

## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: ARQUITECTURAS ORIENTADAS A SERVICIOS

Línea de la Asignatura: Ingeniería de Software

Tiempo de dedicación del estudiante a las actividades de:

DOC-TIS-TPS-HRSTOT-CREDITOS

(48 - 20 - 100 - 168 - 6)

### 1. Historial de la asignatura

Fecha de revisión / actualización	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preliminar Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba 8 al 12 de julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba 9 al 13 de enero de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

### 2. Pre-requisitos y correquisitos.

Prerrequisitos	
Asignatura	Temas

Correquisitos	
Asignatura	Temas

	Programación Orientada a Objetos  Manejo básico de bases de datos  Ingeniería de Software  JAVA	Ninguna	Ninguna
--	----------------------------------------------------------------------------------------------------------------------------	---------	---------

### 3. Objetivo de la asignatura

Aplicar los conocimientos necesarios para el desarrollo y publicación de servicios web seguros a partir de componentes creados en Java, así como trabajar con una unión de componentes antiguos y nuevos a partir de los servicios web.

### 4. Aportación al perfil del graduado.

Este curso proporciona al egresado los elementos necesarios para solucionar problemas prácticos con el desarrollo de aplicaciones basadas en servicios Web, permitiéndole construir aplicaciones robustas bajo el enfoque de tecnologías emergentes

### 5. Contenido temático.

Unidad	Tema	Subtemas
1.	Integración Basada en Arquitectura Orientada a Servicios  DOC: 10  Objetivo: Conocer los conceptos básicos de la arquitectura orientada a servicios	1.1. Computación Orientada a Servicios 1.2. La Arquitectura Orientada a Servicios 1.3. Modelos de servicio 1.4. Integración de aplicaciones: EAI vs. Middleware. El bus empresarial de servicios (ESB) como backbone de integración 1.5. Ingeniería del Software Orientado a Servicios
2	Lenguajes de Descripción de Servicios Web  DOC: 10	2.1. eXtensible Markup Language (XML) 2.2. XML Namespaces 2.3. XML Schema Definition

	Objetivo: Conocer los elementos básicos de los lenguajes basados en XML para la descripción de servicios Web	Language (XSD) 2.4. Expresiones XPath 2.5. Otras tecnologías (XQuery, XSL-T, etc.) 2.6. WSDL
3	Desarrollo de servicios Web.  DOC: 10  Objetivo: Conocer la pila tecnológica de los servicios Web	3.1. La Arquitectura de Servicios Web 3.2. Definición de servicios: WSDL 3.3. Formato de mensajes y protocolo de mensajería: SOAP 3.4. Descripción, publicación, descubrimiento e integración: UDDI 3.5. Invocación de servicios. Gestión del estado del servicio 3.6. Soporte de la plataforma Java EE para el desarrollo de Servicios Web: JAX-WS, OSGi, etc.
4	Implementando SOA con Java EE: Web Tier y Bussiness-Tier  DOC: 10  Objetivo: Desarrollar servicios Web en Java Enterprise Edition	4.1. El estilo arquitectónico REST 4.2. La Arquitectura Orientada a Recursos (ROA) 4.3. Definición del modelo de recursos 4.4. Diseño de URIs y Clientes REST 4.5. Formatos de representación de recursos: XML vs. JSON, Atom 4.6. Diseño e implementación de servicios RESTful orientados a recursos 4.7. Mashups de servicios 4.8. Soporte de la plataforma Java EE para el desarrollo de Servicios Web RESTful: JAX-RS 4.9. Marcos de trabajo: Restlet, Ruby on Rails, Django
5	Casos de Estudio SOA/Calidad de Servicio  DOC: 8	5.1. Introducción a los procesos de negocio y a su gestión/monitorización 5.2. Especificación de procesos de negocio

	Objetivo: Identificar casos de estudio para la composición y calidad de servicios Web	5.3. Definición/ejecución de procesos mediante orquestación: BPEL4WS, BPMN 5.4. Definición/ejecución de procesos mediante coreografía: WS-CDL 5.5. Otras especificaciones: XPDL, WSCI/WSCL - WS-Coordination, BPSS, RNIF. Motores/interoperabilidad de procesos: Wf-XML del WfMC, BPSI del OMG
--	---------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 6. Metodología de Desarrollo del Curso.

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## 7. Sugerencias de evaluación.

La evaluación de la materia debe realizarse día a día, basándose en el desempeño de las diferentes actividades:

- Ejercicios realizados durante clase
- Ejercicios realizados de forma independiente
- Reportes de investigaciones (por ejemplo: resúmenes, mapas conceptuales, mapas mentales y conclusiones entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y Software de Apoyo.

- Nicolai M. Josuttis, "SOA in Practice: The Art of Distributed System Design", O'Reilly, 2008
- Eric van der Vlist, "XML Schema" O'Reilly, 2002
- Leonard Richardson, Sam Ruby, "RESTful Web Services" O'Reilly, 2007
- Bill Burke, "RESTful Java with JAX-RS" O'Reilly, 2010

- Subbu Allamaraju, "RESTful Web Services Cookbook", O'Reilly, 2010
- Mark D. Hansen, "SOA Using Java Web Services", Prentice Hall, 2007
- Gopalan Suresh Raj et al., "Implementing SOA with the Java EE 5 SDK", Sun Microsystems, 2006
- Eric Jendrock et al., "The Java EE 6 Tutorial: Basic Concepts (4th Edition)", Sun Microsystems, 2010
- Eric Jendrock et al. "The Java EE 6 Tutorial: Advanced Topics (4th Edition)", Sun Microsystems, 2010
- Gustavo Alonso et al., "Web Services: Concepts, Architectures and Applications", Springer, 2004.

## 9. Actividades propuestas.

Unidad	Actividad
1	<ul style="list-style-type: none"> <li>• Investigar los conceptos básicos de la Arquitectura Orientada a Servicios</li> <li>• Configurar e Instalar el JAX-WS.</li> <li>• Creación de un nuevo proyecto Web basado en SOA en NetBeans.</li> </ul>
2	<ul style="list-style-type: none"> <li>• Investigar la estructura básica de un documento WSDL</li> <li>• Realizar diversas representaciones de clases de objetos en un documento WSDL</li> <li>• Integración de diversas operaciones y servicios en un documento WSDL</li> </ul>
3	<ul style="list-style-type: none"> <li>• Desarrollo de un servicio Web que simule una calculadora</li> <li>• Desarrollo de un servicio Web que simule una agenda de contactos</li> <li>• Desarrollo de un servicio Web con conexión a base de datos</li> </ul>
4	<ul style="list-style-type: none"> <li>• Desarrollo de un servicio Web basado en REST</li> <li>• Invocación de servicios Web en redes sociales que usen APIs basadas en REST</li> </ul>
5	<ul style="list-style-type: none"> <li>• Desarrollar una composición de servicios en BPEL</li> <li>• Desarrollar una coreografía de servicios en WS-CDL</li> </ul>

## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: ARQUITECTURA DE SOFTWARE

Línea de la Asignatura: Ingeniería de Software

Tiempo de dedicación del estudiante a las actividades de:

DOC-TIS-TPS-HRSTOT-CRÉDITOS

(48 – 20 – 100 – 168 - 6)

### 1. Historial de la asignatura

Fecha de revisión / actualización	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo de 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta preliminar Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre de 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba 8 al 12 de julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba 6 al 10 de agosto de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

### 2. Pre-requisitos y co-requisitos

Pre-requisitos		Co-requisitos	
Asignatura	Temas	Asignatura	Temas
Ingeniería de software			

### 3. Objetivo de la asignatura

Aplicar los fundamentos de arquitecturas y estilos arquitectónicos en el desarrollo de sistemas de software.

### 4. Aportación al perfil del graduado

Este curso proporciona al egresado las herramientas necesarias para analizar, diseñar e implementar arquitecturas de software.

## 5. Contenido temático

Unidad	Temas	Subtemas
1	Conceptos básicos.  DOC: 4.  Objetivo: Conocer la idea detrás de la arquitectura de software.	1.1. Arquitectura de la Web 1.2. Arquitectura sobre el escritorio 1.3. Productividad y líneas de productos 1.4. Visualización de turbina
2	Diseño de arquitecturas y conectores.  DOC: 10.  Objetivo: Conocer el proceso de diseño de una arquitectura, estilos, patrones y conectores.	2.1. El proceso de diseño 2.2. Concepción arquitectónica 2.3. Estilos y patrones arquitectónicos 2.4. Fundamentos de conectores 2.5. Roles de conectores 2.6. Tipos de conectores
3	Modelación, análisis e implementación.  DOC: 10.  Objetivo: Conocer el proceso de desarrollo de arquitecturas de software, así como las herramientas disponibles para su modelación.	3.1. Conceptos de modelación 3.2. Vistas y puntos de vista 3.3. Técnicas de modelación específica 3.4. Visualización (UML vs. xADL) 3.5. Alcance del análisis 3.6. Técnicas de análisis 3.7 Frameworks para implementación
4	Estilos arquitectónicos.  DOC: 4.  Objetivo: Profundizar en la importancia y aplicación de los estilos arquitectónicos.	4.1. Arquitecturas distribuidas y en red 4.2. Arquitecturas descentralizadas 4.3. Arquitecturas orientadas a servicios 4.4. Arquitecturas para dominios específicos
5	Diseño para propiedades no funcionales  DOC: 10  Objetivo: Conocer las guías de diseño de arquitecturas de software con calidad.	5.1. Eficiencia 5.2. Complejidad 5.3. Escalabilidad y heterogeneidad 5.4. Adaptabilidad 5.5. Confiabilidad
	Líneas de productos de software	6.1. Ingeniería de software

6	DOC: 10  Objetivo: Conocer el rol de la arquitectura de software específicas del dominio.	específica del dominio 6.2. Arquitectura de software específica del dominio (DSSA) 6.3. DSSA, líneas de productos y estilos arquitectónicos 6.4. Ejemplos de DSSA
---	-------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 6. Metodología de desarrollo del curso

Se sugieren las siguientes actividades como parte del curso:

- Exposición por parte del profesor.
- Análisis de ejemplos: estilos y patrones.
- Modelación de arquitecturas con UML y xADL.
- Prácticas.

## 7. Sugerencias de evaluación

Se recomienda realizar evaluación en cada sesión, con base a las siguientes actividades:

- Comprensión de conceptos.
- Ejercicios extra clase.
- Exposición de temas.
- Exámenes teórico-prácticos.
- Desarrollo de proyectos.

## 8. Bibliografía

- Taylor, Medvidovic, Dashofy; Software Architecture – Foundations, theory and practice; Wiley, 2010.
- Hofmeister, Nord, Soni; Applied Software Architecture; Addison Wesley, 2000.
- Spinellis, Gousios; Beautiful Architecture; O'Reilly, 2009.
- Rozanski, Woods; Software Systems Architecture; Addison-Wesley, 2005.
- Lattanze; Architecting Software Intensive Systems; CRC Press, 2009.
- Clemes, Bachmann, Bass, Garlan, Ivers, Little, Nord, Stafford; Documenting Software Architectures – Views and Beyond; Addison-Wesley, 2002.
- Buschmann; Pattern-Oriented Software Architecture; Vols. 1, 2, 3, 4, 5; Wiley, 2001.
- Gomaa; Designing Software Product Lines with UML; Addison-Wesley, 2004.
- Gorton; Essential Software Architecture; Springer, 2011.
- van der Linden; Software Product Lines in Action; Springer, 2007.

## 9. Actividades propuestas

Unidad	Actividades
--------	-------------



1	<ul style="list-style-type: none"> <li>• Ilustrar con ejemplos de arquitecturas chicas, medianas y grandes, los conceptos básicos.</li> </ul>
2	<ul style="list-style-type: none"> <li>• Explicar cada uno de los patrones y estilos arquitectónicos mediante un ejemplo claramente ilustrativo.</li> <li>• Analizar la relación entre estilos y patrones arquitectónicos.</li> </ul>
3	<ul style="list-style-type: none"> <li>• Mostrar la forma de modelar arquitecturas mediante UML y xADL.</li> <li>• Utilizar software de apoyo; para xADL utilizar ArchStudio.</li> </ul>
4	<ul style="list-style-type: none"> <li>• Ilustrar la importancia de los estilos arquitectónicos con ejemplos.</li> <li>• Visualizar ejemplos en xADL.</li> </ul>
5	<ul style="list-style-type: none"> <li>• Aplicar las guías de diseño en un ejemplo arquitectónico simple.</li> </ul>
6	<ul style="list-style-type: none"> <li>• Explicar la importancia de la LPS y su relación con arquitecturas específicas de un dominio.</li> <li>• Destacar los conceptos de variabilidad, variación y variantes.</li> <li>• Ilustrar la implementación de una LPS sencilla mediante algún lenguaje de componentes, como CaesarJ o ECaesarJ.</li> <li>• Modelar en xADL la LPS sencilla.</li> </ul>

Nombre de la asignatura: **Base de datos**

Línea de trabajo: Ingeniería de Software  
Tecnologías Web

DOCS – TIS – TPS – Horas totales - Créditos

48 – 20 – 100 – 168 - 6

## 1. HISTORIAL DE LA ASIGNATURA

Fecha revisión/actualización	Participantes	Observaciones (cambios o justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preeliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 8 al 12 de Julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis y conformación de la asignatura

## 2. PRE-REQUISITOS Y CORREQUISITOS

Asignatura optativa

Pre-requisitos	
Asignatura	Tema
Ninguna	

Correquisitos	
Asignatura	Tema
Ninguna	

## 3. OBJETIVO DE LA ASIGNATURA

Aplicar los conceptos, teorías y perspectivas que permitan utilizar la tecnología de base de datos en el desarrollo de sistemas de software.

## 4. APORTACIÓN AL PERFIL DEL GRADUADO

Esta asignatura aporta al perfil del graduado elementos para modelar y diseñar correctamente bases de datos que serán utilizadas en diversas aplicaciones, asegurando la integridad, disponibilidad y confidencialidad de la información almacenada.

## 5. CONTENIDO TEMÁTICO POR TEMAS Y SUBTEMAS

UNIDAD	TEMAS	SUBTEMAS
1	Conceptos de Bases de Datos relacionales.  DOC: 6  Objetivo: Revisar los conceptos básicos de las bases de datos relacionales	1.1 Modelos de bases de datos 1.2 Características del modelo relacional 1.3 Restricciones
2	Modelado de Bases de Datos Relacionales  DOC: 8  Objetivo: El alumno modelará bases de datos, usando el modelo relacional	2.1 Modelo E-R 2.2. Modelo E-R extendido 2.3 Transformación del modelo E-R a tablas
3	Normalización Adicional: 1FN, 2FN, 3FN, FNBC.  DOC: 10  Objetivo: El alumno aplicará la teoría de normalización	3.1 Conceptos básicos 3.2 Primera forma normal 3.3. Segunda forma normal 3.4 Tercera forma normal 3.5 Forma normal Boyce- Codd 3.5 Formas normales avanzadas
4	Diseño de Bases de Datos Relacionales.  DOC: 10  Objetivo: El alumno diseñará bases de datos	4.1 Características de SQL 4.2 Lenguaje de definición de datos 4.3 Lenguaje de manipulación de datos 4.4 SQL procedural
5	Administración de Bases de Datos  DOC: 6	5.1 Recuperación  5.2 Control de concurrencia  5.3 Seguridad
6	Modelos diversos de Bases de Datos  DOC: 8	6.1 Bases de datos orientadas a objetos

UNIDAD	TEMAS	SUBTEMAS
		6.2 Tendencias actuales de bases de datos

## 6. Metodología de desarrollo del curso

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Presentaciones de temas por parte de los estudiantes
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## 7. Sugerencias de evaluación

La evaluación de la asignatura debe ser continua y formativa por lo que se hará con base en el desempeño de diversas actividades:

- Solución de casos prácticos solicitados durante las actividades.
- Participación y desempeño en el aula.
- Reportes de prácticas
- Ejercicios realizados en forma independiente
- Reportes de las investigaciones (pueden ser: resúmenes, mapas conceptuales, mapas mentales, conclusiones, entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. BIBLIOGRAFÍA Y SOFTWARE DE APOYO

- Date, C.J., Introducción a los sistemas de base de datos, séptima edición, Addison-Wesley Iberoamericana, S.A.
- Korth, Henry F., Silberschatz A., Fundamentos de bases de datos, Mc Graw Hill
- Elmasri, Armes A., Navathe, Shamkant B., Fundamentos de sistemas de bases de datos Addison Wesley
- Irene Luque R., Miguel A. Gómez N., Enrique López E., Gonzalo Cerruela G., Bases de Datos Desde Chen hasta Codd con Oracle, Alfaomega, 2002
- Groff R. James / Weinberg N. Paul, The complete reference SQL Second Edition, Mc Graw Hill, 2002
- Adoración de Miguel Castaño, Mario Piattini Velthuis, Esperanza Marcos Martínez. Diseño de Bases de datos relacionales, Alfaomega
- Connolly, T. y Begg, C. Database Systems. Addison-Wesley 2003

## 9. PRÁCTICAS PROPUESTAS

Unidad	Actividad
1	<p>Buscar información sobre las características de modelo relacional y sus restricciones.</p> <p>Elaborar un mapa conceptual que muestre las características esenciales del modelo relacional.</p>
2	<p>Revisar las características del modelo Entidad – Relación y Entidad – Relación Extendido</p> <p>Desarrollo de ejercicios aplicando el modelo ER y ERE</p> <p>Revisar las reglas de transformación del modelo ER y ERE a tablas</p> <p>Desarrollo de ejercicios de transformación del modelo ER y ERE a tablas</p>
3	<p>Revisar y analizar los conceptos de la teoría de normalización.</p> <p>Desarrollo de ejercicios aplicando las diferentes formas normales</p>
4	<p>Utilizando SQL, crear bases de datos en un gestor de bases de datos, estableciendo las restricciones de integridad adecuadas.</p> <p>Elaborar las consultas sobre las bases de datos creadas.</p> <p>Elaborar procedimientos almacenados y triggers.</p>
5	<p>Revisar las características sobre recuperación, concurrencia y seguridad que proporciona un gestor de bases de datos.</p>
6	<p>Resolver problemas de modelado de bases de datos orientadas a objetos.</p> <p>Investigar las características generales de otros modelos</p>

#### 10. NOMBRE Y FIRMA DEL CATEDRATICO RESPONSABLE

Nombre

Firma

CELIA ROMERO TORRES

\_\_\_\_\_

## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: Bases de Datos y Bases de Conocimiento  
Línea de trabajo: Ingeniería de Software  
Tiempo de dedicación del estudiante a las actividades de:  
DOC-TIS-TPS-HRSTOT-CREDITOS  
(48 - 20 - 100 - 168 - 6)

### 1. Historial de la asignatura

Fecha de revisión / actualización	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preliminar Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 8 al 12 de Julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis y conformación de la asignatura

### 2. Pre-requisitos y correquisitos.

Es necesario que los alumnos cuenten con conocimientos y habilidades para el modelado y consultas de bases de datos relacionales, proceso de normalización y generación de procesos almacenados.

### 3. Objetivo de la asignatura.

Modelar e implementar aplicaciones relacionadas con bases de conocimiento.

### 4. Aportación al perfil del graduado.

Este curso proporciona al egresado los elementos necesarios para solucionar problemas prácticos relacionados con el análisis, almacenamiento y explotación del conocimiento de las organizaciones, específicamente en lo referente a almacenes de datos, minería de datos y minería web, que le permitirán proponer soluciones a las necesidades de apoyo a la toma de decisiones de cualquier institución o empresa.

