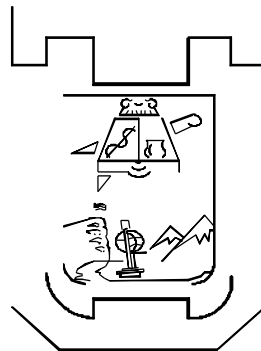


UNIVERSIDAD DE TARAPACA
ESCUELA UNIVERSITARIA DE INGENIERIA INDUSTRIAL,
INFORMÁTICA Y DE SISTEMAS
ÁREA DE INGENIERÍA EN COMPUTACIÓN E INFORMÁTICA
Arica - Chile
2015



PROGRAMA
MAGÍSTER EN INGENIERÍA DE
SOFTWARE

2 0 1 5

Contenido

I.- PRESENTACIÓN.....	4
II.- IDENTIFICACIÓN DEL PROGRAMA.....	6
III.- OBJETIVOS DEL PROGRAMA.....	6
3.1. Objetivo General.....	6
3.2. Objetivos Específicos:.....	6
IV.- PERFIL DEL MAGÍSTER EN INGENIERIA DE SOFTWARE.....	7
V.- GRADO Y DIPLOMA QUE OTORGA EL PROGRAMA.....	7
VI.- REQUISITOS DE INGRESO.....	8
VII.- DURACIÓN MÁXIMA DEL PROGRAMA.....	8
VIII.- CAMPO OCUPACIONAL POSIBLE.....	9
IX.- PLAN DE ESTUDIOS.....	9
9.1. Finalidad del Plan de Estudios.....	9
9.2. Aspectos Generales.....	9
9.3. Modalidad de Trabajo.....	10
9.4. Asignaturas del Plan de Estudios.....	11
9.5. Fluxograma del Plan del Programa de Magíster en Ingeniería de Software.....	12
9.6. Actividad Final de Postgrado.....	13
9.7. Requisitos de Graduación.....	13
X. PROGRAMAS DE ASIGNATURAS.....	14
- INGENIERÍA DE SOFTWARE AVANZADA.....	14
- ARQUITECTURA DE SOFTWARE.....	16

- ELECTIVO I.....	18
- GESTIÓN DE PROCESO Y CALIDAD DE SOFTWARE.....	19
- ELECTIVO II.....	21
- ELECTIVO III.....	22
- SEMINARIO DE PROYECTO DE GRADO	23
- ACTIVIDAD FINAL DE POSTGRADO.....	24
XI. EJECUCION.....	25
XII. PROFESORES.....	25
12.1 PROFESORES PERMANENTES DEL PROGRAMA.....	25
12.2 PROFESORES COLABORADORES.....	26
ANEXOS.....	27
- INGENIERIA DE REQUERIMIENTOS.....	28
- METODOS FORMALES DE INGENIERIA DE SOFTWARE.....	30
- TOPICOS AVANZADOS DE BASES DE DATOS.....	32
- INTELIGENCIA DE NEGOCIOS.....	35
- PROGRAMACION PARALELA MODERNA	37
- VISION COMPUTACIONAL	40
- DATA MINING EN LA WEB	42
- RECUPERACION DE INFORMACION.....	45

I.- PRESENTACIÓN.-

El programa de Magíster en Ingeniería de Software tiene una orientación profesional, cuyo principal propósito es formar profesionales con profundos conocimientos técnicos, teóricos y prácticos que los habiliten para dirigir y llevar a cabo proyectos de software exitosos, utilizando métodos y herramientas de actualidad ampliamente difundidos.

La competitividad de las organizaciones produce una constante demanda de ingenieros de software, para que desarrollen en el corto plazo y con calidad los sistemas de software que les permitan ser cada día más eficientes. La Ingeniería de Software, consciente de que el diseño y construcción de software es un proceso complejo que involucra y combina aspectos tecnológicos (computación y comunicaciones), y propios de las organizaciones (como la dinamicidad del medio que obliga a rápidos y constantes cambios, incluso durante la fase de construcción), se ha esforzado en el desarrollo de métodos y herramientas que lo apoyen. Sin embargo, los logros obtenidos no reflejan los esfuerzos realizados, y se continúa con los problemas de plazos de desarrollos no cumplidos, presupuestos largamente excedidos, productos de escasa calidad, y usuarios y clientes insatisfechos. Justificaciones a esta situación se pueden encontrar en:

- Desconocimiento por parte de los desarrolladores de aspectos de importancia crítica en el diseño y construcción del software, y
- Falta de rigurosidad para enfrentar el proceso de desarrollo de sistemas de Software.

Por lo tanto, para cumplir con las demandas organizacionales, relativas al desarrollo de producto de software, se requiere de profesionales con el perfil inicialmente descrito que hagan del diseño y la construcción de software una práctica de ingeniería, es decir, que exista un proceso de desarrollo sistemático y controlado.

Actualmente, a nivel nacional, existe una variada oferta de programas iguales o similares en el área de la Ingeniería de Software. Por otro lado, tanto a nivel local, como regional, existe un particular interés de parte de Ingenieros del área de Computación que se desempeñan en instituciones privadas, educacionales, y del sector público, por seguir un programa de esta naturaleza. La demanda de un programa de Magíster como el que se propone, puede ir más allá que lo meramente local y regional, principalmente debido a:

- Existencia de un particular interés por cursar en la Universidad de Tarapacá programas de formación profesional y especialización de parte de personas y profesionales de países vecinos, tales como Perú y Bolivia y
- Existencia en la ciudad de Santiago de una masa crítica de ex alumnos de las carreras de Ingeniería Civil/Ejecución en Computación e Informática, particularmente interesados.

Desde el punto de vista de los objetivos institucionales, este programa le permitiría a la Universidad de Tarapacá los siguientes beneficios:

- Proyección dentro del contexto de su política de Internacionalización dirigida a países vecinos, y la consolidación de un polo de desarrollo importante entre las Universidades Regionales de la zona Norte.
- Mantener presencia en el resto del país por medio de la provisión de educación continua a sus titulados del área de computación en particular.

