

**FORMATO PROPUESTA DE DESARROLLO PROGRAMA DE CURSO****CODIGO:**
FO-M-DC-05-01**VERSION:**
2**FECHA:**
2010-04-19**PAGINA:**
1 de 4**1. IDENTIFICACIÓN**

Nombre de la Asignatura DISEÑO DE SISTEMAS DE INFORMACIÓN		Código 72020601		Área Ingeniería Aplicada
Naturaleza Teórico– Práctica	No de Créditos 3	TP Trabajo Presencial 3	TD Trabajo Dirigido 2	TI Trabajo Independiente 4
Semestre Sexto	Duración 80 horas	Habilitable No	Homologable Si	Validable No

PRE-REQUISITO: No aplica al plan de estudios actual.**2. JUSTIFICACIÓN**

Formar al Ingeniero de Sistemas de la Universidad de la Amazonia, para el uso de metodologías, técnicas y herramientas en el diseño de software para avanzar en los criterios de calidad en la producción de software de forma industrial. El diseño es un área de trabajo rigurosa y con la necesidad de tener conocimientos en programación, análisis e implementación para lograr establecer los elementos necesarios desde ámbito teórico para luego llevarlos a producción.

3. COMPETENCIAS**3.1 Competencias Generales**

Debe poder diseñar soluciones de software a nivel de arquitectura, mediante la aplicación de metodologías, lenguajes, modelos y estándares de calidad, basados en UML aplicables a entornos empresariales.

3.2 Competencias Específicas

- ✓ Producir aplicaciones de software a nivel de diseño, haciendo uso de las mejores prácticas que aseguren la calidad del producto bajo el lenguaje de Modelamiento Unificado UML.
- ✓ Diseñar soluciones de software a través de la aplicación de metodologías, fundamentos del diseño, diseño modular, diseño arquitectónico, diseño procedimental y diseño orientado a los objetos,
- ✓ Interpretar los diseños para generar soluciones prácticas en el ámbito del desarrollo y aplicarlos a un entorno problemático real.

4. OBJETIVOS

- Presentar las nuevas tendencias de la Ingeniería del Software con objetos y componentes,



FORMATO PROPUESTA DE DESARROLLO PROGRAMA DE CURSO

CODIGO:
FO-M-DC-05-01

VERSION:
2

FECHA:
2010-04-19

PAGINA:
2 de 4

- utilizando una flotación estándar con UML. Presentar las nuevas tendencias de la Ingeniería del Software con objetos y componentes, utilizando una flotación estándar con UML.
- Brindar elementos teóricos y prácticos para la elección y reconocimiento de patrones arquitecturales de aplicaciones de software.

5. CONTENIDO TEMÁTICO Y ANÁLISIS DE CRÉDITOS

Contenido temático

Proyecto de desarrollo de software apoyado con UML.

Unidad 1. Arquitecturas de Software

Arquitectura de 3 Capas

Arquitectura .Net

Arquitectura J2EE

Unidad 2. Diseño con UML

Definición de UML y aplicaciones

Diagrama de Clases

Diagrama de Secuencia

Diagrama de Actividades

Unidad 3. Patrones de diseño creacionales y estructurales

Builder y Abstrac Factory

Factory Method y Prototype y Singleton

Adapter, Bridge, Composite y Decorator

Façade, Proxy y Flyweight

Unidad 4. Patrones de diseño comportamiento

Chain Responsibility, Command, Interpreter

Iterator, Mediator, Memento y Observer

State, Strategy, Template Method y Visitor

Análisis de Créditos

TEMAS	TRABAJO PRESENCIAL	TRABAJO DIRIGIDO	TRABAJO INDEPENDIENTE
Unidad 1. Arquitecturas de Software			
Arquitectura de 3 Capas	3	2	4
Arquitectura .Net	3	2	4
Arquitectura J2EE	4	2	6
Unidad 2. Diseño con UML			
Definición de UML y aplicaciones	3	2	4
Diagrama de Clases	5	4	6
Diagrama de Secuencia	3	2	4
Diagrama de Actividades	3	2	4
Unidad 3. Patrones de diseño creacionales y estructurales			