## 5. Contenido temático.

Unidad	Tema	Subtemas
1.	Datawarehouse.  DOC: 12  Objetivo: modelar e implementar almacenes de datos (datawarehouse)	1.1 Introducción 1.2 Conceptos sobre data mart, cubo, almacén 1.3 Usos de los almacenes de datos 1.4 Modelo multidimensional 1.5 Implementación de un modelo multidimensional 1.6 OLAP
2.	Introducción a la Minería de Datos.  DOC: 10  Objetivo: conocer los conceptos relacionados con la Minería de Datos	2.1 Introducción 2.2 KDD 2.3 Modelos de Minería de Datos 2.4 Herramientas para desarrollar modelos 2.5 Aplicaciones de Minería de Datos
3	Técnicas de Minería de Datos.  DOC: 12  Objetivo: conocer y utilizar las diferentes técnicas de Minería de Datos	3.1 Regresión Lineal 3.2 Árboles de Decisión 3.3 Modelos Estadísticos 3.4 Redes Neuronales 3.5 Agrupamiento (clustering)
4	Web Mining  DOC: 14  Objetivo: conocer los conceptos relacionados con la minería de datos aplicada a Web	4.1 Conceptos generales 4.2 Minería de Contenido 4.3 Minería de Estructura 4.4 Minería de Uso o Navegación

## 6. Metodología de Desarrollo del Curso.

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## **7. Sugerencias de evaluación.**

La evaluación de la materia debe realizarse día a día, basándose en el desempeño de las diferentes actividades:

- Ejercicios realizados durante clase
- Ejercicios realizados de forma independiente
- Reportes de investigaciones (por ejemplo: resúmenes, mapas conceptuales, mapas mentales y conclusiones entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## **8. Bibliografía y Software de Apoyo.**

1. Data Warehouses and OLAP: Concepts, Architectures and Solutions  
R. Wrembel, C. Koncilia  
IGI Publishing © 2007  
ISBN-10: 1599043653  
ISBN-13: 978-1599043654
2. The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling  
Second Edition  
R. Kimball, M. Ross  
John Wiley & Sons © 2002
3. The Data Warehouse ETL Toolkit  
Second Edition  
R. Kimball  
John Wiley & Sons, 2008  
ISBN 9780470149775
4. Building the Data Warehouse  
Fourth Edition  
W. H. Inmon  
John Wiley & Sons © 2005  
ASIN: B003VSR790
5. Principles of Data Mining  
D. Hand, H. Mannila, P. Smyth  
The MIT Press © 2001  
ISBN:9780262082907
6. Data Mining: Concepts, Models, Methods, and Algorithms  
M. Kantardzic  
John Wiley & Sons © 2011  
ISBN-10: 0470890452  
ISBN-13: 978-0470890455



7. Data Mining and Knowledge Discovery Handbook  
O. Maimon, L. Rokach  
Springer © 2005  
ISBN-10: 0387244352  
ISBN-13: 978-0387244358
8. Data Mining and Statistical Analysis Using SQL  
J. Lovett Jr. R. Trueblood  
Apress © 2001  
ISBN:9781893115545
9. Emerging Technologies of Text Mining: Techniques and Applications  
H. do Prado, E. Ferneda  
IGI Global © 2008  
ISBN:9781599043739
10. Mining the Web: Discovering Knowledge from Hypertext Data  
S. Chakrabarti  
Morgan Kaufmann; 1st edition  
ISBN-10: 9781558607545  
ISBN-13: 978-1558607545
11. Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications)  
B. Liu  
Springer; 2nd printing edition, 2009  
ISBN-10: 3540378812  
ISBN-13: 978-3540378815
12. Mining the Social Web: Analyzing Data from Facebook, Twitter, LinkedIn, and Other Social Media Sites  
Matthew A. Russell  
O'Reilly Media; 1 edition, 2011  
ISBN-10: 9781449388348  
ISBN-13: 978-1449388348.

### **Software**

Oracle SQL Developer Datamodeler <http://www.oracle.com/technetwork/developer-tools/datamodeler/index.html>

PostgreSQL <http://www.postgresql.org>

SQL Server 2005 (con Servicios de Análisis, versión empresarial de prueba)  
<http://www.microsoft.com/spain/technet/sql2005/download.msp>

Weka [http://www.cs.waikato.ac.nz/ml/weka/index\\_downloading.html](http://www.cs.waikato.ac.nz/ml/weka/index_downloading.html)

ht://Miner <http://www.htminer.org/en/>

## 9. Actividades propuestas.

Unidad	Actividad
1	<ul style="list-style-type: none"><li>• Investigar conceptos generales de Datawarehouse</li><li>• Investigar casos de éxito sobre implementaciones Datawarehouse</li><li>• Investigar herramientas para el desarrollo de Datawarehouse</li><li>• Realizar modelado multidimensional</li><li>• Implementar modelo multidimensional</li><li>• Implementar análisis sobre el modelo multidimensional (OLAP)</li><li>• Implementar un caso práctico de Datawarehouse que incluya OLAP</li></ul>
2	<ul style="list-style-type: none"><li>• Investigar conceptos generales de Minería de Datos</li><li>• Investigar metodologías para el desarrollo de aplicaciones de Minería de Datos</li><li>• Investigar casos de éxito sobre aplicaciones de Minería de Datos</li><li>• Plantear aplicación de la metodología KDD para casos de estudio</li><li>• Realizar ejercicios de modelos sobre la(s) herramienta(s) elegida(s)</li><li>• Implementar un modelo para Minería de Datos</li></ul>
3	<ul style="list-style-type: none"><li>• Investigar ventajas y desventajas de las diferentes técnicas de Minería de Datos</li><li>• Analizar adecuaciones a los modelos de Minería de Datos para aplicar las distintas técnicas de Minería de Datos</li><li>• Realizar adecuaciones a los modelos realizados en la unidad anterior para poder aplicar las distintas técnicas de Minería de Datos</li><li>• Aplicar las distintas técnicas de Minería de Datos a los modelos realizados en la unidad anterior</li><li>• Aplicar al menos tres técnicas de Minería de Datos al modelo generado previamente y analizar los resultados</li></ul>
4	<ul style="list-style-type: none"><li>• Investigar usos de la Minería de Datos aplicada a Web</li><li>• Investigar casos de éxito sobre Web Mining</li><li>• Investigar problemática de Web Mining (invasión a la privacidad)</li><li>• Modelar una implementación de Web Mining</li><li>• Plantear los distintos tipos de minería en web para sitios definidos</li></ul>

## 10. Nombre y Firma del Catedrático Responsable.

Beatriz Alejandra Olivares Zepahua.

Nombre de la asignatura: Calidad de Software
Línea de trabajo: Ingeniería de Software
DOCS – TIS – TPS – Horas totales - Créditos 48 – 20 – 100 – 168 - 6

## 1. Historial de la Asignatura

<b>Fecha revisión/actualización</b>	<b>Participantes</b>	<b>Observaciones (cambios o justificación)</b>
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preeliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 8 al 12 de Julio de 2011 Fecha del 9 al 13 de enero de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba  Fecha del 9 al 13 de enero de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

## 2. Pre-requisitos y Co-requisitos

Como pre-requisito se requieren conocimientos de la materia Ingeniería de software

### 3. Objetivo de la asignatura

Conocer técnicas y modelos de Aseguramiento de la Calidad de Software para poder examinar y evaluar el software y los procesos de construcción del mismo en una organización y proponer acciones para su mejora..

### 4. Aportación al perfil del graduado

Este curso proporciona al egresado los conocimientos fundamentales necesarios para examinar, evaluar y mejorar el proceso de construcción de software en la empresa en la cual se desempeñe, de manera que contribuya a mejorar la competitividad de ésta.

De forma específica el curso apoya a:

- Contextualizar los diferentes modelos más utilizados para la mejora de la calidad del software en el ámbito internacional y en México, identificando las características de cada modelo.
- Analizar las métricas del software más comunes.
- Aplicar pruebas del software, mediciones y análisis
- Analizar los aspectos que instrumentan la administración y la evaluación de la calidad del software en las etapas generales del proceso de desarrollo
- Dar soluciones a problemas prácticos con el uso adecuado de tecnologías de cómputo.
- Contribuir en la competitividad de las empresas mediante la aplicación adecuada y efectiva de las tecnologías de computación.
- Formar, desarrollar y evaluar grupos de trabajo de alto rendimiento.
- Proporcionar asesoría técnica a las instituciones para optimizar sus recursos de información.

### 5. Contenido Temático.

UNIDAD	TEMAS	SUBTEMAS
1	Conceptos básicos de calidad y calidad de software  DOC: 4	1.1 Antecedentes de la calidad del software 1.1.1 Productividad y competitividad del software 1.1.2 Problemas en la Industria de Software 1.2 Conceptos básicos de calidad 1.2.1 Definición de calidad y calidad de software 1.2.2 Importancia de la calidad 1.2.3 Quién define la calidad 1.2.4 Gestión, control y aseguramiento de la Calidad de Software

UNIDAD	TEMAS	SUBTEMAS
2	Aseguramiento de la calidad del software (SQA)  DOC: 8	2.1 Relación de la Ingeniería del software con SQA. 2.2 Definición y propósito del SQA. 2.3 Problemas que resuelve la SQA. 2.4 Calidad del software en el ciclo de vida del mismo. 2.5 Roles y responsabilidades de los equipos de desarrollo. 2.6 Habilidades y capacidades del personal de SQA. 2.7 Actividades del SQA. 2.8 Plan SQA 2.9 Métodos y herramientas.
3	Estándares y modelos de procesos aplicados al software.  DOC: 8	3.1 Introducción a los estándares y modelos de calidad. 3.2 ISO 9000, ISO 9001 3.3 CMMI-DEV, PSP, TSP (SED) 3.4 ISO/IEC 15504 Evaluación de procesos de software 3.5 Modelo MoProSoft 3.6 Modelo competisoft
4	Pruebas de software.  DOC: 10	4.1 Conceptos Básicos de Pruebas 4.2 Probando a través del Ciclo de Vida de Desarrollo de Software 4.3 Pruebas de Caja Blanca 4.4 Pruebas de Caja Negra 4.5 Pruebas de usabilidad 4.6 Herramientas de Pruebas 4.7 Administración del Proceso de Pruebas
5	Medición y análisis.  DOC: 8	5.1 Introducción a las métricas 5.2 Modelos y métricas 5.3 Método Meta-Pregunta-Métrica 5.4 Puntos Funcionales 5.5 Relación de las métricas con los estándares y modelos de procesos
6	Evaluación de la calidad del proceso de desarrollo de software.	6.1 Generalidades 6.2 La administración de la calidad del software. 6.3 La documentación. 6.4 La evaluación de la calidad del proceso de desarrollo de software: Aspectos generales. 6.5 Evaluación de la calidad: determinación de los requerimientos del software. 6.6 Evaluación de la calidad: Análisis

UNIDAD	TEMAS	SUBTEMAS
		<p>general y detallado del software</p> <p>6.7 Evaluación de la calidad: Diseño general y detallado del software.</p> <p>6.8 Evaluación de la calidad: Prueba de aceptación del software.</p> <p>6.9 o Evaluación de la calidad: Operación y mantenimiento del software.</p>

## 6. Metodología de desarrollo del curso

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Presentaciones de temas por parte de los estudiantes
- Lecturas comentadas en grupo
- Desarrollo de prácticas
- Desarrollo de proyectos

## 7. Sugerencias de evaluación

La evaluación de la asignatura debe ser continua y formativa por lo que se hará con base en el desempeño de diversas actividades:

- Solución de casos prácticos solicitados durante las actividades.
- Participación y desempeño en el aula.
- Reportes de prácticas
- Ejercicios realizados en forma independiente
- Reportes de las investigaciones (pueden ser: resúmenes, mapas conceptuales, mapas mentales, conclusiones, entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y software de apoyo

Erick Braude y Michael Bernstein, Software Engineering Modern Approaches, John Wiley & Sons, Inc., 2011

Piattini y Garcia. Calidad en el desarrollo y mantenimiento del software. Ed- RA-MA. 2002.

Fenton y Pfleeger. Software Metrics.PWS Publishing Co.2002.

Mario Piattini, et. Competisoft. Mejora de procesos software para pequeñas y medianas empresas y proyectos, RA-MA Editorial

El modelo de procesos para la industria del software, MoProsoft v 1.3, Mayo de 2003

Robert Futrell. Quality Software Project Management. Prentice Hall, 2002

Stephen H. Kan. Metrics and Models in software Quality Engineering. Addison-Wesley, 2002.

Watts S. Humphrey, TSP: Leading a Development Team (The SEI Series in Software Engineering), Addison Wesley, 2005

Watts S. Humphrey, PSP A self- improvement Process for Software Engineers, Addison Wesley, 2010

Mary b. Chrissis, Mike Konrad, Sandy Shrum, "CMMI for Development", Third Edition, 2011

## 9. Actividades propuestas

Unidad	Actividad
1	Investigar y discutir en clase los conceptos relacionados con la calidad del software.
2	<p>Discutir en clase la relación de la calidad con el ciclo de vida del software.</p> <p>Realizar una tabla que especifique los roles, responsabilidades, habilidades y capacidades que debe tener el personal encargado de la calidad del software.</p> <p>Investigar qué herramientas existen para apoyar la calidad del software, elaborar una tabla y analizarla en grupo y aplicar alguna en un caso de estudio.</p>
3	<p>Investigar los diferentes estándares que existen relacionados con la calidad del software, identificando sus características.</p> <p>Realizar una tabla comparativa de los estándares de calidad recalcando sus ventajas y desventajas.</p> <p>Elaborar un plan de calidad para un caso de estudio.</p>
4	<p>Comentar las diferencias entre las pruebas de caja negra, caja blanca y usabilidad.</p> <p>Realizar prácticas elaborando diferentes tipos de pruebas de</p>

	software.  Buscar y analizar diferentes herramientas que apoyen la realización de pruebas al software.
5	Investigar y discutir en grupo las diferentes métricas de calidad del software e identificar su utilización en el proceso de aseguramiento de calidad.
6	Identificar las actividades relacionadas con la administración de la calidad del software en el ciclo de vida del mismo. .

#### **10.Nombre y firma del catedrático responsable**

Ma. Antonieta Abud Figueroa



## TEMARIO DE LA ASIGNATURA

**Nombre de la asignatura:** Computación Paralela y Distribuida

**Línea de trabajo:** Ingeniería de Software

**Tiempo de dedicación del estudiante a las actividades de:**

DOC-TIS-TPS-HRSTOT-CREDITOS

(48 - 20 - 100 - 168 - 6)

### 1. 1. Historial de la Asignatura

<b>Fecha revisión/actualización</b>	<b>Participantes</b>	<b>Observaciones (cambios o justificación)</b>
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preeliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 8 al 12 de Julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis y conformación de la asignatura
Instituto Tecnológico de Orizaba Fecha Marzo 2014	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Revisión de la asignatura

### 2. Pre-requisitos y correquisitos.

Prerrequisitos	
Asignatura	Temas

Correquisitos	
Asignatura	Temas

### **3.- Objetivo de la asignatura**

- Introducir al alumno en los paradigmas de programación de sistemas paralelos y distribuidos: pase de mensajes, memoria compartida y paralelismo de datos. La asignatura es especialmente práctica, y se usarán diferentes sistemas, desde multiprocesadores de memoria compartida y multicores hasta sistemas de memoria distribuida y clusters para el diseño y validación de los programas.
- Introducir al alumno en tópicos relacionados con la evaluación y mejora del rendimiento de programas paralelos y distribuidos.

### **4.- Aportación al perfil del graduado.**

- Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.
- Que los estudiantes sepan aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio
- Que los estudiantes sean capaces de integrar conocimientos y enfrentarse a la complejidad de formular juicios a partir de una información que, siendo incompleta o limitada, incluya reflexiones sobre las responsabilidades sociales y éticas vinculadas a la aplicación de sus conocimientos y juicios
- Que los estudiantes sepan comunicar sus conclusiones y los conocimientos y razones últimas que las sustentan a públicos especializados y no especializados de un modo claro y sin ambigüedades.
- Conocimiento de los conceptos y las técnicas de la Computación Paralela.
- Tener destreza en la utilización de las herramientas que permiten aprovechar al máximo un sistema paralelo
- Aptitud para plantear, modelar y resolver problemas que necesitan de las técnicas de Computación Paralela
- Aptitud para resolver problemas computacionales complejos en Ingeniería utilizando técnicas de Computación de Altas Prestaciones.
- Capacidad para proyectar, calcular y diseñar productos, procesos, instalaciones en el ámbito de la Computación Paralela y Distribuida.
- Capacidad para el modelado matemático, cálculo y simulación en centros tecnológicos y de ingeniería de empresa, particularmente en tareas de investigación, desarrollo e innovación en todos los ámbitos afines a la Computación.
- Elaborar adecuadamente y con cierta originalidad composiciones escritas o argumentos motivados, redactar planes, proyectos de trabajo, artículos científicos y formular hipótesis razonables.

## 5. Contenido temático.

Unidad	Temas
1. Introducción	1.1. Modelos de sistemas paralelos y distribuidos. 1.2. Modelos de programación paralela y distribuida. 1.3. Medida del rendimiento.  (2 horas presenciales de teoría)
2. Programación de los sistemas de memoria distribuida.	2.1. Paradigmas. 2.2. Pase de mensajes. MPI. 2.3. Factores que afectan al rendimiento. Mejora del rendimiento.  (4 horas presenciales de teoría y 20 de prácticas)
3. Principios del Diseño de Algoritmos Paralelos.	3.1 Características de los Algoritmos Paralelos. 3.2 Descomposición, técnicas de descomposición: Recursiva, Datos, Exploratorias, entre otras. 3.3 Granularidad y Concurrencia. 3.4 Tareas, relación entre tareas. 3.5 Consecuencias de la inter-relación de tareas y técnicas para reducir su costo. 3.6 Mapping de tareas. Procesos y Procesadores.  (4 horas presenciales de teoría y 15 de prácticas)
4. Programación de los sistemas de memoria compartida.	4.1. Paradigmas. 4.2. Programación con OpenMP. 4.3. Factores que afectan al rendimiento. Mejora del rendimiento.  (5 horas presenciales de teoría y 22 de prácticas)
5. Paralelismo de datos.	5.1. Programación con HPF. 5.2. Factores que afectan al rendimiento. Mejora del rendimiento.  (2 horas presenciales de teoría)

## 6. Metodología de desarrollo del curso. Se establecen las estrategias y las actividades que sean funcionales y adecuadas para lograr el aprendizaje de los estudiantes.

El material de trabajo para este curso incluye los libros y manuales recomendados. Dada la cantidad de información que se encuentra en la web, se utilizará su acceso como herramienta básica de obtención de información.

Se propone que los alumnos trabajen individualmente en el desarrollo de las prácticas propuestas.

La metodología didáctica se centrará especialmente en el trabajo individual de los alumnos, y se hará un especial esfuerzo para fomentar la discusión con el profesor en clase y el uso de las tutorías individuales.

7. **Sugerencias de evaluación.** Se exponen las estrategias, los procedimientos y las actividades de evaluación que, retomados de la experiencia de los cuerpos académicos, sean adecuados para una evaluación correcta.

La realización de las prácticas es el aspecto más relevante de esta materia, y es un requisito obligatorio para superarla. Se evaluarán de manera continua durante su realización en el laboratorio, y su evaluación supondrá un 75% de la calificación final. El alumno que no supere la evaluación en el laboratorio podrá realizar un examen de prácticas, que será el modo de evaluarlas.

La teoría se evaluará mediante un examen final, que supondrá el 25% de la calificación final.

Se considerará no presentado la no asistencia al examen teórico.