II.- IDENTIFICACIÓN DEL PROGRAMA.-

DENOMINACIÓN DEL PROGRAMA	: MAGÍSTER EN INGENIERÍA DE SOFTWARE
GRADO QUE OTORGA	: MAGÍSTER EN INGENIERÍA DE SOFTWARE
DURACIÓN DE LOS ESTUDIOS	: TRES SEMESTRES ACADÉMICOS
RÉGIMEN	: MIXTO
MODALIDAD	: PRESENCIAL O SEMIPRESENCIAL
TIPO DE CURRÍCULUM	: SEMIFLEXIBLE
COBERTURA	: NACIONAL O INTERNACIONAL
UNIVERSIDAD	: UNIVERSIDAD DE TARAPACÁ
ESCUELA RESPONSABLE	: ESCUELA UNIVERSITARIA DE INGENIERÍA INDUSTRIAL, INFORMÁTICA Y DE SISTEMAS
UNIDAD RESPONSABLE	: ÁREA DE INGENIERÍA EN COMPUTACIÓN E INFORMÁTICA.

III.- OBJETIVOS DEL PROGRAMA.-

3.1. Objetivo General:

El programa de Magíster en Ingeniería de Software corresponde a un ciclo de formación de tipo profesional, orientado a mejorar las prácticas en la disciplina de desarrollo de software y la calidad de los productos obtenidos.

3.2. Objetivos Específicos:

- Adquirir el conocimiento técnico, teórico y práctico, requerido para el desarrollo efectivo de grandes y complejos sistemas de software.
- Desarrollar las habilidades requeridas para llevar a cabo un proyecto de software exitoso.

- Utilizar los estándares, métodos, herramientas y tecnologías actualmente adoptadas por la industria.
- Familiarizarse con las innovaciones más promisorias de la disciplina.

IV.- PERFIL DEL MAGÍSTER EN INGENIERIA DE SOFTWARE.-

El programa está dirigido principalmente a ingenieros del área de computación y habilitará a los participantes a gestionar eficientemente proyectos de compleja envergadura y desarrollar sistemas informáticos exitosos.

Al finalizar el programa, el graduado de Magíster en Ingeniería de Software deberá evidenciar:

- Conocimiento y competencia para contribuir efectivamente en el proceso de desarrollo de grandes y complejos sistemas de software.
- Capacidad para gerenciar proyectos de software.
- Capacidad para asesorar técnicamente proyectos organizacionales que consideran la aplicación de tecnologías de la información.

V.- GRADO y DIPLOMA QUE OTORGA EL PROGRAMA.-

Al aprobar todas las asignaturas, incluyendo la Tesis, al alumno se le otorgará el grado de:

- **Magíster en Ingeniería de Software.**

Si el alumno aprueba todas las asignaturas, sin incluir la Tesis, puede solicitar el Diploma de:

- **Diplomado en Ingeniería de Software**

VI.- REQUISITOS DE INGRESO.-

Podrán ingresar directamente al programa todas aquellas personas que posean un título profesional o grado con al menos cuatro años de estudios formales, que sean del área de computación y/o informática o que acrediten tener conocimientos en lenguajes de programación, algoritmos y estructuras de datos, bases de datos, sistemas operativos, redes e Ingeniería de Software. Otras personas, con título profesional de al menos cuatro años, podrán ingresar al programa, siempre y cuando se sometan a un período de nivelación determinado por el Comité Académico del Programa, consistente de algunas de las siguientes asignaturas:

- Algoritmos y estructuras de datos.
- Bases de datos.
- Sistemas Operativos.
- Redes y Comunicación de datos.
- Ingeniería de Software

VII.- DURACIÓN MÁXIMA DEL PROGRAMA.-

El programa tiene una extensión de tres semestres académicos. Sin embargo, los alumnos que ingresan al programa podrán convalidar las asignaturas descritas en los dos primeros semestres del plan.

De no ser concluida la Actividad Final de Postgrado en los plazos estipulados, el alumno deberá solicitar prórroga por un semestre. Este podrá solicitar un máximo de dos prórrogas en el transcurso de su permanencia en el programa.

VIII.- CAMPO OCUPACIONAL POSIBLE.-

El graduado de este programa podrá desarrollarse profesionalmente en:

- Organizaciones de cualquier índole que posean áreas propias de desarrollo de sistemas de software.
- Organizaciones orientadas a brindar servicios de desarrollo de sistemas de software.
- Prestaciones de asesorías orientadas a la incorporación de tecnologías y al desarrollo de sistemas de software en las organizaciones.
- Instituciones de educación como docente del área de Ingeniería de Software.

IX.- PLAN DE ESTUDIOS.-

9.1 Finalidad del Plan de Estudios:

El presente plan de estudios se orienta básicamente a la formación de Magísteres en Ingeniería de Software.

9.2 Aspectos Generales:

El programa se desarrollará en el Área de Ingeniería en Computación e Informática, perteneciente a la Escuela Universitaria de Ingeniería Industrial, Informática y de Sistemas de la Universidad de Tarapacá. Para estos efectos el área cuenta con los siguientes antecedentes.

- Experiencia de casi treinta años en la formación de profesionales en el área de computación, los que gozan de un reconocido prestigio en el ámbito nacional.
- Experiencia de haber dictado en reiteradas oportunidades los programas Magíster en Ingeniería de Software, de Post Título de Diplomado en Informática y Diplomado en Ingeniería de Software en Arica Chile y en el Perú, en las ciudades de Tacna, Arequipa y Lima.

- Actualmente su planta académica está conformada por:
 - ✓ Cuatro ingenieros con el grado de Doctor en la especialidad.
 - ✓ Dos ingenieros con el grado de Magíster o Master en la especialidad.
 - ✓ Un Ingeniero Civil en Computación e Informática
 - ✓ Un Ingeniero Civil Químico

- Medios y Laboratorios adecuadamente equipados con hardware y software de diversas plataformas y actualizados a las necesidades de hoy en día:
 - ✓ Salas de trabajo grupal
 - ✓ Pizarra digital
 - ✓ Televisores
 - ✓ Red Cableada e Inalámbrica
 - ✓ Sala de Servidores
 - ✓ Laboratorio de Ingeniería de Datos
 - ✓ Laboratorio de Redes Computacionales
 - ✓ Laboratorio de Programación

9.3 Modalidad de Trabajo:

El programa de Magíster en Ingeniería de Software se desarrollará bajo modalidad presencial o semipresencial. La modalidad semipresencial es para posibilitar la implantación del programa en ciudades distintas a Arica y eventualmente en el extranjero, en tal caso, las actividades de aprendizajes deberán reforzarse mediante el uso de tecnologías de la información y de las comunicaciones.

