1.2	Objetivo de la fase de concepción El alumno conceptualizará el objetivo de la fase de concepción de acuerdo a RUP.	5 MIN
3.1.3	Actividades a realizar durante la fase de concepción El alumno entenderá y enumerará las actividades que deben realizarse durante la fase de concepción basándose en RUP.	45 MIN



3.2	Notación UML		1 HRS
	El alumno se familiarizará con las vistas y diagramas a elaborar en la fase de concepción. Especialmente con la notación de diagramas de casos de uso de negocio (Business Use Case Model).		
3.3	Documentos a Elaborar		10 HRS
	El alumno elaborará los documentos que permiten registrar los modelos y descripciones correspondientes a la fase de concepción a través de RUP.		
3.3.1	Visión		4 HRS
	El alumno desarrollará cada elemento contenido en el documento de visión de forma teórico-práctica, el cuál captura las características sobre el proyecto de software a realizar y su entorno de aplicación.		
3.3.2	Contrato		3 HRS
	El alumno desarrollará cada elemento de un contrato de desarrollo de software de manera teórico-práctica.		
3.3.3	Business Use Case Model		3 HRS
	El alumno desarrollará por lo menos un caso de estudio para modelar un diagrama de caso de uso desde la perspectiva de negocio de manera teórico-práctica.		
MODULO 4. Fase de Elaboración.			15 HRS
El alumno definirá la arquitectura de sistemas de software a través de la comprensión del proceso de análisis y diseño en RUP usando la notación UML.			
4.1	Conceptos básicos		2 HRS
	El alumno entenderá la definición, objetivos y actividades involucradas en la fase de elaboración.		
4.1.1	Definición de la fase de elaboración		15 MIN
	El alumno definirá la fase de elaboración de acuerdo a la información proporcionada en clase.		
4.1.2	Objetivo de la fase de elaboración		45 MIN
	El alumno conceptualizará el objetivo de la fase de elaboración de acuerdo a RUP. Distinguiendo dos fases principales análisis y diseño.		
4.1.3	Actividades a realizar durante la fase de elaboración		1 HRS
	El alumno entenderá y enumerará las actividades que deben realizarse durante la fase de elaboración basándose en RUP.		



4.2	Notación UML		1 HRS
	El alumno se familiarizará con las vistas y diagramas a elaborar en la fase de elaboración. Especialmente con la notación de diagramas de casos de uso, diagramas de clases y objetos, diagramas de iteración (Colaboración y Secuencia), y los diagramas de estados y actividades.		
4.3	Documentos a Elaborar		12 HRS
	El alumno elaborará los documentos que permiten registrar los modelos y descripciones correspondientes a la fase de concepción a través de RUP.		
	4.3.1	Fase de análisis	4 HRS
		El alumno desarrollará por lo menos un diagrama de Use Case Model, y un diagrama de clases y objetos (modelo conceptual) a través de un caso de estudio, permitiéndole realizar actividades teórico-prácticas.	
	4.3.2	Fase de Diseño	8 HRS
		El alumno desarrollará por lo menos un diagrama de colaboración, secuencia y un diagrama de estados y actividades a través de un caso de estudio, permitiéndole realizar actividades teórico-prácticas	
MODULO 5. Fase de Construcción.			15 HRS
El alumno podrá definir la versión operacional de aplicaciones de software, y las pruebas que se utilizarán para determinar la calidad del mismo.			
5.1	Conceptos básicos		2 HRS
	El alumno entenderá la definición, objetivos y actividades involucradas en la fase de construcción.		
	5.1.1	Definición de la fase de construcción	15 MIN
		El alumno definirá la fase de construcción de acuerdo a la información proporcionada en clase.	
	5.1.2	Objetivo de la fase de construcción	45 MIN
		El alumno conceptualizará el objetivo de la fase de construcción de acuerdo a RUP. Distinguiendo dos fases principales implementación y pruebas.	
	5.1.3	Actividades a realizar durante la fase de construcción	1 HRS
		El alumno entenderá y enumerará las actividades que	



		deben realizarse durante la fase de construcción basándose en RUP.	
5.2	Notación UML		1 HRS
	El alumno se familiarizará con las vistas y diagramas a elaborar en la fase de elaboración. Especialmente con la notación de diagramas de componentes y de despliegue.		
5.3	Documentos a Elaborar		12 HRS
	El alumno elaborará los documentos que permiten registrar los modelos y descripciones correspondientes a la fase de construcción a través de RUP.		
	5.3.1	Fase de implementación	9 HRS
		El alumno desarrollará por lo menos un diagrama de componentes y de despliegue correspondiente a la vista física, permitiéndole realizar actividades teórico-prácticas entre las cuales se encuentran la generación de código y el proceso de ingeniería inversa.	
	5.3.2	Fase de pruebas	3 HRS
		El alumno documentará por lo menos un caso de prueba, permitiéndole realizar actividades teórico-prácticas.	

CRITERIOS DE EVALUACIÓN

- 50% Prácticas de laboratorio, tareas, e investigaciones, todo promediado.
- 50% Proyecto final dividido de la siguiente forma.
 - 15% Conceptualización y Análisis
 - 15% Diseño
 - 20% Implementación

BIBLIOGRAFÍA

BÁSICA

TITULO	AUTOR	EDITORIAL	AÑO DE EDICIÓN	% DE COBERTURA DEL CURSO
UML : modelado de software para profesionales	Fontela, Carlos.	Alfaomega	2011	40%
Manual de Prácticas para la asignatura de Taller de Ingeniería de Software II.	Santoyo-Sanchez, Alejandra.	Departamento de Ciencias Computacionales	2008	60%



Manual de UML.	Kimmel, Paul	McGraw-Hill	2007	40%
COMPLEMENTARIA				
TITULO	AUTOR	EDITORIAL	AÑO DE EDICIÓN	% DE COBERTURA DEL CURSO
Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures	Hassan Gomaa	Cambridge University Press	2011	20%
Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise (Agile Software Development Series)	Dean Leffingwell	Addison-Wesley	2002	20%
Utilización de UML	Booch Grady	Addison-Wesley	2007	60%

OTRAS REFERENCIAS	
DIRECCIÓN	% DE COBERTURA DEL CURSO
http://www.rational.com/uml	10%
http://www.sei.cmu.edu/	10%

REVISIÓN REALIZADA POR:

NOMBRE DEL PROFESOR	FIRMA
<u>MC María Isabel Cibrian Decena</u>	_____
<u>ME Hassem Rubén Macías Brambila</u>	_____
<u>MDH Myriam Azucena Nuño Gómez</u>	_____

Vo.Bo. Presidente de Academia

MC Angel Tonatiuh Hernández Casas

Vo.Bo. Jefe del Departamento

Dr. Carlos Alberto López Franco