8. **Bibliografía y Software de apoyo.** Se enumera la bibliografía y el software de apoyo recomendado, además de las fuentes de información de distinta índole (hemerográficas, videográficas, electrónicas, etc.).

1. Foster. Designing and Building Parallel Programs, (libro on-line) <http://www-unix.mcs.anl.gov/dbpp/text/node2.html>
2. B. Wilkinson, M. Allen. Parallel Techniques and Applications using Networked Workstations and Parallel Computers, Prentice-Hall, 1999
3. R. Doallo, V.M. Carneiro, B.B. Fraguera, J. Touriño. Multiprocesadores: Estructura y Programación, Ediciones Tórculo, 1995
4. P.S. Pacheco. Parallel Programming with MPI, Morgan Kaufmann Publishers, 1997
5. The Message Passing Interface (MPI) standard. <http://www-unix.mcs.anl.gov/mpi>
6. R. Chandra, L. Dagum, D. Kohr, D. Maydan, J. McDonald, R. Menon. Parallel Programming in OpenMP, Morgan Kaufmann Publishers, 2001
7. OpenMP: Simple, Portable, Scalable SMP Programming. <http://www.openmp.org>
8. C.H. Koelbel, D.B. Loveman, R.S. Schreiber, G.L. Steele, M.E. Zosel. The High Performance Fortran Handbook, The MIT Press, 1994
9. The High Performance Fortran Home Page. <http://www.crpc.rice.edu/HPFF>
10. F. Almeida, D. Giménez, J. M. Mantas, A. M. Vidal Introducción a la programación paralela, Madrid : International Thomson Publishing, 2008
11. A. Grama, Introduction to parallel computing, Harlow : Pearson Educación 2nd. ed., 2003.
12. R. Chandra, L. Dagum, D. Kohr, J. McDonald, R. Menon, Parallel programming in OpenMP, 2001.
13. A. V. Maciá, J. L. Pérez, Introducción a la programación en MPI, Valencia : Universidad Politécnica de Valencia 2000.
14. J. Sanders, E. Kandrot. CUDA by example : an introduction to general-purpose GPU, Addison-Wesley, 2011.
15. D. B. Kirk, W. W. Hwu Programming massively parallel processors : a hands-on approach, Burlington, MA : Morgan Kaufmann, 2010.
16. O. Bonorden, N. Huppelshauser, B. Juurlink, I. Rieping - PUB library, Release 6.0 - User guide and function reference. University of Paderbon, Germany. 1998.
17. O. Bonorden, B. Juurlink, I. von Otte, I. Rieping - The Paderborn University BSP (PUB) Library - Design, Implementation and performance. 13th. International Parallel Processing Symposium 10th. Symposium on Parallel and Distributed Processing (IPPs/SPDP). 1999.
18. R. Chandra, L. Dagum, D. Kohr, D. Mandan, J. McDonald and R. Menon – Parallel Programming in OpenMP – Morgan Kaufmann Publishers. 2001
19. I. Foster - Designing and Building Parallel Programs. Addison-Wesley. 1994.
20. J.A. González, C. León, M.F. Piccoli, M. Printista, J.L. Roda, C. Rodríguez, F. Sande - Performance Prediction of Oblivious BSP Programs - Euro-Par 2001. Parallel Processing. 7th. International Euro-Par Conference. Pp. 96-105. Springer. Manchester, UK, August, 2001.
21. J.A. González, C. León, M.F. Piccoli, M. Printista, J.L. Roda, C. Rodríguez, F. Sande - Oblivious BSP - Expuesto en el 6to. International Euro-Par Conference. Publicado en EURO-PAR 2000, Parallel Processing(Springer). ACM. Pp. 682-685. Munich, Germany. Agosto-Setiembre de 2000.

22. M. Goudreau, K. Lang, S. Rao - Towards Efficiency and Portability: Programming with the BSP model. SPAA'96: 8th Annual ACM SPAA. PP 1-12. 1996.
23. A. Grama, A. Gupta, G. Karypis, and V. Kumar - Introduction to Parallel Computing. Addison-Wesley, 2003.
24. M. Gou and Layang – High Performance Computing : Paradigm and Infrastructure. Wiley. 2006.
25. J. Keller, C. Kesler, J. Larsson - Practical PRAM programming. John Wiley & Sons inc.. 2001.
26. C. Leopold - Parallel and Distributed Computing: A survey of models, paradigms, and approaches. John Wiley & Sons inc.. 2001.
27. W.F. McColl - General purpose parallel computing. Gibbons and Spirakis. Pp 337-391. 1993.
28. W.F. Mccoll - BSP Programming. DICMACS Series in Discrete Mathematics and Theoretical Computer Science. Vol: 0. May, 1994.
29. S. Pelagatti – Structured Development of Parallel Programs – Taylor & Francis -1998.
30. M.F. Piccoli – Computación Paralela Anidada a través de Hipercubos Generalizados. Tesis Doctoral. Universidad Nacional de San Luis. 2005.
31. A.M. Printista – Análisis y Predicción de Modelos de Computación Paralela. Tesis Doctoral. Universidad Nacional de San Luis. 2004.
32. M. Quinn - Parallel Computing. Theory and Practice. Second Edition. McGraw-Hill. Inc. 1994.
33. L.G. Valiant - A Bridging Model for Parallel Computation. Communications of the ACM. Vol. 33, Nº 8. Pp 103-111. 1990.
34. L. Valiant - General Purpose parallel Architectures- Handbook of Theoretical Computer Science. Edit. In J. van Leeuwen. Cap. 18, pp 943-971. 1990.
35. B. Wilkinson, M. Allen - Parallel Programming: Tecniques and Aplications using Networked Workstations and Parallel Computers. Prentice Hall. 1999.

## 9. Actividades propuestas.

Unidad	Actividad
1	• Abierta al catedrático
2	• Abierta al catedrático
3	• Abierta al catedrático
4	• Abierta al catedrático
5	• Abierta al catedrático

## 10.- Nombre y firma del catedrático responsable

---

## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: DISEÑO DE ONTOLOGÍAS

Línea de la Asignatura: Tecnologías Web

Tiempo de dedicación del estudiante a las actividades de:

DOC-TIS-TPS-HRSTOT-CREDITOS

(48 - 20 - 100 - 168 - 6)

### 1. Historial de la asignatura

Fecha de revisión / actualización	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preliminar Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba 8 al 12 de julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba 6 al 10 de agosto de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

### 2. Pre-requisitos y correquisitos.

Prerrequisitos	
Asignatura	Temas

Correquisitos	
Asignatura	Temas

	XML
--	-----

--	--

### 3. Objetivo de la asignatura

Aplicar los conceptos fundamentales de la Ingeniería Ontológica y de la Web Semántica en el desarrollo de aplicaciones abarcando los aspectos prácticos de la selección y aplicación de métodos y técnicas, metodologías, lenguajes y herramientas para la construcción y uso de ontologías.

### 4. Aportación al perfil del graduado.

Este curso proporciona al egresado los elementos necesarios para solucionar problemas prácticos relacionados con el desarrollo de ontologías, permitiéndole construir aplicaciones robustas bajo el enfoque de las nuevas tecnologías

### 5. Contenido temático.

Unidad	Tema	Subtemas
1.	Conceptos Básicos de Ontologías.  DOC: 10  Objetivo: conocer los conceptos requeridos para el desarrollo de ontologías	1.1 Introducción 1.2 Antecedentes 1.3 Metadatos 1.4 Clases, relaciones, funciones, axiomas, instancias 1.5 Tecnologías relacionadas
2	Construcción de Ontologías.  DOC: 10  Objetivo: conocer y aplicar los métodos y herramientas necesarios para la construcción de ontologías	2.1 Methontology 2.2 Herramientas 2.3 Diseño de una ontología
3	Implementación de Ontologías en Lenguajes Ontológicos.  DOC: 12  Objetivo: conocer y aplicar los lenguajes relacionados con la implementación de	3.1 RDF 3.2 RDF-S 3.3 OWL 3.4 DL 3.5 SPARQL

	ontologías	
4	APIs de Ontologías y Razonadores  DOC: 12  Objetivo: conocer las APIs existentes para la construcción, consulta y razonamiento con ontologías	4.1 FaCT ++ 4.2 Pellet 4.3 Jena 4.4 Sesame
5	Introducción a la Web Semántica  DOC: 4  Objetivo: conocer las generalidades de la Web Semántica y su relación con las ontologías	5.1 Conceptos 5.2 Justificación 5.3 Estándares 5.4 Ejemplos

## 6. Metodología de Desarrollo del Curso.

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## 7. Sugerencias de evaluación.

La evaluación de la materia debe realizarse día a día, basándose en el desempeño de las diferentes actividades:

- Ejercicios realizados durante clase
- Ejercicios realizados de forma independiente
- Reportes de investigaciones (por ejemplo: resúmenes, mapas conceptuales, mapas mentales y conclusiones entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y Software de Apoyo.

- Ontology Learning and Knowledge Discovery Using the Web: Challenges and Recent Advances (Premier Reference Source). Wilson Wong. ISBN 978-1609606251
- Learning SPARQL. Bob DuCharme. O'Reilly. ISBN 978-1449306595



- Owl: Representing Information Using the Web Ontology Language. Lee W. Lacy. ISBN 978-1412034487
- Practical RDF. Shelley Powers. O'Reilly. ISBN 978-0596002633
- Programming the Semantic Web. Toby Segaran. O'Reilly. ISBN 978-0596153816
- Semantic Web Programming. John Hebel et al. ISBN 978-0470418017
- JENA. <http://jena.apache.org>
- Protégé. <http://protege.stanford.edu>
- OWL. <http://owl.w3.org/2004/OWL>
- Sesame. <http://www.openrdf.org>
- Twinkle. <http://www.ldodds.com/projects/twinkle>

## 9. Actividades propuestas.

Unidad	Actividad
1	<ul style="list-style-type: none"> <li>• Investigar definiciones de conceptos principales</li> <li>• Identificar clases, relaciones, funciones y axiomas sobre uno o más dominios</li> <li>• Investigar ontologías disponibles en Web</li> </ul>
2	<ul style="list-style-type: none"> <li>• Aplicar Methontology en el diseño de una ontología</li> <li>• Construir una ontología en Protégé</li> </ul>
3	<ul style="list-style-type: none"> <li>• Revisar código de una ontología tanto en RDF como en OWL</li> <li>• Realizar consultas sobre una ontología con DL</li> <li>• Realizar consultas sobre una ontología con SPARQL</li> <li>• Investigar otros lenguajes de consulta</li> </ul>
4	<ul style="list-style-type: none"> <li>• Construir una ontología y realizar consultas sobre ella mediante el uso de APIs</li> <li>• Realizar razonamiento sobre una ontología mediante el uso de APIs</li> </ul>
5	<ul style="list-style-type: none"> <li>• Investigar ejemplos de aplicaciones web semánticas</li> <li>• Diseñar una pequeña aplicación para web semántica</li> </ul>

Nombre de la asignatura: Ingeniería de Software

Línea de trabajo: Ingeniería de Software

Tiempo de dedicación del estudiante a las actividades de:  
DOC-TIS-TPS-HRSTOT-CREDITOS  
(48 - 20 - 100 - 168 - 6)

## 1. Historial de la Asignatura

Fecha revisión/actualización	Participantes	Observaciones (cambios o justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preeliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 8 al 12 de Julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis y conformación de la asignatura

## 2. Pre-requisitos y correquisitos.

Prerrequisitos	
Asignatura	Temas

Correquisitos	
Asignatura	Temas
1. Arquitecturas de software. 2. Ingeniería Orientada al Proceso de Software. 3. Calidad del software. 4. Modelado orientado a objetos	Todos

### 3. Objetivo de la asignatura.

Aplicar técnicas de ingeniería para el desarrollo de productos de software, con el propósito de hacer más eficientes los recursos de cómputo disponibles, así como determinar el esfuerzo dedicado en la producción del software.

### 4. Aportación al perfil del graduado.

Este curso proporciona al egresado los fundamentos teóricos y prácticos sobre la disciplina de Ingeniería de Software, en aspectos relacionados con actividades del ciclo de vida del software, administración de proyectos, calidad de software, para su aplicación en la resolución de problemas de ámbito profesional.

### 5. Contenido temático

Unidad	Tema	Subtemas
1.	Procesos de Software  DOC: 8  <b>Objetivo:</b> Identificar los aspectos fundamentales de la ingeniería de software, así como caracterizar a los diferentes modelos de procesos de desarrollo de software.	1.1 Introducción a la Ingeniería del software. 1.2 Las actividades generales del proceso de desarrollo de software. 1.3 Modelos de procesos - Modelo de cascada - Modelo de espiral - Modelo de desarrollo iterativo e incremental - Desarrollo de prototipos - Proceso Unificado de desarrollo de software - Procesos ágiles - Procesos emergentes
2.	Calidad y Medición de los Sistemas de Software.  DOC: 8  Objetivo: Introducir a los conceptos de calidad y tipos de mediciones en proyectos de desarrollo de software.	2.1 Introducción a la calidad del software. 2.2 El Plan de calidad. 2.3 Inspecciones, Revisiones de aseguramiento de la calidad y auditorías. 2.4 Administración de defectos 2.5 Mejora de procesos y métricas del proceso 2.6 Métricas del producto 2.7 Actividades del proceso de medición
3.	Administración de Proyectos de Software.  DOC 6  Objetivo: Introducir a la	3.1 Introducción a la administración de proyectos de desarrollo de software (organización del proyecto, formación de equipos). 3.2 Herramientas para la administración de

	administración de proyectos de desarrollo de software para obtener los conocimientos y habilidades necesarias para la elaboración de un plan de desarrollo de software.	proyectos de desarrollo de software. 3.3 Administración de riesgos. 3.4 Calendarización y planeación de tareas. 3.5 El plan de administración de proyectos de desarrollo de software.
4.	Modelado de sistemas software  DOC: 6  Objetivo: Obtener el conocimiento para elegir los métodos apropiados de modelado y la descripción del modelo abstracto para ciertas clases de sistemas.	4.1 Introducción al modelado 4.2 Herramientas para modelado de datos y UML. 4.3 Modelado de datos 4.4 Modelado orientado a objetos 4.5 Modelado para aplicaciones Web
5.	Ingeniería de requerimientos  DOC 6  Objetivo: Tener conocimientos sobre la variedad de métodos para la obtención de requerimientos, las herramientas y técnicas.	5.1 Introducción a la ingeniería de requerimientos 5.2 Requerimientos funcionales y no funcionales, de alto nivel y detallados. 5.3 Métodos para la captura de requerimientos. 5.4 Herramientas para la ingeniería de requerimientos 5.5 Trazabilidad en el análisis de requerimientos. 5.6 Tareas de la ingeniería de requerimientos. 5.7 Análisis de requerimientos de alto nivel y detallados. 5.8 El documento de requerimientos
6.	Diseño de software  DOC: 10  Objetivo: Obtener conocimientos para aplicar los principios del diseño orientado a objetos y expresarlos usando UML.	6.1 Introducción al diseño de software 6.2 Arquitectura software (diseño arquitectónico) - Categorización de estilos de arquitectura - Patrones arquitectónicos - Tareas del diseño arquitectónico - Documento de arquitectura software 6.3 Diseño Detallado - Introducción al diseño OO - Diseño en el proceso unificado de desarrollo - Diseño detallado en procesos ágiles - standard IEEE 890 para el diseño detallado
7.	Validación del software	7.1 Introducción a la validación y verificación del software.

	<p>DOC: 4</p> <p>Objetivo: Preparar al alumno para desempeñar las funciones de pruebas del software durante todo el proceso de desarrollo de software y proporcionar herramientas básicas para la gestión de esta función.</p>	<p>7.2 pruebas</p> <p>7.3 El plan de validación del software.</p> <p>7.4 El proceso de verificación y validación del software.</p> <p>7.5 Integración de la verificación y validación al proceso de desarrollo de software.</p> <p>7.6 El papel de los estándares en el proceso de verificación y validación del software.</p>
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**6. Metodología de desarrollo del curso.** Se establecen las estrategias y las actividades que sean funcionales y adecuadas para lograr el aprendizaje de los estudiantes.

Participación activa de los estudiantes a partir de actividades guiadas por el profesor, discusión y presentación de casos y temas fundamentales con base en las lecturas asignadas y exposiciones del profesor, exposición de algunos temas de parte de los alumnos, trabajo continuo de ejercicios y tareas.

Se requiere que los estudiantes realicen un proyecto final de construcción de un software, con base en los conceptos proporcionados por el curso. Para ello se asociarán con compañeros(as) de clase y juntos integrarán un grupo de trabajo (de 3 personas) en la empresa u organización seleccionada para la realización del proyecto. Al final se hará entrega de un reporte debidamente estructurado y se debe realizar una presentación del proyecto ante todo el grupo durante la última semana de actividades.

**7. Sugerencias de evaluación.** Se exponen las estrategias, los procedimientos y las actividades de evaluación que, retomados de la experiencia de los cuerpos académicos, sean adecuados para una evaluación correcta.

- Exámenes teóricos y/o prácticos.
- Exposición
- Solución de casos prácticos solicitados durante las actividades, así como sus conclusiones de forma escrita.
- Ejercicios realizados y reportados.
- Tareas
- Proyecto final

**8. Bibliografía y Software de apoyo.** Se enumera la bibliografía y el software de apoyo recomendado, además de las fuentes de información de distinta índole (hemerográficas, videográficas, electrónicas, etc.).

1. Braude, Software Engineering Modern Approaches, Wiley
2. Bruegge, Dutoit, Ingeniería de software orientado a objetos, Prentice Hall.

3. Sommerville, *Ingeniería de software*, Addison Wesley
4. Braude, *Ingeniería de software, un enfoque orientado a objetos*, Alfaomega
5. Pressman, *Ingeniería de software, un enfoque práctico*. Mc Graw Hill
6. Ghezzi, *Fundamentos de ingeniería de software*, Prentice Hall
7. Jacobson, Booch, Rumbaugh, *El proceso unificado de desarrollo de software*, Addison Wesley.
8. P. Kruchten. The Rational Unified Process. An Introduction.
9. Len Bass, et. al. *Software Architecture in Practice*. Addison Wesley
10. Weitzenfeld, *Ingeniería de software orientada a objetos, teoría y práctica con UML y Java*.
11. Artículos de investigación y de difusión sobre temas de Ingeniería de software de la biblioteca electrónica de la ACM u otra biblioteca a la cual se tenga acceso.

Direcciones electrónicas:

1. [www.sei.cmu.edu](http://www.sei.cmu.edu)
2. [www.upedu.org](http://www.upedu.org)
3. [www.agile-spain.com](http://www.agile-spain.com)
4. [www.extremeprogramming.org](http://www.extremeprogramming.org)
5. [www.computer.org](http://www.computer.org)
6. [www.agilemodeling.com](http://www.agilemodeling.com)
7. [www.enterpriseunifiedprocess.com](http://www.enterpriseunifiedprocess.com)
8. [www.agiledata.org](http://www.agiledata.org)

Software:

1. Herramientas CASE para modelado con UML.
2. Herramienta para especificación de requerimientos
3. Herramienta para administración de proyectos.
4. Herramienta para realización de pruebas de software.