9.4 Asignaturas del Plan de Estudios:

Nro. Asig.	Asignaturas	Pre Requisito	Nro. de Créditos
	Primer Semestre (I)		
11	Ingeniería de Software Avanzada	CI	12
12	Arquitectura de Software	CI	8
13	Electivo I	AD	8
Segundo Semestre (II)			
21	Gestión de Proceso y Calidad de Software	CI	8
22	Electivo II	AD	8
23	Electivo III	AD	8
24	Seminario Proyecto de Grado.	I	8
Tercer Semestre (III)			
31	Actividad Final de Postgrado	II	30
Total			90

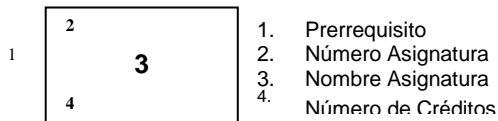
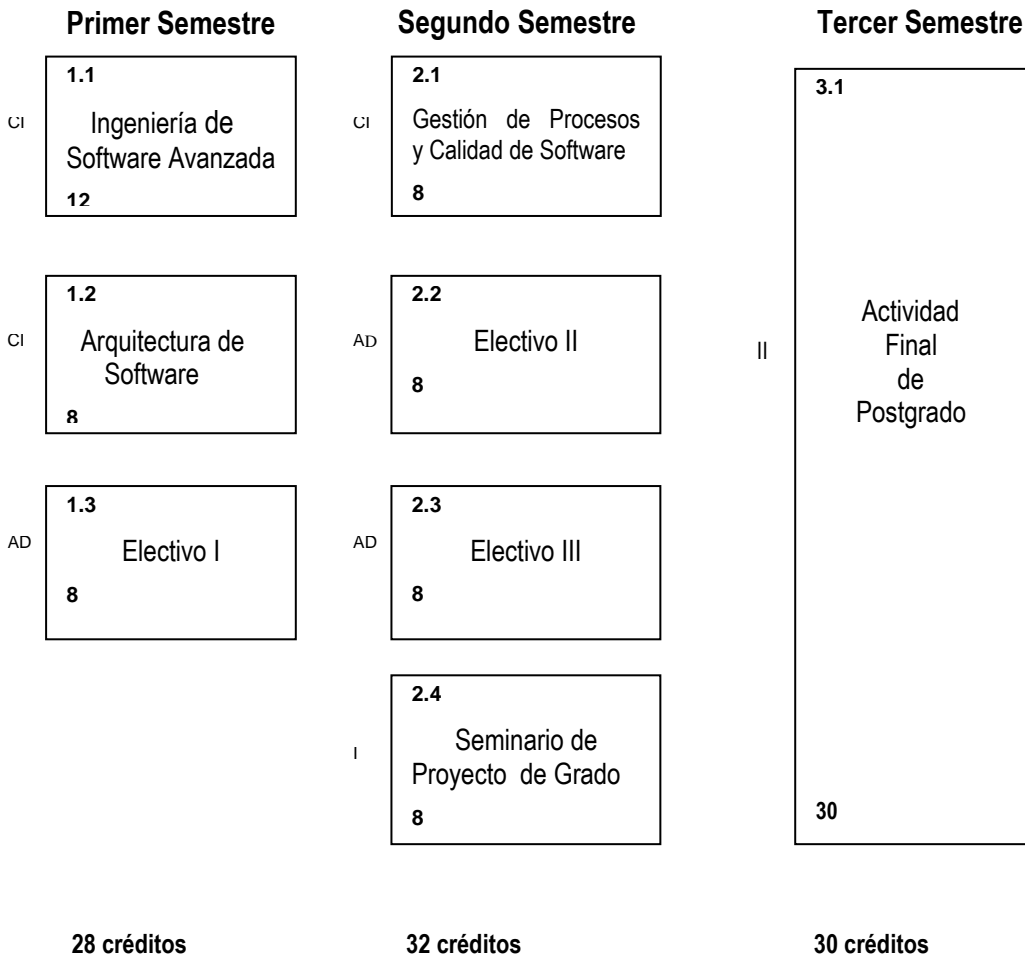
CI : Condiciones de Ingreso

AD: A definir según el electivo

I : Haber completado el primer semestre

II : Haber completado el segundo semestre

9.5 Fluxograma del Plan del Programa de Magíster en Ingeniería de Software:



CI: Condiciones de Ingreso.
 AD: A Definir según contenidos.
 I: Primer Semestre.
 II. Segundo Semestre

Total de créditos (SCT) del programa: 90

9.6 Actividad Final de Postgrado

Como actividad final de postgrado, el estudiante del programa deberá desarrollar un trabajo original de integración de conocimientos o investigación, de carácter individual, bajo la supervisión directa de un profesor tutor. Este trabajo o investigación, debe ser de nivel y complejidad tal, que cumpla con el perfil de formación profesional del programa.

9.7 Requisitos de Graduación

Para obtener el grado académico de Magister en Ingeniería de Software de la Universidad de Tarapacá, el estudiante deberá cumplir con los siguientes requisitos:

- Haber aprobado todos los cursos del plan del Programa de Ingeniería de Software de acuerdo a las normas establecidas en el “Reglamento de Docencia de Programas de Post-grado de la Universidad de Tarapacá” (Decreto Exento No 00.681/2007, y su modificación: Decreto Exento 00.309/2010).
- Haber concluido exitosamente con la Actividad Final de Postgrado del plan del Programa de Ingeniería de Software, cuyo desarrollo y evaluación deberá ceñirse a las normas y procedimientos establecidos en el “Reglamento para la Presentación, Desarrollo y Evaluación de la Actividad Docente Terminal de los Programas de Magister de la EUIIIS” (Resolución Exenta EUI.IIS No. 0.22/2010, y su modificación: Resolución Exenta EUI.IIS No. 0.100/2013).

X.- PROGRAMAS DE ASIGNATURAS.-

PROGRAMA ASIGNATURA	: INGENIERIA DE SOFTWARE AVANZADA
CARÁCTER	: OBLIGATORIO
PRE-REQUISITOS	: CONDICIONES DE INGRESO.
SEMESTRE CURRICULAR	: PRIMERO
HORAS PRESENCIALES	: 6 (4,0,2)
CRÉDITOS (SCT)	: 12

Descripción:

La asignatura Ingeniería de Software Avanzada tiene como objetivo el profundizar en aspectos de especificación, diseño, metodologías y otros, propios de la Ingeniería de Software.

