



## MODALIDAD ACADÉMICA

<b>Asignatura</b>	Ingeniería de Software de Fuentes Abiertas/Libre	
<b>Carrera</b>	INGENIERÍA EN SISTEMAS DE INFORMACIÓN	
<b>Ciclo Lectivo</b>	2014	
<b>Vigencia del programa</b>	Desde el ciclo lectivo 2014	
<b>Plan</b>	2008	
<b>Nivel</b>	<input type="checkbox"/> 3er. Nivel <input type="checkbox"/> 4to. Nivel <input type="checkbox"/> 5to. Nivel	
<b>Coordinador de Cátedra</b>	Ricardo Medel	
<b>Área de Conocimiento</b>	<input type="checkbox"/> Programación <input type="checkbox"/> Computación <input type="checkbox"/> Sistemas de Información <input type="checkbox"/> Gestión Ingenieril <input type="checkbox"/> Modelos <input type="checkbox"/> Complementaria	
<b>Carga horaria semanal</b>	6 HORAS SEMANALES	
<b>Anual/ cuatrimestral</b>	Cuatrimestral - (2° cuatrimestre <i>de X año</i> )	
<b>Modalidad de Dictado</b>	CURSOS PRESENCIAL	
<b>Correlativas para cursarla</b>	Regulares	Aprobadas
	• -	• -
<b>Correlativas para rendirla</b>	Regulares	Aprobadas
	• -	• -
<b>Fundamentación y Relación con el Perfil Profesional</b>	• ¿?	
<b>Objetivos de la Asignatura</b>	<u>Generales</u>	
	<ul style="list-style-type: none"><li>Entender los conceptos fundamentales, historia y actualidad del software de fuentes abiertas/libre.</li><li>Comprender los factores que afectan la gestión de proyectos de software de fuentes abiertas/libre y cómo enfrentarlos.</li><li>Adquirir los conocimientos necesarios para participar de un proyecto de software de fuentes abiertas/libre.</li></ul>	
	<u>Específicos</u>	
	<ul style="list-style-type: none"><li>Tener los conocimientos necesarios que permitan evaluar la conveniencia de utilizar software de fuentes abiertas/libre en la práctica profesional.</li><li>Saber utilizar herramientas conceptuales y de software que permiten participar en proyectos de desarrollo de software de fuentes abiertas/libre.</li><li>Entender los conceptos de calidad e integración</li></ul>	



	<p>continúa aplicados al desarrollo de software de fuentes abiertas/libre y utilizar herramientas apropiadas.</p> <ul style="list-style-type: none"><li>• Obtener la capacidad de analizar cómo el software afecta las relaciones sociales y económicas en la sociedad.</li><li>• Poder definir un modelo de negocios basado en software de fuentes abiertas/libre y el modelo cooperativo de creación de valor.</li></ul>
<p style="text-align: center;"><b><u>Programa Analítico</u></b></p> <p><b>UNIDAD N° 1: Conceptos básicos e historia del Software de fuentes abiertas/libre</b></p> <p><b>OBJETIVOS ESPECÍFICOS:</b> Desarrollar en el estudiante una comprensión de los conceptos fundamentales de software de fuentes abiertas/libre, así como sus diferentes definiciones, su historia y el estado actual del software de fuentes abiertas/libre. Brindar al estudiante herramientas que le permitan comprender los aspectos legales a considerar cuando se hace uso del software de fuentes abiertas/libre en el contexto de su profesión.</p> <p><b>CONTENIDOS:</b> Historia del software. Surgimiento del software privativo. Proyecto GNU: el nacimiento del software libre. Conceptos y definiciones de software libre. La Iniciativa Open Source (OSI): la conceptualización del software de fuentes abiertas. Historia del software de fuentes abiertas/libre en Latinoamérica. Licencias de software: privativas, permisivas, recíprocas totales, recíprocas parciales. Legislación sobre software de fuentes abiertas/libre.</p> <p><b>BIBLIOGRAFÍA:</b></p> <ul style="list-style-type: none"><li>- González Barahona, Jesús, Seoane Pascual, Joaquín, Robles, Gregorio. “Introducción al software libre”, Fundació per a la Universitat Oberta de Catalunya, 2003. ISBN: 84-9788-028-5</li><li>- Williams, Sam. “Free as in Freedom: Richard Stallman's Crusade for Free Software”, O'Reilly, 2002.</li><li>- Raymond, Eric S., “La catedral y el bazar”, O'Reilly, 1997.</li><li>- Gomez Gomez, Víctor Manuel. “Aspectos Históricos del Software Libre en América Latina”, 2005.</li><li>- St. Laurent, Andrew M. “Understanding Open Source and Free Software Licensing”, O'Reilly, 2004, ISBN: 0-596-00581-4.</li><li>- Wheeler, David A. “Make Your Open Source Software GPL-Compatible. Or Else.”, 2014, <a href="http://www.dwheeler.com/essays/gpl-compatible.html">http://www.dwheeler.com/essays/gpl-compatible.html</a></li></ul> <p><b>EVALUACIÓN:</b> Este temario se incluye en los ejercicios prácticos a entregar y en el examen parcial integrador.</p>	
<p><b>UNIDAD N°2: Análisis y gestión de proyectos de Software Libre</b></p> <p><b>OBJETIVOS ESPECÍFICOS:</b> Desarrollar en el estudiante una comprensión de los factores que afectan la gestión de proyectos de software de fuentes abiertas/libre y cómo enfrentarlos.</p>	