## 9. Actividades propuestas.

Unidad	Actividad
1	<ul style="list-style-type: none"> <li>• Investigar los conceptos básicos de la Ingeniería de Software generando un glosario.</li> <li>• Investigar sobre las actividades generales del proceso de desarrollo de software.</li> <li>• Elaborar un cuadro comparativo de los diferentes modelos de proceso de desarrollo de software</li> <li>• Investigar el modelo de proceso que aplican las diferentes compañías productoras de software en México.</li> <li>• Justificar la elección de alguno de los modelos de proceso para el desarrollo de un proyecto de software.</li> </ul>
2	<ul style="list-style-type: none"> <li>• Investigar aspectos básicos de la calidad del software.</li> <li>• Investigar sobre la estructura de un plan de calidad basado en un modelo o estándar.</li> <li>• Elaborar una taxonomía de métricas del software</li> <li>• Elaborar un plan de calidad en base al proyecto final</li> </ul>

	<ul style="list-style-type: none"> <li>• Discutir en clase el plan de calidad planteado.</li> </ul>
3	<ul style="list-style-type: none"> <li>• Investigar sobre los fundamentos de la administración de proyectos (entre ellos el modelo PMBOK).</li> <li>• Investigar sobre la identificación y tratamiento de riesgos</li> <li>• Ejercicio de tratamiento de riesgos para un caso de estudio.</li> <li>• Realizar ejercicios de elaboración de estructuras de tareas para diferentes tipos de proyectos sobre la(s) herramienta(s) elegida(s)</li> <li>• Desarrollar un plan de administración de proyecto de desarrollo de software.</li> <li>• Discutir en clase sobre los aspectos relativos al plan elaborado.</li> </ul>
4	<ul style="list-style-type: none"> <li>• Investigar sobre lenguajes de modelado de software y extensiones existentes.</li> <li>• Elaborar ejercicios sobre modelado de datos</li> <li>• Elaborar ejercicios sobre modelado orientado a objetos</li> <li>• Elaborar ejercicios sobre modelado de aplicaciones Web</li> <li>• Discutir las ventajas y aspectos referentes al modelado.</li> </ul>
5	<ul style="list-style-type: none"> <li>• Investigar las diferentes tareas que se realizan en la ingeniería de requerimientos para la documentación de proyectos de desarrollo.</li> <li>• Documentar en un caso de desarrollo las distintas tareas de la ingeniería de requerimientos.</li> <li>• Investigar y documentar sobre las distintas técnicas que se implementan dentro de las tareas de la ingeniería de requerimientos.</li> <li>• Desarrollar y aplicar distintas técnicas para cada tarea dentro del caso propuesto a desarrollar.</li> <li>• Elaborar una matriz de trazabilidad de acuerdo al caso de desarrollo propuesto.</li> <li>• Aplicar la(s) herramienta CASE para la documentación de requerimientos del caso de desarrollo propuesto.</li> </ul>
6	<ul style="list-style-type: none"> <li>• Investigar conceptos fundamentales al diseño de software</li> <li>• Analizar y caracterizar a los estilos de arquitectura software y patrones arquitecturales.</li> <li>• Elaborar un cuadro comparativo entre el estilo de arquitectura, el patrón arquitectural, la arquitectura de referencia y la arquitectura software.</li> <li>• Modelar la arquitectura para el caso de desarrollo de la unidad anterior.</li> <li>• Revisar y reportar un análisis de las arquitecturas software de diversas aplicaciones.</li> <li>• Elaborar ejercicios de diseño detallado de acuerdo a RUP y procesos ágiles.</li> </ul>

7	<ul style="list-style-type: none"> <li>• Investigar los aspectos fundamentales de la validación y verificación del software.</li> <li>• Elaborar una taxonomía de tipos de pruebas.</li> <li>• Investigar y analizar ejemplos de planes de validación del software.</li> <li>• Investigar sobre estándares para la validación y verificación del software.</li> <li>• Elaborar ejercicios de pruebas empleando una herramienta(s).</li> <li>• Elaborar el plan de validación de software para el caso de desarrollo propuesto.</li> </ul>
---	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**10. Nombre y firma del catedrático responsable.**

**M.C. Ana María Chávez Trejo**



Nombre de la asignatura: Ingeniería de Software Orientada a Procesos

Línea de trabajo: Ingeniería de Software

DOCS – TIS – TPS – Horas totales - Créditos  
48 – 20 – 100 – 168 - 6

## 1. Historial de la Asignatura

Fecha revisión/actualización	Participantes	Observaciones (cambios o justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preeliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 6 al 10 de Agosto de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis y conformación de la asignatura

## 2. Pre-requisitos y Co-requisitos

Ninguno

## 3. Objetivo de la asignatura

Analizar e implementar procesos de ingeniería de software basados en modelos de calidad para la definición de actividades de administración, desarrollo, mantenimiento y operación de sistemas de software.

## 4. Aportación al perfil del graduado

Esta asignatura aporta al perfil del graduado los fundamentos teóricos y prácticos que le permitan comprender los diferentes enfoques en el proceso de desarrollo de software de forma que pueda integrar las buenas prácticas de ingeniería de software en el desarrollo de productos de software, con lo que podrá proponer soluciones innovadoras y/o ofrecer asesoría técnica a empresas de desarrollo de software.

## 5. Contenido Temático.

Unidad	Temas	SubTemas
1	<p>El proceso: Una visión general DOC: 4</p> <p>Objetivo: Comprender los conceptos y modelos sobre procesos de software.</p>	<p>1.1 Definición de proceso de software</p> <p>1.2 Modelos de procesos de software</p> <p>1.3 Actividades del proceso de software</p> <p>1.3.1 Seguimiento y control</p> <p>1.3.2 Medición y análisis</p> <p>1.3.3 Estimación</p>
2	<p>El Proceso de Gestión de Proyectos Software.</p> <p>DOC: 8</p> <p>Objetivo: Conocer y entender las actividades relacionadas con la gestión de proyectos de software.</p>	<p>2.1 Introducción a la gestión de proyectos software</p> <p>2.2 Gestión de riesgo</p> <p>2.3 Gestión de personal</p> <p>2.4 Trabajo en equipo</p> <p>2.5 Planificación de un proyecto de software</p>
3	<p>El proceso de administración de la configuración del software</p> <p>DOC: 6</p> <p>Objetivo: Comprender los procesos y procedimientos implicados en la administración de cambio de software.</p>	<p>3.1 Gestión de las configuraciones del software</p> <p>3.2 Control de cambios</p> <p>3.3 Control de versiones</p> <p>3.4 Construcción del sistema</p> <p>3.5 Gestión de entregas de software</p>
4	<p>El proceso de desarrollo software</p> <p>DOC: 18</p> <p>Objetivo: Identificar las fases generales del proceso de desarrollo de software y comprender el proceso unificado de desarrollo de software.</p>	<p>4.1 Fases generales del proceso de desarrollo de software</p> <p>4.2 El proceso unificado de desarrollo de software.</p> <p>4.2.1 Introducción</p> <p>4.2.2 Fases</p> <p>4.2.3 Disciplinas y flujos de trabajo</p> <p>4.2.4 Cómo aplicar el proceso unificado de desarrollo de software</p>

<b>Unidad</b>	<b>Temas</b>	<b>SubTemas</b>
5	<p>El proceso ágil de desarrollo de software</p> <p>DOC: 6</p> <p>Objetivo: Conocer el manifiesto ágil y las diferencias entre el desarrollo ágil y los tradicionales, así como conocer las prácticas clave en los enfoques ágiles de desarrollo de software.</p>	<p>5.1 Fundamentos de los Métodos ágiles</p> <p>5.2 Administración de un proyecto ágil</p> <p>5.3 Mejores prácticas en el desarrollo ágil</p> <p>5.4 Programación Extrema</p> <p>5.5 Scrum</p>
6	<p>El proceso de mantenimiento de software y el proceso de operación y soporte de sistemas de software</p> <p>DOC: 6</p> <p>Objetivo: Comprender que los sistemas de software deben mantener su utilidad y aprender los diferentes tipos de mantenimiento y los factores que afectan los costos, así como el uso de sistemas heredados.</p>	<p>6.1 Conceptos básicos</p> <p>6.2 Proceso de mantenimiento del software</p> <p>6.3 Problemas clave en el mantenimiento de software</p> <p>6.4 Técnicas para el mantenimiento de software.</p>

## 6. Metodología de desarrollo del curso

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Presentaciones de temas por parte de los estudiantes
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## 7. Sugerencias de evaluación

La evaluación de la asignatura debe ser continua y formativa por lo que se hará con base en el desempeño de diversas actividades:

- Solución de casos prácticos solicitados durante las actividades.
- Participación y desempeño en el aula.
- Reportes de prácticas
- Ejercicios realizados en forma independiente
- Reportes de las investigaciones (pueden ser: resúmenes, mapas conceptuales, mapas mentales, conclusiones, entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y software de apoyo

1. Sommerville, Ingeniería de software, Addison Wesley
2. Pressman, Ingeniería de software, un enfoque práctico. Mc Graw Hill
3. Braude, Software Engineering Modern Approaches, Wiley
4. Bruegge, Dutoit, Ingeniería de software orientado a objetos, Prentice Hall.
5. Braude, Ingeniería de software, un enfoque orientado a objetos, Alfaomega
6. Ghezzi, Fundamentos de ingeniería de software, Prentice Hall
7. Jacobson, Booch, Rumbaugh, El proceso unificado de desarrollo de software, Addison Wesley.
8. P. Kruchten. The Rational Unified Process. An Introduction.
9. Len Bass, et. al. Software Architecture in Practice. Addison Wesley
10. Artículos de investigación y de difusión sobre temas de Ingeniería de software de la biblioteca electrónica de la ACM u otra biblioteca a la cual se tenga acceso.

## 9. Actividades propuestas

Unidad	Actividad
1	Realizar lecturas sobre el tema de proceso de desarrollo de software.  Comentar en clase las lecturas realizadas.
2	Discutir en grupo sobre las estrategias a seguir durante la gestión de software.  Investigar sobre herramientas de software que apoyen la planificación de un proyecto de software.

	Elaborar la planificación para un caso de estudio, incluyendo actividades, recursos y tiempos.
3	<p>Realizar lecturas y relacionadas con el tema de administración de la configuración de software, y comentarlas en clase.</p> <p>Analizar un producto de software que permita la gestión de versiones.</p> <p>Elaborar una práctica donde se utilice el producto de control de versiones analizado en el punto anterior.</p>
4	<p>Analizar las diferentes fases que comprende el proceso de desarrollo de software y elaborar una tabla que los resuma.</p> <p>Resolver un caso de estudio en donde se aplique el proceso unificado de desarrollo de software.</p>
5	<p>Analizar el manifiesto ágil y determinar las diferencias entre un desarrollo ágil y uno tradicional.</p> <p>Resolver un caso de estudio aplicando un método ágil, ya sea Scrum o XP.</p>
6	<p>Elaborar un ensayo sobre los problemas que se presentan en el mantenimiento de software.</p> <p>Analizar un caso de estudio donde se determine la forma de atender el mantenimiento de software.</p> <p>Analizar un caso de estudio que involucre un sistema heredado.</p>

#### **10.Nombre y firma del catedrático responsable**

Ma. Antonieta Abud Figueroa

Nombre de la asignatura: Interfaz Humano - Computadora

Línea de trabajo: Tecnologías Web

DOCS – TIS – TPS – Horas totales - Créditos  
48 – 20 – 100 – 168 - 6

## 1. Historial de la Asignatura

Fecha revisión/actualización	Participantes	Observaciones (cambios o justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preeliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 8 al 12 de Julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis y conformación de la asignatura

## 2. Pre-requisitos y Co-requisitos

Pre-requisitos	
Asignatura	Temas
Tecnologías de Programación	Todas
Modelado Orientado a Objetos	Todas
Desarrollo Web	Todas

Co-requisitos	
Asignatura	Temas

## 3. Objetivo de la asignatura

Determinar la usabilidad de los sistemas de software (mediante el uso de las métricas apropiadas), para diseñar interfaces de usuario final eficientes.

#### 4. Aportación al perfil del graduado

Esta asignatura aporta al perfil del graduado elementos los conocimientos necesarios para poder diseñar y evaluar los sistemas en función de su usabilidad pensando aun en personas con capacidades diferentes. También un alumno graduado conocerá herramientas que le permitirán integrar nuevas tecnologías en el desarrollo de sistemas.

#### 5. Contenido Temático.

Unidad	Temas	Sub Temas
1	Introducción  DOC: 12  Objetivo: Conocer la evolución de las interfaces humano computadora	1.1 Objetivo del estudio de las interfaces humano-computadora 1.2 El factor humano 1.3 Evolución de las interfaces 1.3.1 La interfaz por línea de órdenes 1.3.2 Menús y formularios 1.3.3 Manipulación directa 1.3.4 Interacción asistida (agentes y asistentes) 1.3.5 El futuro (Computadoras emocionales) 1.4 Uso de las metáforas 1.5 Paradigmas interactivos actuales
2	Diseño de Interfaces  DOC: 16  Objetivo: Conocer e identificar los principios de diseño de interfaces así como su ciclo de desarrollo, en soluciones reales	2.1 Principios de diseño 2.1.1 Diseño de IHC centrado en la Tarea 2.1.2 Diseño de IHC centrado en el Usuario 2.2 Ciclo de vida de una interfaz de usuario 2.3 Análisis de tareas 2.4 Tipos de dispositivos 2.5 Formas de interacción
3	Evaluación de usabilidad (usability) de IHC  DOC: 10  Objetivo: Identificar las	3.1 Accesibilidad 3.2 Internacionalización 3.3 Estándares y guías

Unidad	Temas	Sub Temas
	diferentes métricas para evaluar la usabilidad de una aplicación	
4	<p>Interfaces emergentes</p> <p>DOC: 10</p> <p>Objetivo: Identificar las características y usos de los patrones de diseño estructurales y aplicarlas en casos reales</p>	<p>4.1 Interfaces para la Web</p> <p>4.2 Interfaces de Voz</p> <p>4.3 Interfaces especiales para personas con capacidades diferentes</p>

## 6. Metodología de desarrollo del curso

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Presentaciones de temas por parte de los estudiantes
- Discusiones en grupo
- Resolución de casos de estudio
- Comentar artículos del uso de interfaces humano-computadora en otras universidades
- Desarrollo de aplicaciones
- Prácticas

## 7. Sugerencias de evaluación

La evaluación de la asignatura debe ser continua y formativa por lo que se hará con base en el desempeño de diversas actividades:

- Solución de casos prácticos solicitados durante las actividades.
- Participación y desempeño en el aula.
- Reportes de prácticas
- Ejercicios realizados en forma independiente
- Reportes de las investigaciones (pueden ser: resúmenes, mapas conceptuales, mapas mentales, conclusiones, entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y software de apoyo



1. Jakob Nielsen. **Usability Engineering**. Morgan Kauffman Ed. 1993
2. Harold Thimbleby. **User Interface Design**. ACM Press. 1990
3. Christine Faulkner. **The Essence of Human Computer Interaction**. Prentice Hall.1997
4. John. M. Carroll. **Human Computer Interaction in the New Millenium**. Addison Wesley. 2001
5. Yvonne Rogers, Helen Sharp, Jenny Preece. **Interaction Design**. John Wiley & Sons. 2002
6. Rogers, Sharp, Benyon, Holland and Carey. **Human Computer Interaction**. Addison Wesley. 1994
7. Robert W. Bailey. **Human Performance Engineering: Using Human Factors / Ergonomics to Achieve Computer Usability**. 2nd Edition, Prentice Hall. 1989
8. Ben Shneiderman. **Designing the User Interface**. Addison Wesley. 2009.
9. Dermot Browne. **Structured User Interface Design for Interaction Optimisation**. Prentice Hall. 1993
10. Donald A. Norman. **The Design of Everything Things**. Reissue edition (March 1990)
11. Sullivan and Tyler. **Intelligent User Interfaces**. Addison Wesley. 1993
12. Jakob Nielsen. **Designing Web Usability**. New Riders.1999
13. Deitel, Deitel & Nieto, 2nd. Edition. **Internet & World Wide Web: How to Program**. Prentice Hall. 2007
14. Ed. Brenda Laurel. **The Art of Human Computer Interface Design**. Addison Wesley. 1990

## 9. Actividades propuestas

Unidad	Actividad
1	<p>Desarrollar un trabajo que muestre en una línea de tiempo con fechas, personas que participaron y productos más conocidos, la evolución de las interfaces</p> <p>Desarrollar un programa en Java para el ambiente de MS-DOS, que</p>

	<p>nos permita generar interfaces de texto (botones, etiquetas y cuadro de texto), posteriormente la interfaz puede ser probada simulando su actividad a responder a los eventos.</p> <p>Desarrollar una metáfora por equipo donde se desarrollen los iconos correspondientes a una determinada actividad de trabajo. Cada actividad será diferente por equipo.</p> <p>La metáfora será implementada en un lenguaje de programación visual.</p>
2	<p>Generar un trabajo de investigación donde se compare como funcionaria un mismo programa centrando el diseño de su interfaz en el usuario y el mismo diseño de su interfaz centrado en la tarea</p> <p>Desarrollar un trabajo que muestre en una línea de tiempo con fechas, personas que participaron y productos más conocidos, la evolución de los dispositivos de entrada</p>
3	<p>Se formarán grupos de trabajo. Cada grupo redactará un escenario y una tarea específica que involucre las siguientes consignas:</p> <p>Búsqueda de información o hechos concretos: por ejemplo, ¿cuál es la velocidad de conexión más alta actualmente?</p> <p>Juicio de valores: por ejemplo, ¿cree usted que una conexión de 512 Kbps será suficiente para su sitio?</p> <p>Comparación de hechos o información concreta: por ejemplo, ¿cuál es la conexión con la mejor relación costo/beneficio entre 128, 256 y 512 Kbps?</p> <p>Comparación de Juicio de valores: por ejemplo, ¿por qué es –o no– válida la elección por relación costo/beneficio para un Gerente de Telecomunicaciones conservador?</p> <p>Cada tarea tiene que tener un tiempo máximo y un criterio de éxito (aunque no siempre hay uno sólo, como en el caso de un juicio de valores).</p> <p>Haremos 4 ciclos de 15 minutos en los cuales, 4 personas de cada grupo funcionarán como participantes en los tests de los otros grupos.</p> <p>Cada usuario completará las tareas que pueda hacer en ese tiempo. Cuando un usuario termine la tarea debe avisarlo. En ese momento, o si se pasa el plazo máximo de la tarea, se cuenta el tiempo utilizado.</p> <p>Para cada tarea y a medida que el usuario está avanzando con ella, los observadores deben anotar el tiempo empleado, si la tarea se cumplió con éxito, si hubo errores y de qué tipo, y cualquier otra observación o hecho de valor (exclamaciones o expresiones del usuario, puntos difíciles, etc.).</p> <p>Al terminar, cada grupo:</p> <p>Hará una síntesis de los errores encontrados y sumará su puntaje.</p> <p>Comparará los mitos con las observaciones.</p> <p>Sugerirá cambios que reduzcan los errores de usabilidad</p>

	<p>observados, por orden de gravedad desde el mayor al menor.</p> <p>Tomar un software por equipo y evaluar:</p> <ol style="list-style-type: none"> <li>1. Cercanía entre el sistema y el mundo real</li> <li>2. Visibilidad del estatus del sistema</li> <li>3. Libertad de control por el usuario</li> <li>4. Consistencia y Estándares</li> <li>5. Prevención de Errores</li> <li>6. Se puede reconocer mejor que recordar</li> <li>7. Flexibilidad y eficiencia de uso</li> <li>8. Diseño minimalista y estético</li> <li>9. Ayudas y documentación</li> <li>10. Ayudar a reconocer, diagnosticar y recobrase de errores</li> </ol>
4	<p>Desarrollar un programa que utilice alguno de los asistentes de Microsoft.</p> <p>Este asistente podrá contestar 10 preguntas personales sobre el alumno. La pregunta será introducida en forma de texto.</p> <p>Una pregunta obligatoria que todos los asistentes deben poder responder es:</p> <p>¿Qué preguntas puedo hacerte? y responderá con las 10 preguntas sobre el alumno que tendrá programadas.</p> <p>En mínimo 3 de esas preguntas el asistente deberá mostrarnos una imagen.</p> <p>Para que el programa finalice se le debe de teclear: Nos vemos « nombre », donde « nombre », es un nombre que el alumno le de al asistente.</p> <p>Si alguna pregunta no la reconoce lo debe de indicarlo.</p> <p>Hasta este punto el proyecto vale 85 puntos.</p> <p>Si se le adiciona que en un intervalo de 5 preguntas y se repite una el asistente debe responder, eso ya lo preguntaste y responderla nuevamente.</p> <p>Hasta este punto el proyecto vale 90 puntos.</p> <p>Si el asistente realiza las respuestas en lenguaje hablado a través de las bocinas de la computadora, el proyecto alcanzará los 95 puntos.</p> <p>Si el asistente puede responder mediante el lenguaje hablado el proyecto tendrá un valor de 100 puntos.</p>

10. Nombre y firma del catedrático responsable

Giner Alor Hernández

\_\_\_\_\_

Nombre de la asignatura: Introducción a la Inteligencia Artificial
Línea de trabajo: Inteligencia Artificial
DOCS – TIS – TPS – Horas totales - Créditos 48 – 20 – 100 – 168 - 6

## 1. Historial de la Asignatura

<b>Fecha revisión/actualización</b>	<b>Participantes</b>	<b>Observaciones (cambios o justificación)</b>
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preeliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 8 al 12 de Julio de 2011 Fecha del 9 al 13 de enero de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba  Fecha del 9 al 13 de enero de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

## 2. Pre-requisitos y Co-requisitos

Como pre-requisito se requieren conocimientos de las materias de algoritmos paralelos, estadística y probabilidad, cálculo integral y diferencial,

### 3. Objetivo de la asignatura

Aplicar las técnicas básicas de Inteligencia Artificial requeridas para el desarrollo de Sistemas Inteligentes.