Objetivo General:

- Aplicar conceptos, experiencias avanzadas y analizar el estado del arte en la Ingeniería de Software.

Objetivos Específicos:

Al término del curso el alumno será capaz de:

- Diseñar una interfaz gráfica de usuario para un software
- Especificar formalmente requisitos de software
- Diseñar un plan de configuración del software
- Diseñar un plan de auditoría informática

Contenidos Programáticos:

- Diseño de Interfaces Gráficas de Usuario Usables
- Métodos Formales de Especificación
- Gestión de la Configuración de Software
- Auditoría Informática

- Tópicos del Estado del Arte en la IS

Metodología:

Desarrollo de clases expositivas y análisis de casos de estudio por parte del profesor de los principales temas y/o contenidos del programa, y desarrollo de un proyecto de aplicación, investigación bibliográfica por parte de los alumnos, y discusión de lecturas de interés.

Evaluación:

Será definida por el académico y entregada a los alumnos al inicio de la asignatura.

Bibliografía Básica:

- Ian Sommerville. Software Engineering, Addison Wesley.
- Roger Pressman, Ingeniería de Software. Un enfoque práctico. McGraw Hill.

PROGRAMA ASIGNATURA	: ARQUITECTURA DE SOFTWARE
CARÁCTER	: OBLIGATORIO
PRE-REQUISITOS	: CONDICIONES DE INGRESO
SEMESTRE CURRICULAR	: PRIMERO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Descripción:

Este curso cubre aspectos de cómo grandes y complejos sistemas de software son organizados en términos de sus componentes e interrelaciones (arquitectura), las ventajas/desventajas de elección de una arquitectura determinada y de formalismos para la especificación de arquitecturas de software.

Objetivos Generales.

- Conocer el rol de la arquitectura de software en el proceso de desarrollo de software.
- Conocer e Identificar los distintos estilos Arquitecturales.
- Evaluar y Aplicar los estilos arquitecturales en el desarrollo de software.

Contenidos Programáticos.:

- Propósitos de la Arquitectura de Software
- Atributos de Calidad de la Arquitectura
- Estilos Arquitecturales
- Principios y Guías en el diseño arquitectural
- Arquitectura y procesos de desarrollo

Metodología:

Desarrollo de clases expositivas y análisis de casos de estudio por parte del profesor de los principales temas y/o contenidos del programa, y desarrollo de

trabajos prácticos y/o de investigación bibliográfica por parte de los alumnos, y/o discusión de lecturas de interés.

Evaluación:

Será definida por el académico y entregada a los alumnos al inicio de la asignatura.

Bibliografía Básica:

- Craig Larman, UML y Patrones: Una Introducción al Análisis y Diseño Orientado a Objetos y al Proceso Unificado, Pearson, Prentice Hall, 2003.
- Schmidt. Pattern Oriented Software Architecture II, John Wiley, 2000.
- Cristine Hofmeister et al. Applied Software Architecture, Addison Wesley, 1999.
- Mary Shaw et al. Software Architecture: Perspective on an Emerging Discipline, Addison Wesley, 1998.
- Len Bas et al. Software Architecture in Practice, Addison Wesley, 1998.

PROGRAMA ASIGNATURA	: ELECTIVO I
CARÁCTER	: OBLIGATORIO
PRE-REQUISITOS	: A DEFINIR SEGÚN CONTENIDO
SEMESTRE CURRICULAR	: PRIMERO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Objetivos Generales:

Los electivos pertenecen al currículo del programa profesionalizante y su objetivo es permitir al alumno profundizar en algunos tópicos particulares de la Ingeniería de Software.

Las asignaturas que el programa ofrecerá como electivo pueden ser algunas de las expuestas en el anexo A, o cualquiera otra que el comité de programa considere pertinente según los contenidos que trate, y el desarrollo tecnológico alcanzado por la ingeniería de software al momento de su dictación.

Contenidos Programáticos:

Depende de la Asignatura.

Metodología:

Depende de la Asignatura.

Evaluación:

Será definida por el académico y entregada a los alumnos al inicio de la asignatura.

Bibliografía Básica:

Depende de la Asignatura.

PROGRAMA ASIGNATURA : GESTIÓN DE PROCESO Y CALIDAD DE SOFTWARE

CARÁCTER	: OBLIGATORIO
PRE-REQUISITOS	: CONDICIONES DE INGRESO
SEMESTRE CURRICULAR	: SEGUNDO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Descripción:

Este curso cubre los conceptos y aspectos claves existentes en el proceso de software, administración de proceso de desarrollo, y tópicos relacionados con la administración de calidad del producto software (tales como métricas, herramientas y modelos de mejoramiento de calidad).

Objetivos Generales:

- Entender la importancia del proceso de desarrollo.
- Aplicar los conceptos de procesos en el desarrollo de software.
- Entender la importancia de la Calidad del producto software.

Contenidos Programáticos:

- Métricas y Estimación de Costos
- Planificación de Proyecto Software.
- Calidad de Software.
- Administración de Configuración.
- Modelos de Calidad.

Metodología:

Desarrollo de clases expositivas y análisis de casos de estudio por parte del profesor de los principales temas y/o contenidos del programa, y desarrollo de trabajos prácticos y/o de investigación bibliográfica por parte de los alumnos, y/o discusión de lecturas de interés.

Evaluación:

Será definida por el académico y entregada a los alumnos al inicio de la asignatura.

Bibliografía Básica:

- Ivar Jacobson. The Unified Software Development Process, Addison Wesley, 1999.
- Pankaj Jalote et al. CMM in Practice, Addison Wesley, 1999.
- Ian Sommerville. Software Engineering (quinta edición), Addison Wesley, 1998.
- Watts S. Humphrey. Introduction to the Personal Software Process, Addison Wesley, 1996.
- Practical Software Metrics for Project Management and Software Improvement, Prentice Hall, 1992.
- Watts S. Humphrey. Managing the Software Process. Addison Wesley, 1989.

PROGRAMA ASIGNATURA	: ELECTIVO II
CARÁCTER	: OBLIGATORIO
PRE-REQUISITOS	: A DEFINIR SEGÚN CONTENIDO
SEMESTRE CURRICULAR	: SEGUNDO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Objetivos Generales:

Los electivos pertenecen al currículo del programa profesionalizante y su objetivo es permitir al alumno profundizar en algunos tópicos particulares de la Ingeniería de Software.