**FORMATO PROPUESTA DE DESARROLLO PROGRAMA DE CURSO**CODIGO:
FO-M-DC-05-01VERSION:
2FECHA:
2010-04-19PAGINA:
3 de 4

Builder y Abstrac Factory	3	2	4
Factory Method y Prototype y Singleton	3	2	4
Adapter, Bridge, Composite y Decorator	3	2	4
Façade, Proxy y Flyweight	3	2	4
Unidad 4. Patrones de diseño comportamiento			
Chain Responsibility, Command, Interpreter	3	2	4
Iterator, Mediator, Memento y Observer	4	3	6
State, Strategy, Template Method y Visitor	5	3	6
TOTAL DE HORAS DEL CURSO	48	32	64
TOTAL CRÉDITOS:	3		

6. Estrategias Metodológicas**Trabajo presencial:**

El desarrollo de la materia se hará fundamentalmente, con clases magistrales, en las que se presentará la temática conceptual. Se soportará esta teoría con prácticas sobre herramientas que soportan diseño con UML.

Trabajo dirigido:

Elaboración de talleres de diseño en clase con el objetivo de afianzar los conceptos teóricos

Trabajo independiente:

- Consultas y lecturas para evaluar la profundidad de los conocimientos impartidos en el trabajo dirigido.
- Desarrollo de un proyecto de software a la par del curso aplicándole todas las temáticas vista en el curso para entregarlo como trabajo final.

La metodología planteada, se establece semestralmente de común acuerdo entre los estudiantes y el docente (Acuerdo Pedagógico).

7. RECURSOS.

Sala de clase, sala de computadores, software de apoyo a UML, Videobeam

8. EVALUACIÓN

La valoración final del estudiante; estará comprendida entre uno coma cero (1.0) y cinco coma cero (5,0) siendo la aprobación igual o superior a tres cero (3,0). Se obtendrá mediante la suma de las evaluaciones parciales (70%) y finales (30%), de la siguiente manera:

- 70%: La evaluación se sujeta al acuerdo pedagógico de la asignatura, con el fin de verificar los logros y competencias alcanzados a través de su estudio.
- 30%: La evaluación se sujeta al acuerdo pedagógico de la asignatura, con el fin de verificar los logros y competencias alcanzados a través de su estudio



UNIVERSIDAD DE LA
AMAZONIA

FORMATO PROPUESTA DE DESARROLLO PROGRAMA DE CURSO

CODIGO:
FO-M-DC-05-01

VERSION:
2

FECHA:
2010-04-19

PAGINA:
4 de 4

9. BIBLIOGRAFÍA

BOOCH, Grady. Análisis y Diseño Orientado a Objetos. Editorial Addison Wesley. Diaz de Santos. 1996

BOOCH, Grady- RUMBAUCH, James-JACOBSON, Ivar. El Lenguaje Unificado de Modelado. Editorial Addison Wesley. 1999

COAD, Peter y YQURDON, Edward. Object Oriented Analysis. Prentice Hall. 1990

McCONNELL, Steve. Desarrollo y Gestión de Proyectos Informáticos. Editorial McGraw-Hill.

PRESSMAN, Roger. Ingeniería del software. Un enfoque práctico. Cuarta Edición. Editorial McGraw-Hill.

ROBERT C., Martín. Object Oriented C++ Application Using The Booch Method. Editorial Prentice Hall.

STEVENS, Perdita-POOLEY, Rob. Utilización de UML en Ingeniería del software con objetos y componentes.

Fuentes electrónicas:

http://www.infor.uva.es/~mla/juna/isl/materiales/metodologia_analisis.pdf 14 de febrero de 2011 última revisión.