### 4. Aportación al perfil del graduado

Esta asignatura aporta al perfil del egresado de la maestría en Sistemas Computacionales la capacidad de aplicar técnicas de Inteligencia Artificial mediante el desarrollo y programación de modelos matemáticos, estadísticos y de simulación a la solución de problemas complejos de control automático, diagnóstico, toma de decisiones, clasificación, minería de datos, es decir, problemas propios de la Inteligencia Artificial e integrar sus soluciones a situaciones del mundo real.

Asimismo, podrá identificar y abstraer problemas científicos a partir del entorno real, y ser capaces de aplicar metodologías científicas de investigación para resolverlos, así como de comunicar sus resultados en forma efectiva, objetiva y rigurosa.

### 5. Contenido Temático.

UNIDAD	TEMAS	
1	Definiciones, historia, Objetivos y Problemas de la Inteligencia Artificial.  DOC: 4	1.1 Conceptos de IA 1.2 Evolución histórica de la IA 1.3 Actualidad de la IA 1.4 Objetivos de la IA 1.5 Problemas de la IA 1.6 Futuro de la IA
2	Estrategias para el Planteamiento de Problemas de Inteligencia Artificial  DOC: 6	2. Clasificaciones 2.1 Nivel de información 2.1.1 Informadas 2.1.2 No informadas 2.2 Procedimiento de selección de reglas 2.2.1 Irrevocables 2.2.2 Tentativas 2.3 Estrategias retroactivas (Backtracking) 2.4 Estrategias de exploración de grafos

UNIDAD	TEMAS	
3	Métodos de búsqueda de soluciones con o sin adversarios  DOC: 6	3.1. Solución de problemas con búsqueda. 3.2. Espacios de estados. 3.2.1. Determinísticos. 3.2.2. No determinísticos. 3.3. Métodos de búsqueda. 3.3.1. Primero en anchura (breadth-first). 3.3.2. Primero en profundidad (depth-first). 3.3.3. Grafos O. 3.3.4. Grafos A. 3.4. Satisfacción de restricciones. 3.5. Teoría de juegos.
4	Métodos de representación del conocimiento  DOC: 8	4.1 Clasificación del conocimiento 4.2 Representación del conocimiento con métodos simbólicos 4.2.1 Árboles de decisión 4.2.2 Sistemas expertos 4.2.3. Lógica difusa 4.2.4. Sistemas basados en casos 4.2.5 Agentes inteligentes 4.2.6 Métodos basados en los primeros principios basados en el modelo MBM 4.3 Representación del conocimiento con métodos conexionistas 4.3.1 Redes neuronales 4.4 Comparación entre los métodos conexionistas 4.5. Sistemas híbridos Inteligentes 4.6 Formas de integración de técnicas de computación 4.7 Arquitectura híbrida modular a partir del paradigma de agentes
5	Sistemas Expertos y Sistemas Multiagentes  DOC: 8	5.1 Conceptos de Sistema Experto (SE) 5.2 Estructura básica de un SE 5.3 Tipos de SE 5.4 Tareas, ventajas y limitaciones de los Sistemas Expertos 5.5 Evolución del SE

UNIDAD	TEMAS	
		5.6 Conceptos de Sistema Multiagentes 5.7 Características 5.8 Auto-organización y auto dirección 5.9 El estudio de sistemas multiagente 5.10 Frameworks 5.11 Desarrollo orientado a agentes 5.12 Aplicaciones
6	Lógica Difusa y Algoritmos Genéticos  DOC: 8	6.1 Conceptos Básicos. 6.2 Números Difusos 6.3 Operaciones con Conj. Difusos 6.4 Variables Lingüísticas: 6.5 Lógica Difusa 6.6 Sistemas difusos 6.7 Control Difuso: 6.8 Bases de Datos Relacionales Difusas (BDRD): 6.9. División Relacional Difusa. 6.10 Aplicaciones 6.11 Conceptos básicos de Algoritmos genéticos 6.12 Antecedentes y evolución 6.13 Problemática 6.14 Ventajas, desventajas y Limitaciones 6.15 Algoritmo Genético Simple 6.16. Algoritmos Genéticos paralelos 6.17. Algoritmos Genéticos basados en Lógica Borrosa 6.18. Ejemplos de Aplicación
7	Percepción visual, Lenguaje Natural  DOC: 8	7.1 Conceptos básicos de percepción visual. 7.2 Usos 7.3 Control de procesos 7.4 Componentes de un sistema de percepción visual 7.4.1 El sistema de iluminación. 7.4.1.1 Fuentes de luz 7.4.1.2 Esquemas de iluminación 7.4.1.3 La cámara 7.4.2.4. Sensores digitales de imagen 7.4.2.5 objetivos 7.5 Como funciona la visión artificial

UNIDAD	TEMAS	
		7.5.1 Adquisición de imagen 7.5.2 Preprocesado 7.5.3.2 Segmentación 7.5.4. Representación y descripción (extracción de características) 7.5.5 Reconocimiento e interpretación 7.6 Conceptos de lenguaje natural 7.7 Lenguajes de programación 7.8 Procesamiento computacional del lenguaje natural 7.8.1 Estructuras de frase 7.8.2 Pasos de una comunicación 7.9. Arquitectura del LN 7.10 Análisis y comprensión del lenguaje 7.11 Morfología, sintaxis, semántica 7.12 Aplicaciones del LN

## 6. Metodología de desarrollo del curso

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Presentaciones de temas por parte de los estudiantes
- Análisis de casos donde interviene la inteligencia artificial
- Análisis de artículos sobre diferentes temas de la materia
- Desarrollo de investigaciones sobre diferentes temas de la materia y otros temas complementarios
- Desarrollo de proyectos

## 7. Sugerencias de evaluación

La evaluación de la asignatura debe ser continua y formativa por lo que se hará con base en el desempeño de diversas actividades:

- Solución de casos prácticos solicitados durante las actividades.
- Participación y desempeño en el aula.
- Reportes de prácticas
- Ejercicios realizados en forma independiente
- Reportes de las investigaciones
- Exposición de temas y casos resueltos.



- Desarrollo de proyectos que integren los temas vistos en clase, especialmente los de mayor interés para cada alumno.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y software de apoyo

1. WINSTON, Patrick Henry, *Inteligencia Artificial*, Ed. Addison Wesley Iberoamericana, 1992, ISBN 0-201-51876-7
2. GONZALO P.M & Santos P.M, *Inteligencia Artificial e Ingeniería del Conocimiento*, Ed. AlfaOmega, 2006, ISBN 970-75-1166-2
3. MARTÍN DEL BRIO B. & Saenz M. A. *Redes Neuronales y Sistemas Borrosos*, Ed Alfaomega, 2006, ISBN 978-970-15-1250-0
4. PALMA M. J & al, *Inteligencia Artificial Técnicas, Métodos y Aplicaciones*, Ed. McGrawHill, ISBN 978-84-481-5618-3
5. RICH E, & Knight K, *Inteligencia Artificial*, Ed McGrawHill, 1992, ISBN 84-481-1858-8
6. RUSSELL S, & Norvig P, *Inteligencia Artificial, Un enfoque Moderno*, Ed. Prentice Hall, 2006, ISBN 968-880-682-x
7. GIARRATANO J. & Riley G. *Sistemas Expertos, Principios y programación (CLIPS)*, Ed. International Thompson, 3da. Edición, 1996
8. MOCKER R, & Dologite D.G. *Knowledge-Based Systems: An Introduction to expert systems*. MacMillan, 1992.
9. SUPPES H & Hill H, *Introducción a la lógica matemática*, ed. Reverté, 1988.
10. CUENCA J. *Lógica informática*, Ed. Alianza Editorial. S.A, Madrid, 2da Edición 1986.
11. ROWE N.C. *Artificial Intelligence through PROLOG*. Ed. Prentice Hall, 1988. ISBN 0-13-048679-5
12. FERNÁNDEZ G. Universidad Politécnica de Madrid. Escuela Técnica Superior de Ingenieros de Telecomunicación. Departamento de Ingeniería de Sistemas Telemáticos. Grupo de Sistemas Inteligentes.  
<http://www.gsi.dit.upm.es/~gfer/ssii/rcsi/>

13. HOLLAND, J.H., "Adaptation in Natural and Artificial Systems", University of Michigan Press, 1975, 211 p.
16. GOLDBERG, D.E., "Genetic Algorithms in Search, Optimization and Machine Learning", Addison- Wesley Publishing Company, 1989, 412 p
17. Smith, R.E.; Goldberg, D.E., and Earickson, J.A., "SGA-C : A C-language Implementation of a Simple Genetic Algorithm", TCGA Report No. 91002, The Clearinghouse for Genetic Algorithms, The University of Alabama, May 14, 1991.
18. Kaufmann, A., y Faure, R., "Invitación a la Investigación de Operaciones", Segunda Edición, CECSA, México, 1977, 311 p.
- 19 Porter, K., "Handling Huge Arrays", Dr. Dobb's Journal of Software Tools for the Professional Programmer, Vol. 13, No. 3, 1988, pp. 60-3.
- 20 DIAZ, A. y GLOVER, F. (1996): Optimización Heurística y Redes Neuronales en Dirección de Operaciones e Ingeniería. Paraninfo.
- 21 GOLDBERG, D.E. (1989): Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading, MA.
- 22 HOLLAND, J. (1975): Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor.

## 9. Actividades propuestas

Unidad	Actividad
1	Investigar y discutir en clase los conceptos, importancia, aplicaciones, evolución, actualidad y futuro de la inteligencia artificial.
2	Desarrollar la aplicación de una estrategia para el planteamiento de problemas de inteligencia artificial
3	Aplicar por equipo los diferentes métodos de búsqueda de problemas y exposición del mismo
4	Resolver problemas mediante los diferentes métodos de representación del conocimiento. Conocer y usar software sobre los diferentes temas de esta unidad
5	Analizar un sistema experto y un sistema multiagente. Diseñar una aplicación sobre cada uno de ellos
6	Aplicar un algoritmo genético a un problema y resolverlo, asimismo

	mediante algoritmo genético difuso.
7	Realizar una investigación monográfica sobre percepción visual y realizar un proyecto de aplicación Realizar una investigación monográfica sobre lenguaje natural y realizar un proyecto de aplicación

#### **10.Nombre y firma del catedrático responsable**

Hilarión Muñoz Contreras

## TEMARIO DE LA ASIGNATURA

<p>Nombre de la asignatura: LENGUAJES WEB Línea de la Asignatura: Tecnologías Web</p> <p>Tiempo de dedicación del estudiante a las actividades de: DOC-TIS-TPS-HRSTOT-CREDITOS (48 - 20 - 100 - 168 - 6)</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 1. Historial de la asignatura

Fecha de revisión / actualización	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preliminar Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba 8 al 12 de julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba 9 al 13 de enero de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

### 2. Pre-requisitos y correquisitos.

Prerrequisitos	
Asignatura	Temas

Correquisitos	
Asignatura	Temas

	Programación Orientada a Objetos  Normalización  Manejo básico de bases de datos  PHP	Modelado conceptual de aplicaciones web	WAE Ingeniería Web Modelo de Navegación
--	---------------------------------------------------------------------------------------------------------------	--------------------------------------------------	-----------------------------------------------------

### 3. Objetivo de la asignatura

Desarrollar aplicaciones Web mediante el uso de diferentes tecnologías de información emergentes con el fin de resolver problemas reales que involucren la compartición de información en Internet.

### 4. Aportación al perfil del graduado.

Este curso proporciona al egresado los elementos necesarios para solucionar problemas prácticos con el desarrollo de proyectos enfocados a la Web, permitiéndole construir aplicaciones robustas bajo el enfoque de las nuevas tecnologías

### 5. Contenido temático.

Unidad	Tema	Subtemas
1.	Lenguajes del lado del cliente.  DOC: 10  Objetivo: conocer y aplicar lenguajes de programación que se ejecutan en el cliente dentro de una aplicación web	1.1 Introducción 1.2 HTML 1.3 CSS 1.4 JavaScript
2	Lenguajes del lado del servidor.  DOC: 12  Objetivo: conocer y aplicar lenguajes de programación que se ejecutan en el servidor dentro de una aplicación web	2.1 Introducción 2.2 Java Modelo 2 2.3 JSF
3	Tecnologías para procesamiento asíncrono entre cliente y servidor.	3.1 Introducción 3.2 Servicios Web

	DOC: 12  Objetivo: conocer y aplicar las tecnologías web para comunicación asíncrona entre el cliente y el servidor	3.3 Ajax 3.4 Soporte a Ajax con JSF 2.0
4	Lenguajes para RIAs  DOC: 14  Objetivo: conocer y aplicar los conceptos de Aplicaciones Enriquecidas de Internet	4.1 Introducción 4.2 Frameworks para RIAs 4.2.1 Dojo 4.2.2 Mootools 4.2.3 RichFaces

## 6. Metodología de Desarrollo del Curso.

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## 7. Sugerencias de evaluación.

La evaluación de la materia debe realizarse día a día, basándose en el desempeño de las diferentes actividades:

- Ejercicios realizados durante clase
- Ejercicios realizados de forma independiente
- Reportes de investigaciones (por ejemplo: resúmenes, mapas conceptuales, mapas mentales y conclusiones entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y Software de Apoyo.

Shklar L., Rosen R. ( 2009) Web Application Architecture: Principles, Protocols and Practices

Leonard, L. (2010) JSF 2.0 Cookbook

Burns E., Schalk C. (2009) JavaServer Faces 2.0, The Complete Reference

Holdener A. (2008) Ajax: The Definitive Guide

Crane D., Pascarello E., James D. (2005) Ajax in Action

Katz, M. (2011) Practical Rich Faces

Filocamo D. (2009) JBoss RichFaces 3.3

Campeato, O. (2010). Web 2.0 Fundamentals for Developers: With AJAX, Development Tools, and Mobile Platforms. Jones and Bartlett Publishers, Inc. ISBN0763779733

Dojotoolkit <http://www.dojotoolkit.org>

JSF <http://www.oracle.com/technetwork/java/javasee/javaserverfaces-139869.html>

Mootools <http://mootools.net>

NetBeans <http://www.netbeans.org>

RichFaces <http://www.jboss.org/richfaces>

## 9. Actividades propuestas.

Unidad	Actividad
1	<ul style="list-style-type: none"><li>• Desarrollar un sitio Web aplicando hojas de estilo</li><li>• Desarrollar un sitio Web aplicando hojas de estilo y validaciones del lado del cliente</li><li>• Desarrollar un sitio Web aplicando hojas de estilo y modificaciones visuales, a tiempo de ejecución, usando JavaScript</li></ul>
2	<ul style="list-style-type: none"><li>• Desarrollar tres ejemplos de pequeñas aplicaciones Web utilizando Modelo 2</li><li>• Aplicar componentes GUI de JSF dentro de aplicaciones Web</li><li>• Aplicar componentes de validación de JSF dentro de aplicaciones Web</li><li>• Aplicar los elementos previos de JSF y acceso a base de datos dentro de aplicaciones Web</li><li>• Aplicar plantillas con JSF dentro de aplicaciones Web</li></ul>
3	<ul style="list-style-type: none"><li>• Investigar los estándares utilizados en Servicios Web</li><li>• Desarrollar servicios web (Java y PHP)</li><li>• Desarrollar clientes de servicios web (Java y PHP)</li><li>• Investigar Ajax y su soporte desde JSF</li><li>• Desarrollar aplicaciones Web utilizando JSF, servicios web y Ajax</li></ul>
4	<ul style="list-style-type: none"><li>• Investigar conceptos de RIAs</li></ul>

	<ul style="list-style-type: none"><li>• Investigar patrones de diseño gráfico más comunes</li><li>• Desarrollar aplicaciones web utilizando servicios web y Dojo</li><li>• Desarrollar aplicaciones web utilizando servicios web y Mootools</li><li>• Desarrollar aplicaciones web utilizando RichFaces</li></ul>
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



Nombre de la asignatura:	Modelado Conceptual de Aplicaciones Web
Línea de trabajo:	Ingeniería de Software Tecnologías Web
DOCS – TIS – TPS – Horas totales - Créditos	48 – 20 – 100 – 168 - 6

## 1. Historial de la Asignatura

Fecha revisión/actualización	Participantes	Observaciones (cambios o justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 8 al 12 de Julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis y conformación de la asignatura

## 2. Pre-requisitos y Co-requisitos

Como pre-requisito se requieren conocimientos previos de programación orientada a objetos, programación para Web y modelado con UML.

## 3. Objetivo de la asignatura

Construir aplicaciones WEB, utilizando métodos de modelado.

## 4. Aportación al perfil del graduado

Esta asignatura aporta al perfil del graduado elementos para aplicar métodos y técnicas que permitan desarrollar aplicaciones para la Web conforme a metodologías que aseguren la calidad del producto final. La materia permite al egresado participar en el proceso de desarrollo de aplicaciones para la Web así como ofrecer asesoría técnica a instituciones que deseen desarrollar aplicaciones para la Web.