Las asignaturas que el programa ofrecerá como electivo pueden ser algunas de las expuestas en el anexo A, o cualquiera otra que el comité de programa considere pertinente según los contenidos que trate, y el desarrollo tecnológico alcanzado por la ingeniería de software al momento de su dictación.

Contenidos Programáticos:

Depende de la Asignatura.

Metodología:

Depende de la Asignatura.

Evaluación:

Será definida por el académico y entregada a los alumnos al inicio de la asignatura.

Bibliografía Básica:

Depende de la Asignatura.

PROGRAMA ASIGNATURA	: ELECTIVO III
CARÁCTER	: OBLIGATORIO
PRE-REQUISITOS	: A DEFINIR SEGÚN CONTENIDO
SEMESTRE CURRICULAR	: SEGUNDO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Objetivos Generales:

Los electivos pertenecen al currículum del programa profesionalizante y su objetivo es permitir al alumno profundizar en algunos tópicos particulares de la Ingeniería de Software.

Las asignaturas que el programa ofrecerá como electivo pueden ser algunas de las expuestas en el anexo A, o cualquiera otra que el comité de programa considere pertinente según los contenidos que trate, y el desarrollo tecnológico alcanzado por la ingeniería de software al momento de su dictación.

Contenidos Programáticos:

Depende de la Asignatura.

Metodología:

Depende de la Asignatura.

Evaluación:

Será definida por el académico y entregada a los alumnos al inicio de la asignatura.

Bibliografía Básica:

Depende de la Asignatura.

PROGRAMA ASIGNATURA	: SEMINARIO DE PROYECTO DE GRADO
CARÁCTER	: OBLIGATORIO
PRE-REQUISITOS	: PRIMER SEMESTRE
SEMESTRE CURRICULAR	: SEGUNDO
HORAS PRESENCIALES	: 8 (0,8,0)
CRÉDITOS (SCT)	: 8

Descripción:

Este curso cubre el estudio de la metodología de investigación, formulación y revisión bibliográfica de un proyecto que cumpla con las exigencias del Comité Académico del Programa.

Objetivos Generales:

Conocer y aplicar técnicas de investigación.

Aplicar los conocimientos adquiridos, mediante la formulación de una solución a un problema real.

Contenidos Programáticos:

- Técnicas de investigación.
- Resolución de Problemas.
- Realización de Búsqueda Bibliográfica.
- Formulación de Proyectos.
- Estructura de Publicación y Presentación de Resultados.

Metodología:

Trabajo personal del alumno con exposición oral de resultados.

PROGRAMA ASIGNATURA	: ACTIVIDAD FINAL DE POSTGRADO
CARÁCTER	: OBLIGATORIO
PRE-REQUISITOS	: SEGUNDO SEMESTRE.
SEMESTRE CURRICULAR	: TERCERO
HORAS PRESENCIALES	: 10 (0,10,0)
CRÉDITOS (SCT)	: 30

Descripción:

Corresponde al conjunto de actividades y exigencias que deben cumplir los alumnos en la etapa final del programa para optar al grado de Magíster en Ingeniería de Software.

Objetivos Generales:

El alumno debe desarrollar un trabajo de integración de conocimientos o investigación bajo la supervisión directa de un profesor tutor. Este trabajo o investigación, debe ser de nivel y complejidad tal, que cumpla con el perfil de formación profesional del programa.

El desarrollo y evaluación de esta actividad debe ceñirse a las normas y procedimientos establecidos en el reglamento para la Presentación, Desarrollo y Evaluación de la Actividad Docente Terminal de los Programas de Magister de la EUIIS.

XI. EJECUCIÓN.-

Las normas y procedimientos para la orientación, organización y ejecución de las actividades tanto académicas como administrativas de este programa, estarán regidas por los siguientes documentos institucionales:

- Procedimiento de Admisión y Matrícula para Programas de Postgrado.
- Ordenanza de Conceptualización de Grados Académicos de la Universidad de Tarapacá.
- Reglamento de Docencia de Programas de Postgrados de la Universidad de Tarapacá.
- Reglamento para la Presentación, Desarrollo y Evaluación de la Actividad Docente Terminal de los Programas de Magister de la EUIIS.

XII. PROFESORES.-

12.1. PROFESORES PERMANENTES DEL PROGRAMA.

- **LUIS CÁCERES ALVAREZ**
Doctor en Ciencias de la Computación
- **RAÚL HERRERA ACUÑA**
Doctor of Philosophy mención Computer Science
- **RICARDO VALDIVIA PINTO**
Doctor en Ciencias de la Ingeniería
- **MARCO VILLALOBOS ABARCA**
Doctor en Lenguajes y Sistemas Informáticos e Ingeniería de Software

- **DIEGO ARACENA PIZARRO**

Magíster en Ingeniería Eléctrica área Ingeniería en Computación

- **HÉCTOR BECK FERNÁNDEZ**

Magíster en Ciencia mención Computación

12.2. PROFESORES COLABORADORES

- **HECTOR OSSANDON DIAZ**

Ingeniero Civil en Computación e Informática

- **IBAR RAMÍREZ VARAS**

Ingeniero Civil Químico

A N E X O A

PROGRAMA ASIGNATURA	: INGENIERIA DE REQUERIMIENTOS
CARÁCTER	: ELECTIVO
PRE-REQUISITOS	: INGENIERIA DE SOFTWARE
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Descripción:

Este curso cubre el proceso de Especificación de Requerimientos (obtención, Modelamiento, expresión y validación), la utilización de diversas técnicas y su comparación.

Objetivos:

Al término del curso el alumno deberá ser capaz de:

- Analizar un proceso de negocio y sus necesidades, y traducirlo a un conjunto de requerimientos de software.
- Aplicar métodos y técnicas disponibles a las tareas de especificación y análisis de requerimientos.

Contenidos Programáticos:

- Conceptos de requerimientos y de software.
- Fases en el proceso de obtención de requerimientos.
- Métodos, técnicas y herramientas actuales para la obtención de requerimientos.
- Prácticas industriales y Estándares.
- Administración de Requerimientos.