## 5. Contenido Temático.

Unidad	Temas	SubTemas
1	Técnicas de diseño  DOC: 4  Objetivo: Revisar los diferentes enfoques arquitectónicos que se utilizan en el diseño de software.	1.1 Diseño orientado a objetos 1.2 Diseño top-down 1.3 Diseño bottom-up
2	Modelado orientado a objetos  DOC: 8  Objetivo: Aplicar el lenguaje UML en el modelado de aplicaciones para Web	2.1 Lenguaje UML 2.2 Extensiones de UML para modelado de aplicaciones Web (WAE)
3	Meta-modelado  DOC: 8  Objetivo: Identificar las características del meta-modelado y su influencia en las metodologías para desarrollos Web.	3.1 Introducción al Meta-modelado 3.2 Características de un Meta-modelo 3.3 Arquitectura MDA
4	Ingeniería Web  DOC: 10  Objetivo: Identificar y aplicar las fases del ciclo de desarrollo de ingeniería Web	4.1 Atributos de los sistemas y aplicaciones basados en Web 4.2 Mejores prácticas en ingeniería Web 4.3 Formulación de sistemas basados en Web 4.4 Planeación de proyectos de ingeniería Web 4.5 Modelado de análisis para ingeniería Web 4.6 Modelado de diseño para ingeniería Web
5	Modelo de navegación  DOC: 8  Objetivo: Aplicar técnicas de modelado para la navegación en aplicaciones Web.	5.1 Técnicas de modelado para la navegación Web 5.2 Semántica de la navegación 5.3 Sintaxis de la navegación

Unidad	Temas	SubTemas
6	Métodos de producción de ambientes Web DOC : 10  Objetivo: Aplicar metodologías formales para el modelado y desarrollo de aplicaciones Web.	6.1 UWE 6.2 OOHDM

## 6. Metodología de desarrollo del curso

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Presentaciones de temas por parte de los estudiantes
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## 7. Sugerencias de evaluación

La evaluación de la asignatura debe ser continua y formativa por lo que se hará con base en el desempeño de diversas actividades:

- Solución de casos prácticos solicitados durante las actividades.
- Participación y desempeño en el aula.
- Reportes de prácticas
- Ejercicios realizados en forma independiente
- Reportes de las investigaciones (pueden ser: resúmenes, mapas conceptuales, mapas mentales, conclusiones, entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y software de apoyo

### Bibliografía

1. Pressman, R. S. Ingeniería del Software – Un enfoque práctico. 5ta. Edición. McGraw Hill. Madrid, España. 2002.
2. Braude, Ingeniería de software, un enfoque orientado a objetos, Alfaomega
3. WAE (Web Application Extension for UML) J. Conallen, Rational Software, Modelling Web Application with UML (2001)

4. N. Koch. "A Comparative Study of Methods for Hypermedia Development". Technical Report 9905, Ludwig-Maximilians-Universität München, November (1999).
5. Andreas Kraus, Alexander Knapp and Nora Koch â Model-Driven Generation of Web Applications in UWE, 3rd International Workshop on Model-Driven Web Engineering (MDWE 2006), Como, Italy (2006)

#### **Fuentes electrónicas**

1. Agile Modeling Home Page. <http://www.agilemodeling.com/>
2. UWE: UML-based Web Engineering  
<http://uwe.pst.ifi.lmu.de/teachingCourseMaterial.html>

#### **Software de apoyo**

1. Visual Paradigm Versión Comunitaria.
2. SDK Java

#### **1. Actividades propuestas**

<b>Unidad</b>	<b>Actividad</b>
1	<p>Buscar información y ejemplos sobre los enfoques arquitectónicos de diseño.</p> <p>Discutir en grupo las ventajas y desventajas de cada enfoque de diseño.</p> <p>Elaborar un mapa conceptual que muestre las características esenciales de los diferentes enfoques de diseño</p>
2	<p>Repasar el marco conceptual del lenguaje de modelado UML.</p> <p>Revisar los mecanismos de extensión relacionados con el lenguaje de modelado UML.</p> <p>Discutir en grupo los beneficios que los mecanismos de extensión ofrecen al modelado de aplicaciones para Web.</p> <p>Lectura de material sobre extensiones WAE.</p> <p>Elaborar un resumen con los diferentes estereotipos que las extensiones WAE comprenden para las aplicaciones Web.</p> <p>Revisar ejemplos de aplicaciones Web modeladas con las extensiones WAE.</p> <p>Desarrollo de ejercicios donde se apliquen las extensiones WAE en el modelado de aplicaciones Web, incluyendo la vista de usuario y la de programador.</p>

3	<p>Elaborar un mapa conceptual sobre los conceptos relacionados con el meta-modelo y meta-modelado.</p> <p>Análisis de la arquitectura orientada por modelos MDA.</p> <p>Discusión en grupo de la importancia de la arquitectura MDA en el modelado de aplicaciones para la Web.</p> <p>Identificar metodologías basadas en el modelo MDA</p>
4	<p>Lectura de material relacionado con la ingeniería Web.</p> <p>Elaborar una síntesis que incluya los principales aspectos de la ingeniería Web.</p> <p>Investigar los pasos clave para el desarrollo de una aplicación Web. Discutir y formalizar grupalmente lo investigado.</p> <p>Identificar las fases del ciclo de desarrollo de ingeniería Web.</p> <p>Discutir en grupo la importancia de cada una de las fases del ciclo de desarrollo de software.</p> <p>Solución de casos aplicando la ingeniería Web.</p>
5	<p>Análisis de las características de navegación deseables en las aplicaciones Web.</p> <p>Desarrollo de ejercicios que incluyan modelos de navegación para aplicaciones Web.</p> <p>Solucionar casos aplicando modelos de navegación para aplicaciones Web.</p>
6	<p>Investigar las características generales de las metodologías para aplicaciones Web: OOHDM y UWE</p> <p>Realizar una tabla comparativa de las metodologías OOHDM y UWE</p> <p>Resolver casos de estudio de modelado de aplicaciones Web con las metodologías antes mencionadas.</p>

6. Nombre y firma del catedrático responsable

---

Ma. Antonieta Abud Figueroa

Nombre de la asignatura: Modelado Orientado a Objetos

Línea de trabajo: Ingeniería de Software

DOCS – TIS – TPS – Horas totales - Créditos  
48 – 20 – 100 – 168 - 6

### 1. Historial de la Asignatura

Fecha revisión/actualización	Participantes	Observaciones (cambios o justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preeliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba Fecha del 8 al 12 de Julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis y conformación de la asignatura

### 2. Pre-requisitos y Co-requisitos

Como pre-requisito se requieren conocimientos previos de programación orientada a objetos y modelado con UML.

Como co-requisito se debe cursar al mismo tiempo que la materia Tecnologías de Programación.

### 3. Objetivo de la asignatura

Aplicar la práctica de Modelado de Arquitecturas Orientadas a Objetos con UML, así como los Principios de Diseño para el desarrollo eficiente de productos de software de alta calidad.

### 4. Aportación al perfil del graduado

Esta asignatura aporta al perfil del graduado elementos para solucionar problemas prácticos con el uso adecuado de tecnologías de cómputo, específicamente con la aplicación de los patrones de diseño, con lo cual podrá proponer soluciones

innovadoras en el desarrollo de software. De igual forma tendrá elementos para ofrecer asesoría técnica a instituciones que deseen optimizar sus recursos.

## 5. Contenido Temático.

Unidad	Temas	SubTemas
1	Lenguaje de modelado orientado a objetos DOC: 8  Objetivo: Revisar los elementos del modelado UML necesarios para el uso de patrones de diseño	1.1 Lenguaje UML 1.2 Clases y Clases abstractas 1.3 Herencia y Polimorfismo 1.4 Interfaces 1.5 Diagrama de clases 1.6 Diagrama de secuencia 1.7 Diagrama de colaboración 1.8 Diagrama de estados 1.9 Diagrama de actividades
2	Principios de Diseño  DOC: 8  Objetivo: Entender los principios fundamentales del diseño orientado a objetos y su relación con los patrones de diseño.	2.1 Concepto de diseño 2.2 Tipos de diseño 2.3 Principios básicos de diseño 2.4 Diseño y patrones
3	Patrones de diseño de instanciación  DOC: 8 Objetivo: Identificar las características y usos de los patrones de diseño de instanciación y aplicarlas en casos reales.	3.1 Patrón Abstract Factory 3.2 Patrón Builder 3.3 Patrón Factory Method 3.4 Patrón Prototype 3.5 Patrón Singleton
4	Patrones de diseño estructurales DOC: 10  Objetivo: Identificar las características y usos de los patrones de diseño estructurales y aplicarlas en casos reales	4.1 Patrón Adapter 4.2 Patrón Bridge 4.3 Patrón Composite 4.4 Patrón Decorator 4.5 Patrón Facade 4.6 Patrón Flyweight 4.7 Patrón Proxy
5	Patrones de diseño de comportamiento  DOC: 10 Objetivo: Identificar las características y usos de los	5.1 Patrón Chain of Responsibility 5.2 Patrón Command 5.3 Patrón Interpreter 5.4 Patrón Iterator 5.5 Patrón Mediator

Unidad	Temas	SubTemas
	patrones de diseño de comportamiento y aplicarlas en casos reales	5.6 Patrón Memento 5.7 Patrón Observer 5.8 Patrón State 5.9 Patrón Strategy 5.10 Patrón Template Method 5.11 Patrón Visitor
6	Calidad y Medición de Diseños Orientados a Objetos  DOC: 4  Objetivo: Aplicar métricas para la evaluación de la calidad de diseños orientados a objetos.	6.1 Métricas para el diseño OO 6.2 Métricas orientadas a clases 6.3 Métricas orientadas a objetos

## 6. Metodología de desarrollo del curso

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Presentaciones de temas por parte de los estudiantes
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## 7. Sugerencias de evaluación

La evaluación de la asignatura debe ser continua y formativa por lo que se hará con base en el desempeño de diversas actividades:

- Solución de casos prácticos solicitados durante las actividades.
- Participación y desempeño en el aula.
- Reportes de prácticas
- Ejercicios realizados en forma independiente
- Reportes de las investigaciones (pueden ser: resúmenes, mapas conceptuales, mapas mentales, conclusiones, entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y software de apoyo

### Bibliografía



1. Gamma Erich, Richard Helm, Ralph Johnson, John Vlissides, **Patrones de diseño. Elementos de software orientado a objetos reutilizable**, PRENTICE HALL/PEARSON, 2009
2. Alan Shalloway and James R. Trott, **Design Patterns Explained: A New Perspective on Object-Oriented Design (2nd Edition)**, Addison-Wesley Professional
3. Steven John Metsker and William C. Wake, **Design Patterns in Java(TM)** (Software Patterns Series), Addison-Wesley Professional ©2006
4. Steven John Metske, **Design Patterns Java™ Workbook**, Publisher: Addison Wesley
5. March 25, 2002
6. Stephen H. Kan, **Metrics and Models in Software Quality Engineering** (2nd Edition), Publisher: Addison-Wesley Professional, Pub. Date: September 16, 2002

### Software de apoyo

1. JDK 1.6 o posterior.
2. Eclipse IDE.

### 7. Actividades propuestas

Unidad	Actividad
1	<p>Elaborar modelos de clases que involucren clases abstractas, herencia y polimorfismo.</p> <p>Resolver casos</p> <p>Repasar los conceptos relacionados con el lenguaje de modelado UML.</p> <p>Explorar las estrategias del uso de herencia y polimorfismo.</p>
2	<p>Discutir en grupo la lectura relativa al diseño de software orientado a objetos</p> <p>Discutir en grupo los beneficios que el uso de patrones ofrece en el diseño de software.</p> <p>Lectura de material sobre diseño orientado a objetos.</p> <p>Realizar una lista de los problemas de diseño indicando cómo se solucionan a través del uso de patrones.</p>
3	<p>Análisis de ejemplos donde se apliquen patrones de diseño de instanciación.</p> <p>Desarrollo de ejercicios utilizando los patrones de diseño de instanciación</p> <p>Exposición ante el grupo soluciones a problemas que aplican patrones de</p>

	<p>instanciación.</p> <p>Revisar material sobre los patrones de instanciación.</p> <p>Elaborar una tabla que resuma los patrones de diseño de instanciación incluyendo la intención, criterios de uso, estructura y participantes.</p> <p>Solución de casos aplicando patrones de instanciación.</p>
4	<p>Análisis de ejemplos donde se apliquen patrones de diseño estructurales.</p> <p>Desarrollo de ejercicios utilizando los patrones de diseño estructurales.</p> <p>Exposición ante el grupo de soluciones a los casos que comprenden patrones estructurales.</p> <p>Revisar material sobre los patrones de diseño estructurales.</p> <p>Elaborar una tabla que resuma los patrones estructurales que incluya la intención, criterios de uso, estructura y participantes.</p> <p>Solución de casos aplicando patrones estructurales.</p>
5	<p>Análisis de ejemplos donde se apliquen patrones de diseño de comportamiento.</p> <p>Desarrollo de ejercicios utilizando los patrones de diseño de comportamiento.</p> <p>Exposición de casos resueltos con patrones de comportamiento.</p> <p>Revisar material sobre patrones de comportamiento.</p> <p>Elaborar una tabla que incluya la intención, criterios de uso, estructura y participantes en los patrones de diseño de comportamiento.</p> <p>Solucionar casos aplicando patrones de comportamiento.</p>
6	<p>Presentación de técnicas de medición de calidad de software orientado a objetos.</p> <p>Discutir las diferencias entre las métricas orientadas a clases y a objetos.</p> <p>Revisar material relativo a las métricas del software orientado a objetos</p> <p>Elaborar ejercicios de evaluación de software orientado a objetos.</p>

9. Nombre y firma del catedrático responsable

---

Ma. Antonieta Abud Figueroa

## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: PROGRAMACION PARA DISPOSITIVOS MOVILES

Línea de la Asignatura: Tecnologías Web

Tiempo de dedicación del estudiante a las actividades de:

DOC-TIS-TPS-HRSTOT-CREDITOS

(48 - 20 - 100 - 168 - 6)

### 1. Historial de la asignatura

Fecha de revisión / actualización	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preliminar Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba 8 al 12 de julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba 9 al 13 de enero de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

### 2. Pre-requisitos y correquisitos.

Prerrequisitos	
Asignatura	Temas

Correquisitos	
Asignatura	Temas

	Programación Orientada a Objetos  Manejo básico de bases de datos  JAVA
--	-------------------------------------------------------------------------------------------

Ninguno	Ninguno
---------	---------

### 3. Objetivo de la asignatura

Desarrollar aplicaciones con tecnologías de cómputo móvil e inalámbrico basadas en entornos de programación móvil J2ME.

### 4. Aportación al perfil del graduado.

Este curso proporciona al egresado los elementos necesarios para solucionar problemas prácticos con el desarrollo de aplicaciones para dispositivos móviles, permitiéndole construir aplicaciones robustas bajo el enfoque de tecnologías emergentes

### 5. Contenido temático.

Unidad	Tema	Subtemas
1.	Arquitectura J2ME.  DOC: 3  Objetivo: Conocer la arquitectura básica de las aplicaciones móviles basadas en J2ME	1.1. Introducción 1.1.1 Evolución de Java 1.2. Dispositivos Móviles 1.2.1. Características 1.2.2. Limitaciones 1.3. Tipos de Clientes para Dispositivos móviles 1.3.1. Clientes Thin 1.3.2. Clientes Thick 1.3.3. Comparación entre clientes thin y thick 1.3.4. Tecnologías para clientes thin 1.3.5. Tecnologías para clientes thick 1.4. J2ME 1.4.1. Características 1.4.2. Capas de J2ME

		1.4.3. Otras bibliotecas de funciones
2	<p>Aplicaciones MIDP.</p> <p>DOC: 10</p> <p>Objetivo: Diseñar e implementar la estructura básica de aplicaciones móviles basadas en J2ME</p>	<p>2.1. Arquitectura CLDC/MIDP</p> <p>2.2. Arquitectura de software para dispositivos MID</p> <p>2.3. CLDC (Connected, Limited Device Configuration)</p> <p>2.3.1. CLDC: java.lang</p> <p>2.3.2. CLDC: java.util</p> <p>2.3.3. CLDC: javax.microedition.io</p> <p>2.4. MIDP (Mobile Information Device Profile)</p> <p>2.4.1. MIDP: midlet</p> <p>2.4.2. MIDP: lcdui</p> <p>2.4.3. MIDP: rms</p> <p>2.5. Ciclo de vida de una aplicación MIDP</p> <p>2.5.1. startApp()</p> <p>2.5.2. pauseApp()</p> <p>2.5.3. destroyApp()</p>
3	<p>Creación de Interfaces de Usuario.</p> <p>DOC: 12</p> <p>Objetivo: Planear, diseñar e implementar interfaces de usuario en J2ME.</p>	<p>3.1 Interfaz MIDP de alto nivel</p> <p>3.2. Componentes de alto nivel</p> <p>3.2.1. Display</p> <p>3.2.2. Displayable</p> <p>3.2.3. Command</p> <p>3.2.4. CommandListener</p> <p>3.2.5 Screen</p> <p>3.2.6. TextBox</p> <p>3.2.7. List</p> <p>3.2.8. Alert</p> <p>3.2.9. Form</p> <p>3.2.10. Item</p> <p>3.2.11. DateField</p> <p>3.2.12. TextField</p> <p>3.2.13. Gauge</p> <p>3.2.14. ChoiceGroup</p> <p>3.3. Comandos de usuario</p> <p>3.3.1. OK</p> <p>3.3.2. CANCEL</p> <p>3.3.3. BACK</p> <p>3.3.4. STOP</p> <p>3.3.5. HELP</p>

		3.4. Diseño de pantallas
4	Administración de Datos en Dispositivos Móviles  DOC: 10  Objetivo: conocer la administración de datos en dispositivos móviles	4.1. Acceso a Bases de Datos 4.2. Manejo de archivos 4.3. Estructura de archivos y carpetas
5	Redes de Datos, Servicios y Seguridad  DOC: 20  Objetivo: conocer las diferentes prestaciones de servicios ofrecidos en los dispositivos móviles	5.1. Llamados a procedimientos Remotos 5.2. Integración con servicios Web 5.3. Cámara 5.4. Geo-localización 5.5. Seguridad
6	Otras Tecnologías para Móviles  DOC: 12  Objetivo: Planear, diseñar e implementar aplicaciones móviles basadas en el sistema operativo Android	6.1. Introducción a la plataforma Android 6.2. Instalación del SDK 6.3. Componentes de un proyecto Android 6.4. Codificación de programas 6.4.1. Configurar un emulador 6.4.2. Manipulación de actividades 6.4.3. Uso de menú 6.4.4. Multimedia

## 6. Metodología de Desarrollo del Curso.

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## 7. Sugerencias de evaluación.

La evaluación de la materia debe realizarse día a día, basándose en el desempeño de las diferentes actividades:

- Ejercicios realizados durante clase
- Ejercicios realizados de forma independiente
- Reportes de investigaciones (por ejemplo: resúmenes, mapas conceptuales, mapas mentales y conclusiones entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y Software de Apoyo.