Metodología:

Desarrollo de clases expositivas y análisis de casos de estudio por parte del profesor en los principales temas del programa. Desarrollo de trabajos prácticos, de investigación bibliográfica por parte de los alumnos y discusión de lecturas de interés.

Bibliografía Básica:

- Ian Sommerville. Requirements Engineering: Process and Techniques, John Wiley, 1999.
- Dean Leffingwell et al. Managing Software Requirements: A Unified Approach, Addison Wesley, 1999.
- Ian Sommerville. Software Engineering (quinta edición), Addison Wesley, 1998.
- Roger S. Pressman. Ingeniería de Software: Un Enfoque Práctico (quinta edición), Mc Graw Hill, 1998.

PROGRAMA ASIGNATURA : METODOS FORMALES DE INGENIERIA DE SOFTWARE

CARÁCTER : ELECTIVO

PRE-REQUISITOS : A DEFINIR SEGÚN CONTENIDO

HORAS PRESENCIALES : 4 (4,0,0)

CRÉDITOS (SCT) : 8

Descripción:

Este curso cubre métodos y herramientas formales y su aplicación en las distintas fases del desarrollo (especificación, diseño y verificación de propiedades) de software crítico, el alumno deberá ser capaz de aplicarlos correctamente en el desarrollo de una aplicación práctica.

Objetivos Generales:

- Conocer distintos métodos y herramientas formales disponibles, y cómo ellos pueden ser usados en el desarrollo de software crítico.
- Aplicar métodos formales en la especificación de software crítico.

Contenidos Programáticos:

1. Introducción

- Motivación
- Propósitos de los MF
- Definición
- MF v/s IS
- ¿Por qué usar los MF?

2. Especificaciones Formales

- Definición
- Problemas con las especificaciones no formales
- Clasificación

3. Lenguajes de Especificación

- Redes de Petri
- Lenguaje Z
- OCL

4. Métodos Formales y la Ingeniería de Software

- El proceso de desarrollo de Software
- El proceso de desarrollo con Métodos Formales

Metodología:

Desarrollo de clases expositivas y análisis de casos de estudio por parte del profesor de los principales temas y/o contenidos del programa, y desarrollo de un proyecto de aplicación, investigación bibliográfica por parte de los alumnos, y discusión de lecturas de interés.

Bibliografía Básica:

- J. M. Spivey, (2001), "The Z Notation: a reference manual", Prentice Hall, <http://spivey.oriel.ox.ac.uk/~mike/zrm>
- UML 2.0 OCL Specification, OMG Adopted Specification, <http://www.omg.org/cgi-bin/doc?ptc/2003-10-14>
- Best, Eike, Devillers, Raymond Et Koutny, Maciej, "Petri Net algebra" Berlin. Springer-Verlag, 2001.
- Roger Pressman, Ingeniería del Software. Un enfoque práctico. McGraw Hill
- Villalobos, M, (2011), "Apuntes de Clases".

PROGRAMA ASIGNATURA : TOPICOS AVANZADOS EN BASES DE DATOS

CARÁCTER : ELECTIVO

PRE-REQUISITOS : CONDICIONES DE INGRESO

HORAS PRESENCIALES : 4 (4,0,0)

CRÉDITOS (SCT) : 8

Descripción:

El curso de Tópicos Avanzados en Bases de Datos pretende cubrir aquellos aspectos relacionados con el tópico de Bases de Datos que no son cubiertos en el curso regular en el área. Así, se subentiende que el alumno es capaz de diseñar una base de datos siguiendo alguna metodología basada en modelos, reconoce y utiliza los principales conceptos asociados al modelo relacional y ha trabajado con las principales características de un sistema administrador de bases de datos relacional.

El curso de Tópicos Avanzados en Bases de Datos se hace cargo de temas asociados a modelos emergentes en bases de datos como, bases de datos activas,, espaciales, temporales, objeto-relacionales, orientadas a objetos y NoSQL entre otras.

Objetivos Generales:

Al finalizar este curso, el alumno deberá:

- Diseñar bases de datos utilizando modelos emergentes.
- Implementar bases de datos utilizando conceptos asociados a las nuevas tecnologías en bases de datos.
- Realizar opiniones fundadas sobre las nuevas tecnologías en el área de bases de datos.

Contenidos Programáticos:

1. Extensiones al Modelo Relacional

- Bases de Datos Activas
- Bases de Datos Espaciales
- Bases de Datos Temporales
- Bases de Datos XML

2. Orientación a Objetos en Bases de Datos

- Elementos de Orientación a Objetos
- Bases de Datos Objeto Relacionales
- Bases de Datos Orientadas a Objetos
- Mapeo Objeto Relacional (ORM)

3. Bases de Datos para el problema 4V

- El problema Big Data
- El movimiento NoSQL
- Bases de Datos Orientadas a Documentos
- Bases de Datos Orientadas a Columna
- Bases de Datos de Clave Valor
- Bases de Datos en Grafo

Metodología:

- Se trabajará con equipos de estudiantes que desarrollarán un proyecto de implementación de un modelo de datos utilizando tres propuestas en tecnologías de bases de datos.
- El equipo de estudiantes realizará una propuesta de modelo de datos, el que deberá ser acorde a la tecnología a emplear.

- El equipo de estudiantes deberán implementar su modelo utilizando la tecnología seleccionada, para esto deberá investigar la tecnología, comprenderla y aplicarla al modelo propuesto.
- El equipo deberá concluir la aplicación de cada tecnología con una presentación en la que, además de un informe, deberá mostrar los resultados de su implementación.
- Se incorporarán mecanismos de evaluación que consideren la interdependencia positiva y la exigibilidad personal en el equipo.
- El profesor cumplirá un rol de facilitador y guía de cada actividad, apoyando el desarrollo de la misma.

Bibliografía Básica:

- Fundamentos de Sistemas de Bases de Datos, Quinta Edición
RamezElMasri, ShamkantNavathe, Pearson Educación 2007
Clasificación Sistema de Bibliotecas UTA: S005.74 E161f5E 2007
- Tecnología y Diseño de Bases de Datos
Mario Piatinni, Esperanza Marcos, Belén Bela,
Ra-Ma,2006
- Database Design for Smarties. Using UML for Data Modeling
Robert J. Muller,
Morgan Kaufmann, 1999
Clasificación Sistema de Bibliotecas UTA: 005.74 M912d 1999
- Getting Started with NoSQL
Gaurav Vaish
PACKT Publishing, 2013

PROGRAMA ASIGNATURA	: INTELIGENCIA DE NEGOCIOS
CARÁCTER	: ELECTIVO
PRE-REQUISITOS	: A DEFINIR SEGÚN CONTENIDO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Descripción:

Este curso se orienta a que el estudiante sea capaz de desarrollar una solución de inteligencia de Negocios en la organización, diseñando los procesos necesarios y utilizando las herramientas tecnológicas adecuadas.