- Kim Topley. J2ME in a Nutshell. O'Reilly Media.
- James P. White and David A. Hemphill. Java 2 Micro Edition. Manning Publications.
- James Edward Keogh. J2ME: The Complete Reference. McGraw-Hill/Osborne Media
- Eric van der Vlist, Danny Ayers, Erik Bruchez, Joe Fawcett, Alessandro Vernet. Mobile Web 2.0: The Innovator's Guide to Developing and Marketing Next Generation Wireless/Mobile Applications. Futuretext.
- Zigurd Mednieks, Laird Dornin, G. Blake Meike and Masumi Nakamura. Programming Android. O'Reilly Media; 1 edition
- Satya Komatineni, Dave MacLean and Sayed Hashimi. Pro Android 3. Apress; 1 edition
- Marko Gargenta. Learning Android. O'Reilly Media; 1 edition

## 9. Actividades propuestas.

Unidad	Actividad
1	<ul style="list-style-type: none"> <li>• Investigar la sintaxis de J2ME.</li> <li>• Configurar e Instalar el Java Wireless Toolkit.</li> <li>• Creación de un nuevo proyecto J2ME en NetBeans.</li> <li>•</li> </ul>
2	<ul style="list-style-type: none"> <li>• Investigar la estructura básica de las aplicaciones móviles basadas en J2ME</li> <li>• Investigar el ciclo de vida de las aplicaciones basadas en J2ME</li> <li>• Implementar ejemplos de aplicaciones basadas en J2ME</li> <li>• Compilar y probar la aplicación (MIDlet) en el emulador.</li> <li>• Creación de archivos JAR y JAD a línea de comando</li> </ul>
3	<ul style="list-style-type: none"> <li>• Diseño de formularios en J2ME</li> </ul>

	<ul style="list-style-type: none"> <li>• Implementar validaciones de formularios en J2ME</li> <li>• Creaciones de Listas</li> <li>• Creación de Menus Contextuales</li> <li>• Uso de elementos TextBox, Choice y Form Items</li> </ul>
4	<ul style="list-style-type: none"> <li>• Desarrollar una aplicación móvil en J2ME que contenga el curriculum vitae del estudiante</li> <li>• Desarrollar una aplicación móvil en J2ME para una galería de fotos</li> </ul>
5	<ul style="list-style-type: none"> <li>• Desarrollar una aplicación móvil en J2ME para una agenda de contactos usando una conexión a bases de datos</li> <li>• Desarrollar una aplicación móvil en J2ME para la reproducción de video y audio</li> <li>• Desarrollar una aplicación móvil en J2ME para un videojuego sencillo</li> </ul>
6	<ul style="list-style-type: none"> <li>• Instalación de MotoDev Studio</li> <li>• Ejecución de un programa</li> <li>• Manejo de varias pantallas (Activity)</li> <li>• Uso de Menú y multimedia.</li> <li>• Desarrollo de una agenda de contactos</li> </ul>



## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: PROGRAMACIÓN ORIENTADA A ASPECTOS

Línea de la Asignatura: Ingeniería de Software

Tiempo de dedicación del estudiante a las actividades de:

DOC-TIS-TPS-HRSTOT-CREDITOS

(48 - 20 - 100 - 168 - 6)

### 1. Historial de la asignatura

<b>Fecha de revisión / actualización</b>	<b>Participantes</b>	<b>Observaciones (cambios y justificación)</b>
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba 8 al 12 de julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba 9 al 13 de enero de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

### Pre-requisitos y co-requisitos.

Prerrequisitos	
Asignatura	Temas
Tecnologías de programación	Polimorfismo
	Clases internas
	Genéricos

Correquisitos	
Asignatura	Temas

### 2. Objetivo de la asignatura

Aplicar técnicas de programación orientadas a aspectos para el desarrollo de productos de software enfocados a la reutilización de insumos funcionales y no funcionales.

### 3. Aportación al perfil del graduado.

Esta asignatura proporciona al egresado los elementos necesarios para solucionar problemas prácticos con la evolución de sistemas, con especial énfasis en la parte no funcional.

### 4. Contenido temático.

Unidad	Temas	Subtemas
1	Fundamentos de la separación de asuntos.  DOC: 8.	1.1. Problemas con el modelo de objetos. 1.2. Modelos que extienden al modelo de objetos.
2	Modelos de puntos de unión.  DOC: 8.	2.1. Modelo de flujo de datos del programa. 2.2. Modelo de invocaciones a tiempo de ejecución.
3	Implementación de aspectos.  DOC: 12.	3.1. Lenguaje AspectJ: corte estático y corte dinámico. 3.2. Lenguaje CaesarJ: activación dinámica de aspectos. 3.3. Lenguaje ECaesarJ: máquinas de estado.

		3.4. Lenguaje PtolemyJ: contratos traslúcidos.
4	Polimorfismo aspectual.  DOC: 8.	4.1. Familias de tipos. 4.2. Clases virtuales. 4.3. Refinamiento de familias. 4.4. Upcasting/downcasting familiar
5	Reutilización de insumos sin herencia ni composición.  DOC: 12	5.1. Wrappers. 5.2. Mixins. 5.3. Bindings.

## 6. Metodología de desarrollo del curso.

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor.
- Discusiones en grupo.
- Resolución de casos de estudio.
- Prácticas.

## 7. Sugerencias de evaluación.

La evaluación de la materia se realizará en cada clase, con base en el desempeño de las diferentes actividades:

- Ejercicios realizados durante clase.
- Ejercicios realizados de forma independiente.
- Reportes de investigaciones (por ejemplo: resúmenes, mapas conceptuales, mapas mentales y conclusiones entre otros).
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y software de apoyo.

- Aksit, M. Separation and Composition of Concerns in the Object-Oriented Model. ACM Comput. Surv. 28 (December 1996).
- Aksit, M., Wakita, K., Bosch, J., Bergmans, L., and Yonezawa, A. Abstracting Object Interactions Using Composition Filters. In Object-based Distributed Processing (1993), Springer-Verlag, pp. 152-184.
- Avgustinov, P., Christensen, A. S., Hendren, L. J., Kuzins, S., Lhoták, J., Lhoták, O., de Moor, O., Sereni, D., Sittampalam, G., and Tibble, J. abc: An Extensible

AspectJ Compiler. Transactions on Aspect-Oriented Software Development 1 3880 (October 2005), 293-334.

- Baniassad, E., and Clarke, S. Theme: An Approach for Aspect-Oriented Analysis and Design. In International Conference on Software Engineering (ICSE) (Washington, DC, USA, 2004), IEEE Computer Society Press.
- Bergmans, L., and Aksit, M. The Composition-Filters Object Model. Tech. rep., TRESE group, Department of Computer Science, University of Twente, Enschede, The Netherlands, 1994.
- Bergmans, L., and Aksit, M. Composing Multiple Concerns Using Composition Filters. Tech. rep., TRESE group, Department of Computer Science, University of Twente, Enschede, The Netherlands, 2001.
- Bergmans, L. M. J. Composing Concurrent Objects. PhD thesis, University of Twente, Enschede, The Netherlands, 1994.
- Clarke, S., and Baniassad, E. Aspect-Oriented Analysis and Design - The Theme Approach. Addison-Wesley, March 2005.
- Colyer, A., Rashid, A., and Blair, G. On the Separation of Concerns in Program Families. Tech. Rep. COMP-001-2004, Computing Department, Lancaster University, 2004.
- Elrad, T., Fillman, R. E., and Bader, A. Aspect-Oriented Programming. Communications of ACM, October 2001.
- Filman, R. E., Elrad, T., Clarke, S., and Aksit, M. Aspect-Oriented Software Development. Addison-Wesley, Boston, 2005.
- Filman, R. E., and Friedman, D. P. Aspect-Oriented Programming is Quantification and Obliviousness. Tech. Rep. 01.12, RIACS, May 2001. Presented at Workshop on Advanced Separation of Concerns, OOPSLA October 2000.
- Foundation, T. E. AspectJ. <http://www.eclipse.org/aspectj/>.
- Harrison, W., and Ossher, H. Subject-Oriented Programming (A Critique of Pure Objects). ACM SIGPLAN Notices 28, 10 (1993), 411-428.
- Hilsdale, E., and Hugunin, J. AdviceWeaving in AspectJ. In AOSD'04: Proceedings of the 3rd International Conference on Aspect-Oriented Software Development (2004), pp. 26-35.
- Jacobson, I. Language Support for Changeable Large Real Time Systems. In OOPLSA'86: Conference proceedings on Object-oriented programming systems, languages and applications (New York, USA, 1986), ACM, pp. 377-384.
- Jacobson, I., and Ng, P.-W. Aspect-Oriented Software Development with Use Cases. Addison Wesley, 2005.
- Kiczales, G., Hilsdale, E., Hugunin, J., Kersten, M., Palm, J., and Griswold, W. G. An Overview of AspectJ. In European Conference on Object-Oriented Programming 2001, Lecture Notes in Computer Science 2072 (June 2001), Springer-Verlag, pp. 327-353.
- Kiczales, G., Hilsdale, E., Hugunin, J., Kersten, M., Palm, J., and Griswold, W. G. Getting Started with AspectJ. Communications of the ACM 44, 10 (October 2001), 59-65.

- Kiczales, G., Irwin, J., Lamping, J., Loingtier, J.-M., Lopes, C. V., Maeda, C., and Mendhekar, A. Aspect-Oriented Programming. ACM Comput. Surv. 28 (1996).
- Laddad, R. AspectJ in Action: Practical Aspect-Oriented Programming. Manning Publications, July 2003.
- Lieberherr, K. Adaptive Object-Oriented Software: The Demeter Method with Propagation Patterns, 1st ed. Pws Pub Co, August 1995.
- Lieberherr, K. Demeter and aspect-oriented programming. <http://www.ccs.neu.edu/home/lieber/connection-to-aop.html>, 1998.
- Mezini, M. Variation-Oriented Programming: Beyond Classes and Inheritance. PhD thesis, Fachbereich Elektrotechnik und Informatik, Siegen, Germany, 1997.
- Mezini, M. Towards Variational Object-Oriented Programming: The Rondo Model. Tech. rep., Darmstadt University of Technology, 1998.
- Mezini, M., and Ostermann, K. Conquering Aspects with Caesar. In AOSD'03: Proceedings of the 2nd International Conference on Aspect-Oriented Software Development (Boston, MA, USA, March 2003), ACM, pp. 90-100.
- Ossher, H., and Tarr, P. Multi-Dimensional Separation of Concerns in Hyperspace. Tech. rep., IBM T. J. Watson Research Center, 1999.
- Ossher, H., and Tarr, P. Multi-Dimensional Separation of Concerns and the Hyperspace Approach. In Symposium on Software Architectures and Component Technology: The State of the Art in Software Development (2000), IBM T. J. Watson Research Center, Kluwer.
- Rashid, A., Sawyer, P., Moreira, A. M. D., and Araujo, J. Early Aspects: A Model for Aspect-Oriented Requirements Engineering. In Proceedings of IEEE Joint International Conference on Requirements Engineering (RE) (2002), pp. 199-202.
- Tarr, P., Ossher, H., Harrison, W., and Sutton, J. S. M. N Degrees of Separation: Multi-Dimensional Separation of Concerns. In Proceedings of International Conference on Software Engineering (ICSE) (1999), IEEE Computer Society Press, pp. 107-119.

## 9. Actividades propuestas.

Unidad	Actividad
1	<ul style="list-style-type: none"> <li>• Analizar las limitaciones del modelo de objetos mediante la revisión de cada extensión propuesta.</li> </ul>
2	<ul style="list-style-type: none"> <li>• Comparar los modelos de puntos de unión.</li> </ul>
3	<ul style="list-style-type: none"> <li>• Programar ejemplos para ilustrar el corte dinámico.</li> <li>• Programar ejemplos para ilustrar el corte estático.</li> <li>• Programar ejemplos para comprender la importancia de la activación dinámica de aspectos.</li> <li>• Comparar el corte del modelo de puntos de unión de AspectJ con las máquinas de estado de ECaesarJ.</li> <li>• Programar ejemplos de aseveraciones en Java y explicar la importancia del diseño por contratos.</li> </ul>

	<ul style="list-style-type: none"> <li>• Desarrollar ejemplos de contratos traslúcidos.</li> </ul>
4	<ul style="list-style-type: none"> <li>• Investigar los conceptos de clases virtuales.</li> <li>• Programar ejemplos de polimorfismo familiar.</li> </ul>
5	<ul style="list-style-type: none"> <li>• Investigar las formas de reutilización de componentes sin utilizar herencia ni composición.</li> <li>• Desarrollar ejercicios de envolventes, composición por tipificación y enlazado.</li> </ul>

## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: Seminario I  
Línea de la Asignatura: Ingeniería de Software  
Tecnologías Web

Tiempo de dedicación del estudiante a las actividades de:  
DOC-TIS-TPS-HRSTOT-CREDITOS  
(32 - 20 - 50 - 102 - 4)

### 1. Historial de la asignatura

Fecha de revisión / actualización	Participantes	Observaciones (cambios y justificación)
5 de septiembre de 2011	Carlos Alberto Ronquillo Salas, Jorge Alberto Gálvez Choy, Herlinda Silva Poot, José de Jesús Casa Jiménez, Cristóbal Aguirre Calderón, Oscar Mario Rodríguez Elías, Rocío Antonio Cruz, Beatriz Barrientos Becerra, Javier Ortiz Hernández, Socorro Sáenz Sánchez	Propuesta Preliminar Definición de la asignatura

### 2. Pre-requisitos y correquisitos.

Prerrequisitos	
Asignatura	Temas
Ninguna	

Correquisitos	
Asignatura	Temas
Ninguna	

### 3. Objetivo de la asignatura

Analizar las tendencias en investigación del área y orientación del programa de estudios correspondiente, considerando el papel de la ciencia y la tecnología, y su

relación con el proceso de innovación en el mundo contemporáneo y sus implicaciones éticas, ambientales, sociales y económicas, para elaborar el planteamiento del problema del proyecto de investigación o de trabajo.

#### 4. Aportación al perfil del graduado.

La asignatura contribuye a la conformación de una actitud crítica, responsable y propositiva en el egresado, ante las implicaciones éticas, ambientales y económicas del proceso de generación y aplicación del conocimiento científico e innovación tecnológica. Asimismo, le permitirá utilizar estos conocimientos en el desarrollo del proyecto de tesis.

#### 5. Contenido temático.

Unidad	Tema	Subtemas
1.	El papel de la ciencia y la tecnología en el mundo contemporáneo	1.1 El proceso tecnológico científico 1.2 Logros y retos de la Ciencia y la Tecnología 1.3 Desarrollo sustentable 1.4 Ética y responsabilidad social
2	Prospectiva de la investigación científica y tecnológica en el área específica	2.1 Análisis y delimitación de tendencias acorde a las líneas de investigación o de trabajo del programa 2.2 Identificación de áreas de oportunidad en el campo de investigación o de trabajo del programa 2.3 Selección y valoración de áreas de oportunidad desde un enfoque de desarrollo sustentable y responsabilidad ética y social
3	Conceptos y teoría de innovación	3.1 Conceptos de innovación 3.2 Modelos de innovación 3.3 Aspectos generales de propiedad intelectual 3.4 El entorno de la innovación en México



4	Planteamiento del problema de investigación o de trabajo	4.1 Conceptualización y análisis del problema 4.2 Delimitación del problema 4.3 Estado del arte
---	----------------------------------------------------------	-------------------------------------------------------------------------------------------------------

## 6. Metodología de Desarrollo del Curso.

Se sugieren las siguientes actividades como parte del curso:

Unidad	Tema	Subtemas
1.	El papel de la ciencia y la tecnología en el mundo contemporáneo	Lectura comentada Estudio de casos Foros de discusión Elaboración de ensayos
2	Prospectiva de la investigación científica y tecnológica en el área específica	Exposición de temas Investigación documental Técnicas de análisis para valoración de oportunidad Extrapolación de tendencias
3	Conceptos y teoría de innovación	Lectura comentada Estudio de casos Foros de discusión Elaboración de ensayos
4	Planteamiento del problema de investigación o de trabajo	Búsqueda en bancos de datos o de patentes Análisis bibliográfico

## 7. Sugerencias de evaluación.

Para la evaluación del Seminario se proponen las siguientes actividades:

- Participación activa en las actividades programadas como foros, lecturas recomendadas, entre otras a fin de evidenciar habilidades argumentativas
- Elaboración de documentos tales como ensayos, reporte de investigación, síntesis entre otros, que demuestren el manejo y aplicación de conceptos revisados en la asignatura
- Desarrollo y defensa de la propuesta preliminar del planteamiento del problema del proyecto de investigación o de trabajo

## 8. Bibliografía y Software de Apoyo.

- Chalmers, A. F. (182) ¿Qué es esa cosa llamada ciencia? Madrid, España. Siglo XXI Editores
- Liz, Manuel (1995) conocer y actuar a través de la tecnología. En: Broncano, Francisco. Nuevas meditaciones sobre técnica. Ed. Trotta. Madrid
- Gutiérrez Garza, E. 2010. De las teorías del desarrollo al desarrollo sustentable. México, Siglo XXI
- Olivé, león. 2007 La ciencia y la tecnología en la sociedad del conocimiento. Ética, política y epistemología. México, FCE:
- Jonas, Hans. 2004. El principio de responsabilidad: ensayo de una ética para la civilización tecnológica. España, Editorial Herder.
- García Palacios, E. M., González Galbarte, J.C., López Cerezo, J.A., Luján, J. L., Gordillo Mariano, M., Osorio, C. y Vladés, C. (2001). Ciencia, Tecnología y Sociedad: una aproximación conceptual. Organización de Estados Iberoamericanos, S.A., Madrid, España
- Chía, J. Y Escalona, C. (2009). La medición del impacto de la ciencia, la tecnología y la innovación en Cuba: análisis de una experiencia. Revista CTS 5(13), pp. 83-96. (<http://www.oei.es/cienciayuniversidad/spip.php?article899>).
- Castells, M. (1994). Silicon Valley, donde todo comenzó. En: Castells, M. y Hall, P. Las Tecnópolis del mundo. Alianza, Madrid, España.
- Ortégón, E., Pacheco, J. Y Prieto, A. (2005). Metodología del marco lógico para la planificación, el seguimiento y la evaluación de proyectos y programas. (ILPES, Santiago de Chile, Chile.
- Christensen, C. 2003. The innovator's solution. Boston, MA.: Harvard Business School Press.
- Chesbrough, H. 2006. Open innovation. Boston, MA.: Harvard Business School Press.
- Von Hippel, E. 1988. The sources of innovation. New York: Oxford University Press.
- Escorsa, P. Y Valls, J. 2003. Tecnología e innovación en la empresa. Ediciones UPC.
- Instituto Mexicano de la Propiedad Industrial. [www.impi.gob.mx](http://www.impi.gob.mx)
- Bazdresch, C. y Meza, L., 2011. La tecnología y la innovación como motores del crecimiento de México. Fondo de Cultura Económica.
- Sagasti, F., 2001. Ciencia, tecnología, innovación. Políticas para América Latina. Fondo de Cultura Económica.

### Bibliografía complementaria.

- Díaz, R. 2009. Desarrollo sustentable. México, McGraw-Hill Interamericana.
- Edward, A. R. y D.W. Orr. 2005. The Sustainability Revolution. Portrait of a Paradigm Shift, USA, Kindle Edition.

- Esquirol, J.M. 2006. El respeto a la mirada atenta: una ética para la era de la ciencia y la tecnología. España, GEDISA.
- Quintero Soto, M.L. y C. Fonseca Hernández. 2008. Desarrollo sustentable: aplicaciones e indicadores. México, Porrúa.
- Medina, M. et al. (2000) ciencia Tecnología / Naturaleza, Cultura en el Siglo XXI. Anthropos, UAM, Madrid, España.
- Hernández Sampieri, R., Fernández collado, C. y Baptista Lucio, P. (2005). Metodología de la investigación. McGraw Hill, México, México
- Friedman, T. 2005. The world is flat. New York: Farrar, Straus and Giroux.
- Drucker, P. 2002. The discipline of innovation. Harvard Business Review.
- Rangel medina, D. 1998. Derecho intelectual. McGraw-Hill, México, México.
- CEPAL, 2009. Innovar para crecer. Santiago de Chile; Naciones Unidas.
- Van Agtmael, A. 2007. The emerging markets century. New York; Free Press.
- Organización Mundial de la Propiedad intelectual. [www.xipo.int](http://www.xipo.int)
- Instituto Nacional del Derecho de Autor. [www.indautor.gob.mx](http://www.indautor.gob.mx)

## 9. Actividades propuestas.

Unidad, tema	Actividad
1, 1.1	1. Comentar las lecturas señaladas en clase, identificando los argumentos más relevantes desde diferentes puntos de vista
1, 1.3	2. A partir de la selección de un caso vinculado a su área, desarrollar un ensayo, considerando un enfoque hacia el desarrollo sustentable.
	3. Participar en foros de discusión , argumentando y defendiendo su punto de vista acerca de un tema.
1, 1.4	4. Analizar casos de estudio de carácter técnico científico e identificar sus implicaciones éticas.
2, 2.1. 2.2	5. Identificar fuentes de información relacionadas con su área de trabajo, y hacer un listado priorizándolas (bases de datos bibliográficas, congresos, revistas, etc.)
	6. Realizar una revisión de bibliografía en las fuentes encontradas para identificar temas de actualidad y relevancia en su área de investigación o de trabajo; realizar un documento escrito.
2, 2.3	7. Aplicar alguna técnica para valorar los temas seleccionados en términos de su impacto ambiental, económico, social y ético.
	8. Exposición ante el grupo del tema seleccionado, argumentando con claridad, pertinencia y viabilidad.
3, 3.1	9. Elaborar un mapa conceptual en que se presenten los puntos más significativos del concepto de innovación
3, 3.2	10. Estudio de casos de los diferentes modelos de innovación

3, 3.3	11. Realizar búsquedas en bases de datos y búsquedas sobre patentes a nivel nacional e internacional sobre el tema de investigación o de trabajo para verificar su originalidad e impacto.
3, 3.4	12. Elaborar ensayo sobre el entorno de la innovación en México con estadísticas sobre patentes, apoyos a la investigación-innovación, estructura de financiamiento (ángeles, ventura, etc.)
4, 4.1	13. Conceptualizar el problema específico a tratar dentro de la temática seleccionada
4, 4.2	14. Elaborar un documento describiendo el problema, justificándolo y delimitándolo en el espacio y el tiempo, así como en función de su impacto ambiental, económico, social y ético.
4, 4.3	15. Buscar en bancos de datos, patentes, entre otros, si existen trabajos que busquen resolver el problema planteado, o que estén relacionados con el mismo.
	16. Elaborar el árbol de problemas con la técnica de marco lógico.
	17. Elaborar un escrito sobre el estado del arte del problema planteado, y sus posible soluciones

#### **10. Nombre y firma del catedrático responsable.**

**Dra. Herlinda Silva Poot**

**Dr. Javier Ortiz Hernández**

**M.C. Beatriz Barrientos Becerra**

**M.C. Socorro Sáenz Sánchez**

**M.C. Cristobal Gerardo Aguirre Calderón**

**Dr. Oscar mario Rodríguez Elías**

**M.C. Jorge Alberto Gálvez Choy**

**M.C. Carlos Alberto Ronquillo Salas**

## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: TÉCNICAS AVANZADAS DE PROGRAMACIÓN  
Línea de la Asignatura: Ingeniería de Software

Tiempo de dedicación del estudiante a las actividades de:  
DOC-TIS-TPS-HRSTOT-CRÉDITOS  
(48 – 20 – 100 – 168 - 6)

### 1. Historial de la asignatura

Fecha de revisión / actualización	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo de 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta preliminar Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre de 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba 8 al 12 de julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba 6 al 10 de agosto de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

### 2. Pre-requisitos y co-requisitos

Pre-requisitos		Co-requisitos	
Asignatura	Temas	Asignatura	Temas
Tecnologías de programación			

### 3. Objetivo de la asignatura

Aplicar los mecanismos de programación con genéricos, meta-programación, reutilización mediante *mixins* y envolventes, así como de programación generativa para el desarrollo de aplicaciones reconfigurables a tiempo de ejecución.

#### 4. Aportación al perfil del graduado

Este curso proporciona al egresado las herramientas necesarias para resolver problemas que requieren de formas avanzadas de reutilización útiles en el contexto de objetos, componentes, arquitecturas de software y nuevos paradigmas.

#### 5. Contenido temático

Unidad	Temas	Subtemas
1	Programación con genéricos y polimorfismo con genéricos.  DOC: 12.  Objetivo: Conocer y el polimorfismo con genéricos y su aplicación en colecciones de datos.	1.1 Genéricos 1.2. Límites 1.3. Polimorfismo
2	Meta-programación.  DOC: 12.  Objetivo: Conocer las anotaciones, meta-anotaciones y su aplicación para el desarrollo de programas procesadores de programas.	2.1. Anotaciones básicas en Java 2.2. Programación de anotaciones 2.3. Procesadores de anotaciones
3	Polimorfismo en familias de tipos: mixins y envoltentes.  DOC: 12.  Objetivo: Conocer las capacidades de polimorfismo en el contexto de variabilidad.	3.1. Mixins 3.2. Polimorfismo con mixins 3.3. Wrappers 3.4. Polimorfismo con wrappers 3.5. Variabilidad
4	Programación generativa.  DOC: 12.  Objetivo: Conocer la programación generativa y su aplicación en la reutilización de componentes a tiempo de ejecución.	4.1. Introducción 4.2. Componentes auto configurables 4.3. Generadores 4.4. Variabilidad 4.5. Reutilización a tiempo de ejecución

#### 6. Metodología de desarrollo del curso

Se sugieren las siguientes actividades como parte del curso:

- Exposición por parte del profesor.
- Análisis de ejemplos y casos.
- Investigación de tecnología disponible.
- Prácticas.

## 7. Sugerencias de evaluación

Se recomienda realizar evaluación en cada sesión, con base a las siguientes actividades:

- Ejercicios en clase.
- Ejercicios extra clase.
- Exposición de temas.
- Exámenes teórico-prácticos.
- Desarrollo de proyectos.

## 8. Bibliografía y software de apoyo

- Bruce Eckel; Thinking in Java 4<sup>th</sup> ed.; Prentice Hall.
- CaesarJ Project: <http://caesarj.org/>
- ECaesarJ Project: <http://ample.holos.pt/pageview.aspx?pageid=80&langid=1>
- Krzysztof Czarnecki; Generative Programming; <http://www.issi.uned.es/doctorado/generative/Bibliografia/TesisCzarnecki.pdf>
- Javassist; <http://www.csg.ci.i.u-tokyo.ac.jp/~chiba/javassist/>

## 9. Actividades propuestas

Unidad	Actividades
1	<ul style="list-style-type: none"> <li>• Desarrollar ejemplos de genéricos.</li> <li>• Desarrollar ejercicios aplicando límites.</li> </ul>
2	<ul style="list-style-type: none"> <li>• Aplicar las anotaciones básicas de Java.</li> <li>• Desarrollar ejercicios de anotaciones aplicando los procesadores disponibles en Java, tal como APT.</li> </ul>
3	<ul style="list-style-type: none"> <li>• Desarrollar ejemplos de colaboraciones en CaesarJ o en ECaesarJ</li> <li>• Desarrollar ejemplos de mixins y wrappers con CaesarJ o con ECaesarJ.</li> <li>• Desarrollar ejemplos de variabilidad.</li> </ul>
4	<ul style="list-style-type: none"> <li>• Revisar GenVoca.</li> <li>• Desarrollar ejemplos con Javassist a partir de las ideas de la herramienta Jenerator.</li> </ul>

## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: TECNOLOGÍAS DE LA WEB SEMÁNTICA  
Línea de la Asignatura: Tecnologías Web

Tiempo de dedicación del estudiante a las actividades de:  
DOC-TIS-TPS-HRSTOT-CREDITOS  
(48 - 20 - 100 - 168 - 6)

### 1. Historial de la asignatura

Fecha de revisión / actualización	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preliminar Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba 8 al 12 de julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba 6 al 10 de agosto de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura

### 2. Pre-requisitos y correquisitos.

Prerrequisitos	
Asignatura	Temas

Correquisitos	
Asignatura	Temas



Diseño de ontologías	
----------------------	--

--	--

### 3. Objetivo de la asignatura

Aplicar los conceptos e implicaciones de la Web semántica, así como los diferentes estándares (XML, RDF, FOAF, OWL, SKOS, etc.) y tecnologías implicadas que envuelven el desarrollo actual y futuro de la Web, y los nuevos retos que implica la creación de una Web más inteligente.

### 4. Aportación al perfil del graduado.

Este curso proporciona al egresado los elementos necesarios para solucionar problemas prácticos relacionados con el desarrollo de proyectos enfocados a la Web Semántica, permitiéndole construir aplicaciones robustas bajo el enfoque de las nuevas tecnologías

### 5. Contenido temático.

Unidad	Tema	Subtemas
1.	Conceptos, estándares y usos de la Web Semántica.  DOC: 12  Objetivo: conocer el marco conceptual de la Web Semántica	1.1 Introducción 1.2 Justificación 1.3 Conceptos generales 1.4 Generalidades de XML, RDF, DAML, OWL
2	Ejemplos y casos de uso de la Web Semántica.  DOC: 12  Objetivo: conocer los usos y aplicaciones más comunes de la Web Semántica	2.1 Casos de éxito 2.2 Descripción y recuperación de recursos con RDF 2.3 Tesauros
3	Estándares y Tecnologías para la Representación del Conocimiento en la Web Semántica.  DOC: 12  Objetivo: conocer y aplicar los	3.1 RDF 3.2 OWL 3.3 FOAF 3.4 SKOS 3.5 SEM

	estándares principales de la Web Semántica así como las herramientas que los soportan	
4	Servicios web en la Web Semántica  DOC: 12  Objetivo: conocer y aplicar los conceptos de servicios web dentro de la Web Semántica	4.1 Servicios web 4.2 DAML-S 4.3 OWL - S

## 6. Metodología de Desarrollo del Curso.

Se sugieren las siguientes actividades como parte del curso:

- Exposiciones por parte del profesor
- Discusiones en grupo
- Resolución de casos de estudio
- Prácticas

## 7. Sugerencias de evaluación.

La evaluación de la materia debe realizarse día a día, basándose en el desempeño de las diferentes actividades:

- Ejercicios realizados durante clase
- Ejercicios realizados de forma independiente
- Reportes de investigaciones (por ejemplo: resúmenes, mapas conceptuales, mapas mentales y conclusiones entre otros)
- Exposición de temas y casos resueltos.
- Desarrollo de proyectos que integren los temas vistos en clase.
- Exámenes teóricos y/o prácticos.

## 8. Bibliografía y Software de Apoyo.

- Foundations of Semantic Web Technologies. Pascal Hitzler et al. Chapman & Hall/CRC Textbooks in Computing
- Mining the Social Web: Analyzing Data from Facebook, Twitter, LinkedIn, and Other Social Media Sites. Matthew A. Russell. O'Reilly
- Owl: Representing Information Using the Web Ontology Language. Lee W. Lacy. ISBN 978-1412034487
- Practical RDF. Shelley Powers. O'Reilly. ISBN 978-0596002633
- Programming the Semantic Web. Toby Segaran. O'Reilly. ISBN 978-0596153816

- Semantic Web Programming. John Hebel et al. ISBN 978-0470418017
- SEWESE <http://www-sop.inria.fr/teams/edelweiss/wiki/wakka.php?wiki=Sewese>
- JENA. <http://jena.apache.org>
- Protégé. <http://protege.stanford.edu>
- OWL. <http://owl.w3.org/2004/OWL>
- Sesame. <http://www.openrdf.org>
- RDF. <http://www.w3.org/RDF/>
- FOAF. <http://www.foaf-project.org/>
- Python. <http://www.python.org/>

## 9. Actividades propuestas.

Unidad	Actividad
1	<ul style="list-style-type: none"> <li>• Investigar conceptos</li> <li>• Plantear escenarios de aplicaciones donde se utilicen los conceptos de la web semántica</li> <li>• Plantear aplicación donde mezclen ontologías y web semántica</li> </ul>
2	<ul style="list-style-type: none"> <li>• Investigar casos de éxito</li> <li>• Realizar ontología con RDF y recuperación de información sobre ella</li> <li>• Realizar una aplicación web sobre la ontología</li> </ul>
3	<ul style="list-style-type: none"> <li>• Realizar ejemplos de ontologías con RDF y OWL</li> <li>• Realizar ejemplos con FOAF</li> <li>• Realizar ejemplos con SKOS</li> </ul>
4	<ul style="list-style-type: none"> <li>• Realizar aplicación semántica que consuma servicios web “estándar”</li> <li>• Realizar aplicación que utilice OWL-S</li> </ul>

## TEMARIO DE LA ASIGNATURA

Nombre de la asignatura: TENOLOGÍAS DE INTERNET

Línea de la Asignatura: Tecnologías Web

Tiempo de dedicación del estudiante a las actividades de:

DOC-TIS-TPS-HRSTOT-CREDITOS

(48 - 20 - 100 - 168 - 6)

### 1. Historial de la asignatura

<b>Fecha de revisión / actualización</b>	<b>Participantes</b>	<b>Observaciones (cambios y justificación)</b>
Instituto Tecnológico de Orizaba 10 a 12 de marzo 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Propuesta Preliminar  Definición de la asignatura
Instituto Tecnológico de Orizaba 10 a 12 de septiembre 2010	Docentes del Sistema Nacional de Educación Superior Tecnológica	Segunda reunión de consolidación de los posgrados (computación)
Instituto Tecnológico de Orizaba 8 al 12 de julio de 2011	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Análisis de la asignatura
Instituto Tecnológico de Orizaba 9 al 13 de enero de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Conformación de la asignatura
Instituto Tecnológico de Orizaba 6 al 10 de agosto de 2012	Consejo de Posgrado de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba	Definición del temario final

### 2. Pre-requisitos y correquisitos.

Prerrequisitos	
Asignatura	Temas

Correquisitos	
Asignatura	Temas

### 3. Objetivo de la asignatura

Comprender los protocolos y estándares internacionales de las comunicaciones y Tecnologías de Internet, instalar y configurar redes locales, diseño de la red, especificaciones y justificaciones de sus elementos; principales tendencias y tecnologías en el ámbito de la interconexión, modelado de sistemas WAN

### 4. Aportación al perfil del graduado.

Al finalizar la materia el alumno contará con los conocimientos de implementación de listas de acceso, determinación de rutas, implementación de servicios de red; así como configuración de routers.

### 5. Contenido temático.

Unidad	Temas	
1	<p>Introducción a la tecnología de internet</p> <p>DOC 10</p> <p>Objetivo: Conocer las definiciones, los medios de comunicación actuales, y los sistemas de acceso a la red así como los sistemas, servicios, clientes, tecnologías y protocolos que se usan en Internet.</p>	<p>1.1 Definiciones de Internet</p> <p>1.2 Historia de la Internet</p> <p>1.3 Servicios y Clientes</p> <p>1.4 Tecnologías y Protocolos</p> <p>1.5 Modelos de comunicación</p>
2	<p>Medios de Transmisión y señalización.</p> <p>DOC 15</p> <p>Identificar y reconocer las técnicas de Transmisión de señales y medios de</p>	<p>2.1. Transmisión de Señales</p> <p>2.1.1 Características de una señal analógica y una señal digital</p> <p>2.1.2. Señalización digital y analógica.</p> <p>2.1.3. Multicanalización</p> <p>2.2. Medios de Transmisión</p> <p>2.2.1. Guiados: cobre, fibra óptica</p> <p>2.2.2. No guiados: microondas, radio frecuencias, satelital</p>

	comunicación así como su funcionamiento	2.2.3. Problemas en la transmisión de señales 2.2.4. Estándar EIA/TIA 568, 569
3	<p>Interconexión de servicios y protocolos</p> <p>DOC 20</p> <p>Objetivos: Conocer aplicar la configuración de un host en la red. Conocer los diferentes dispositivos de conectividad y saber en qué casos usar cada uno. Estudiar los algoritmos de ruteo más usados actualmente. Implementación de servicios de red.</p>	<p>3.1 Protocolos</p> <p>3.1.1 TCP/IP</p> <ul style="list-style-type: none"> <li>• Capas</li> <li>• Direccionamiento</li> <li>• Clases de redes</li> <li>• DNS y DHCP</li> <li>• Proxy y NAT</li> </ul> <p>3.1.2 Otros protocolos</p> <p>3.2 Algoritmos de ruteo</p> <p>3.2.1 Ruteo de datagramas IP</p> <p>3.2.2 Salto al Siguiente</p> <p>3.2.3 Vector distancia</p> <p>3.2.4 RIP, RIP II</p> <p>3.2.5 IGRP, EGRP</p> <p>3.2.6 BGP</p> <p>3.3 Servicios</p> <p>3.3.1 Servicios para internet (http, dns, ftp, mail)</p> <p>3.3.2 Servidores de aplicaciones (tomcat, iis, JOnAS)</p> <p>3.3.3 Servidores de almacenamiento (postgres, Sybase, Oracle)</p>
4	<p>Configuración de dispositivos y direccionamiento</p> <p>DOC 13</p> <p>Objetivos: entender y aplicar los conceptos de direccionamiento Lan y las estructuras con su interconexión WAN por medio de simuladores de redes</p>	<p>4.1 Modelo de Referencia TCP/IP</p> <p>4.1.1 Introducción a TCP/IP</p> <p>4.1.2 Direccionamiento IP</p> <p>4.2 Administración de direcciones IP (DHCP, asignación estática)</p> <p>4.3 Protocolos ruteables y de ruteo</p> <p>4.4 Subredes</p> <p>4.5 Capas de transporte y aplicación del modelo TCP/IP</p> <p>4.6 Introducción a Redes Conmutadas Ethernet</p> <p>4.6.1 Ethernet conmutado</p> <p>4.6.2 Dominios de colisión y broadcast</p> <p>4.6.3 Interconexión en redes LAN</p> <p>4.6.4 Interconexión en redes WAN</p>

## 6. Metodología de Desarrollo del Curso.

Se recomienda manejar un problema específico para cada unidad; exposiciones de los temas por parte del profesor, investigación de temas por parte de los alumnos, identificación de problemas y métodos para resolverlos por parte de casos de estudio.

## 7. Sugerencias de evaluación.

- A través de exposiciones y discusión en clases de los temas de todas las unidades dependiendo del tema en particular
- A través de las prácticas de instalación, configuración y administración de los diferentes servicios de red.

## 8. Bibliografía y Software de Apoyo.

Redes de Computadoras, Andrew S. Tenenbaum, 4ª Edición, Perason / Prentice Hall.

Redes Globales de Información con Internet y TCP/IP, 3ª ed., Douglas E. Comer, Pearson.

Interconectividad de redes con TCP/IP, Diseño e Implementación, Vol II 3ª. Ed., Douglas E. Comer, David L. Stevens, Pearson.

Redes locales e internet, introducción a la comunicación de datos,  
St-Pierre Armand, Stéphanos William, Trillas.

Redes de computadoras, internet e interredes, Douglas E. Comer, Prentice Hall.

Manual de referencia CISCO, Brian Hill, Mc Graw Hill.

Microsoft Windows 2000 Active Directory Services, curso oficial de certificación MCSE,  
Microsoft Official Curriculum, Mc Graw Hill.

Data and computer communications, William Stallings, Mc Millan.

## 9. Actividades propuestas.

Se sugiere que las prácticas propuestas a continuación sean realizadas por equipos para fomentar la discusión de ideas e intercambio de experiencias que plantea el curso:

Tema	Práctica
Comunicación a través de la red	Describir la estructura de la red tomando en cuenta los dispositivos y medios necesarios para lograr una comunicación exitosa Explicar las comunicaciones de redes, por medio de las capas de protocolos OSI
Medios de transmisión	Visitar para identificación de los diferentes elementos de un site de comunicaciones Conexión de diferentes tipos de medios de comunicación identificación y practicas de cableado y conexiones inalámbricas
Ruteo	Arranque y configuración básica de un router Conexión de varios routers con diferentes tipos de protocolos

Direccionamiento	Creación de una red wan con al menos 4 routers configurados y direccionar diferentes elementos como cámaras ip, pc, teléfonos ip, etc.
------------------	----------------------------------------------------------------------------------------------------------------------------------------

**10.– Nombre y Firma del Profesor responsable.**

MRT Ignacio López Martínez