Objetivos:

- Comprender la importancia del uso de tecnologías de Inteligencia de Negocios y su aplicabilidad en el soporte de los procesos de toma de decisiones de una organización.
- Diseñar e implementar un datawarehouse como una base de datos orientada al procesamiento analítico.
- Utilizar métodos y herramientas OLAP con una orientación al análisis de información estratégica y a la de toma de decisiones.

Contenidos Programáticos:

1. Introducción a la Inteligencia de Negocios
 - Definiciones Básicas
 - Sistemas Operacionales vs. Sistemas para la Toma de Decisiones
 - OLTP vs OLAP
 - Datawarehouse y Datamarts

2. Diseño Multidimensional

- Cubos Multidimensionales
- MOLAP VS ROLAP
- Esquemas Copo de Nieve vs Esquema Estrella
- Hechos y Dimensiones

3. Tecnologías para la Inteligencia de Negocios

- Herramientas ETL
- Herramientas OLAP
- Herramientas para Reportes
- Plataformas BI

Metodología:

- Clases expositivas por parte del profesor.
- Resolución de problemas individual y colaborativamente por parte del alumno.
- Talleres en un ambiente cliente/servidor.

Bibliografía Básica:

- Building the data warehouse, William H. Inmon, Wiley, 2002.
Clasificación Sistema de Bibliotecas UTA: 658.40380285574 In6b3 2002.
- The data warehouse toolkit: the complete guide to dimensional modeling, Ralph Kimball, Wiley, 2002.
Clasificación Sistema de Bibliotecas UTA: 658.40380285574 K566d2 2002.
- Data warehousing fundamentals: A comprehensive guide for IT professionals, PaulrajPonniah, Wiley, 2001.
Clasificación Sistema de Bibliotecas UTA:658.40380285574 P774d 2001.
- Pentaho Solutions, Business Intelligence and Data Warehousing with Pentaho and MySQL.
RolandBouman, Jos van Dongen, Wiley, 2009.
Clasificación Sistema de Bibliotecas UTA: 005.7565 B665p 2009.

PROGRAMA ASIGNATURA	: PROGRAMACION PARALELA MODERNA
CARÁCTER	: ELECTIVO
PRE-REQUISITOS	: A DEFINIR SEGÚN CONTENIDO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Descripción:

La asignatura entrega los conocimientos recientes, para poder solucionar problemas que aprovechen la potencia paralela con la tecnología actual, ya que los datos cada vez son más masivo o “Big Data”, en donde esta potencia computacional permite extraer conocimiento a partir de la información existentes en las bases de datos históricas que las instituciones o empresas poseen y lograrlo en tiempos apropiados.

Objetivo General:

El objetivo de esta asignatura es entregar conocimientos recientes de estándares de sistemas paralelos en Cluster, tales como OpenMP, OpenCL, CUDA, en especial los sistemas formados por clusters de computadores conectados en red y CPU`s en conjunto con GPU`s. Introduciendo al alumno en el mundo de las arquitecturas y en la programación de aplicaciones paralelas de Alto Desempeño Computacional (HPC).

Contenidos Programáticos:

- Arquitecturas Paralelas
- Modelos de Rendimiento
- Programación basada en paso de mensajes
- Programación basada en CPU`s y GPU`s

- Problemas altamente paralelizables
- Computación segmentada
- Sincronización de procesos
- Balance de Carga
- Algoritmos de detección de término
- Protocolos de término y tiempo real
- Transacciones concurrentes y paralelas
- Casos de Estudios

Metodología:

- Clases expositivas iniciales
- Ejercicios prácticos empleando metodologías activas
- Desarrollo de Proyecto en grupos
- Lecturas de papers

Bibliografía Básica:

- Barry Wilkinson y Michael Allen, "Parallel Programming: Using Networked Workstations and Parallel Computers, Prentice Hall, New Jersey, 2000.
- G.F. Coulouris, J. Dollimore y T. Kindbert: Sistemas Distribuidos: Conceptos y Diseño. 3a. Ed.. Addison Wesley. 2001.
- IBM, "Introduction to Grid Computing with Globus" International Technical Support Organization, RedBook December 2002.
- David B. Kirk and Wen-mei W. Hwu, "Programming Massively Parallel Processors: A Hands-on Approach", Applications of GPU Computing Series, Feb 5, 2010.

- Cameron Hughes and Tracey Hughes, “Professional Multicore Programming: Design and Implementation for C++ Developers”, Wrox Programmer to Programmer, Sep 9, 2008.
- Aaftab Munshi, Benedict R. Gaster, Timothy G. Mattson, James Fung and Dan Ginsburg, “OpenCL Programming Guide”, Addison Wesley, Stanford University, Marzo 2012.
- Open MP, Tutorials en intranet (autorizado por el autor).
- Peter S. Pacheco, Parallel Programming with MPI, Morgan Kaufman Publishers, 1997.
- REUNA “Mallas (GRIDS)”, Edición especial, 2004.
- IEEETFCC, “Grupo de tarea de computación paralela” <http://www.ieeetfcc.org>

PROGRAMA ASIGNATURA	: VISION COMPUTACIONAL
CARÁCTER	: ELECTIVO
PRE-REQUISITOS	: A DEFINIR SEGÚN CONTENIDO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Descripción:

Este curso entrega los conocimientos recientes para incursionar en el procesamiento de imágenes temprano y lograr extraer información a través de técnicas que permitan lograr detección, seguimiento, reconocimiento y aspectos de inteligencias a ser aplicados en Ambientes Robóticos, Control Moderno, Realidad Virtual o Aumentada, Minería de imágenes, entre otras.

Objetivos Generales:

Lograr competencias en:

Desarrollo de aplicaciones con Visión Computacional, en el ámbito que sea de interés, tales como, mezcla de realidades, realidad aumentada, vigilancia, máquina de visión, robótica, entre otras.

Contenidos Programáticos:

- Introducción a la Computación Gráfica y Sistemas de Información gráfica
- Introducción a la Visión Computacional
- Aspectos de Diseño de Software en Visión Computacional
- Procesamiento de imágenes
- Ruido y Filtros
- Detección de rasgos y segmentación
- Problema de Correspondencia y matching
- Calibración de Cámaras

- Geometría Epipolar
- Reconstrucción 3D
- Reconocimiento de patrones
- Clasificadores
- Estudio de Casos

Metodología:

- Clases expositivas iniciales
- Ejercicios prácticos empleando metodologías activas
- Desarrollo de Proyectos en grupos
- Lecturas de papers

Bibliografía Básica:

- D. Aracena, C. Tozzi, (2001) “Realidad Aumentada Integrando Visión Computacional” Tutorial Universidad San Agustín. Arequipa Perú. En Biblioteca UTA.
- An Invitation to 3-D Visión: From Images to Geometric Models, Yi Ma, S. Soatto, J. Kosecka, and S. Sastry, Springer-Verlag, November 2006.
- Múltiple View Geometry in Computer Vision, R. Hartley and A. Zisserman, 2da. Edition, Cambridge Press, 2004.
- R. González, R. Woods, S. Eddins (2004) “Digital Image Processing: Using Matlab” Prentice Hall (pearson Education). New York.

PROGRAMA ASIGNATURA	: DATA MINING EN LA WEB
CARÁCTER	: ELECTIVO
PRE-REQUISITOS	: A DEFINIR SEGÚN CONTENIDO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Descripción:

El curso permitirá identificar las técnicas actuales de Minería de Datos, aplicar correctamente las técnicas de Minería de Datos en aplicaciones en la web, elaborar conclusiones al extraer datos y utilizar herramientas de extracción de datos para el uso de las técnicas de Data Mining.

Objetivos Generales:

- Determinar fortalezas y limitaciones de las técnicas de Minería de Datos.
- Identificar y aplicar la técnica apropiada en la aplicación en estudio, especialmente en la web.

Contenidos Programáticos:

- Introducción
- Reglas de asociación y patrones secuenciales
- Aprendizaje supervisado
- Aprendizaje no supervisado
- Aprendizaje supervisado parcialmente
- Recuperación de Información y búsqueda en la Web
- Análisis de redes sociales
- Web Crawling

Metodología:

El alumno deberá complementar su asistencia a las clases expositivas del profesor con las actividades:

1. Trabajo práctico
 - Exposición oral del trabajo realizado
 - Confección de informe escrito
2. Resolución de problemas
 - Problemas presentados por el profesor en el aula
3. Búsqueda de información
 - En la web sobre los temas que se tratarán
4. Lectura de artículos científicos
 - Seleccionados por el profesor y relevantes para el desarrollo del curso

Bibliografía Básica:

- Baeza-Yates, R. and Ribeiro-Neto, Berthier; Modern Information Retrieval, Second edition; ACM Press Books; 2011.
- Baeza-Yates, R. y Gionis A., Yahoo! Research, Barcelona, Spain & Santiago, Chile (slides en la página web del curso).
- Hernández Orallo, J. Ramírez Quintana, M. J. Ferri Ramírez, C.; Introducción a la Minería de Datos; Pearson Educación S.A., Madrid, 2004.
- Liu, B.; Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data; Second Edition; Springer Berlin 2011.
- Nettleton, F. David; Data Mining of Social Networks represented as graphs; Computer Science Review (2013), doi:101016/j.cosver.2012.001.
- Patel, Nitin. 15.062 Data Mining, Spring 2003. (MIT OpenCourseWare:

- Massachusetts Institute of Technology), <http://ocw.mit.edu/courses/sloan-school-of-management/15-062-data-mining-spring-2003> (Accessed 15 Jan, 2014). License: Creative Commons BY-NC –SA.

PROGRAMA ASIGNATURA	: RECUPERACION DE INFORMACION
CARÁCTER	: ELECTIVO
PRE-REQUISITOS	: A DEFINIR SEGÚN CONTENIDO
HORAS PRESENCIALES	: 4 (4,0,0)
CRÉDITOS (SCT)	: 8

Descripción:

El curso describirá cuál es el problema actual de recuperación de la información en medios digitales, en especial en la Web. Permitirá distinguir los diversos modelos y variables de recuperación de información. Se describirán las técnicas que permiten evaluar la recuperación de información dada una consulta, los diversos formatos de grabación de información en medios digitales, los tipos y propiedades de una consulta en diversos tipos de textos digitales.

Objetivos Generales:

Al final de la asignatura el estudiante deberá:

Describir cómo encontrar y evaluar información en grandes bases de datos textuales, semi o no estructuradas, y sus aplicaciones en particular la Web, con el propósito de realizar la recuperación de textos, imágenes, sonido o datos de otras características, de manera pertinente y relevante.

Contenidos Programáticos:

- Introducción
- Modelación de Sistemas de Recuperación e Información
- Evaluación de la recuperación
- Relevancia de la retroalimentación y expansión de la consulta
- Documentos: Lenguajes y propiedades
- Consultas: Lenguajes y propiedades
- Clasificación de textos
- Indexación y búsqueda

Metodología:

- Sesiones teóricas de clases
- Búsqueda de información en la web
- Realización de un proyecto, trabajo grupal
- Lectura de artículos en revistas especializadas

Bibliografía Básica:

- Manning, Christopher D; Raghavan, Prabhakar; Schtze, Hinrich: Introduction to Information Retrieval. Cambridge University Press, 2008. Disponible en web: <http://nlp.stanford.edu/IR-book/pdf>
- [Distintos Autores]: Posicionamiento Web. Revista El Profesional de la Información, Vol. 14, números 1 y 2. 2005.
- Moens Marie-Francine, Information Extraction: algorithms and prospects Información , Vol. 14, números 1y 2. 2005.
- Antoniou, Grigoris & Van Harmelen, Frank. A semantic Web Primer. MIT, 2004.
- Ian H. Witten, Alistair Moffat and Timothy C. Bell. Managing Gigabyte: compressing and indexing documents. Morgan Kauffman, 2da ed., 1999.
- Dale R. Handbook of Natural Language Processing. New York, 2000.
- Nadeau D. and Sekine S., A survey of named entity recognition and classification. Lingüística e Investigaciones, vol. 30 N°1, 2007.