Brindar al estudiante herramientas conceptuales y de software que le permitan participar o manejar una comunidad enfocada al desarrollo de software de fuentes abiertas/libre.

#### **CONTENIDOS:**

Tipos de comunicación en proyectos de software de fuentes abiertas/libre, ventajas y limitaciones (email, IRC, foros, blogs). Sistemas de control de versión y su impacto sobre el modelo de contribuciones (SVN, Git, Mercurial, Bazaar). Sistemas de seguimiento de Bugs (Bugzilla, Redmine). Distintas formas de participación en comunidades de software de fuentes abiertas/libre. Sistemas de reputación en proyectos de software de fuentes abiertas/libre y meritocracia.

#### **BIBLIOGRAFÍA:**

- Fogel, Karl. “Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre”, O’Reilly, 2007.
- Raymond, Eric S. “La catedral y el bazar”, O’Reilly, 1997.
- Hernández Gómez, Lorena. “Herramientas software para el trabajo científico colaborativo”, Trabajo de Fin de Máster, Universidad de Salamanca, 2011.
- Collins-Sussman, Ben, Fitzpatrick, Brian W., Pilato, C. Michael. “Control de versiones con Subversion”, O’Reilly, 2004.
- Loeliger, Jon, McCullough, Matthew. “Version Control with Git, 2nd Edition”, O’Reilly, 2012.
- O’Sullivan, Bryan. “Mercurial: The Definitive Guide”, O’Reilly, 2009.
- Gyerik, Janos. “Bazaar Version Control”, Packt Publishing, 2013.
- Lesyuk, Andriy. “Mastering Redmine”, Packt Publishing, 2013.

#### **EVALUACIÓN:**

Este temario se incluye en los ejercicios prácticos a entregar y en el examen parcial integrador. Asimismo, su comprensión será evaluada por la participación del estudiante en un proyecto de software de fuentes abiertas/libre real.

### **UNIDAD N° 3: Diseño e implementación de Software Libre**

#### **OBJETIVOS ESPECÍFICOS:**

Desarrollar en el estudiante los conceptos básicos requeridos para analizar y valorar los aportes de codificación en una comunidad de software de fuentes abiertas/libre.

Brindar al estudiante los conocimientos necesarios para aportar código en un proyecto de software libre.

#### **CONTENIDOS:**

El código auto-documentado. Guías de estilo. Filosofía Unix de Diseño. *Refactoring*. Patrones de diseño. Participación como desarrollador en un proyecto de software de fuentes abiertas/libre.

#### **BIBLIOGRAFÍA:**

- Fogel, Karl. “Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre”, O’Reilly, 2007.
- Martin, Robert C. “Clean Code”, 2008. ISBN-10: 0132350882
- Fowler, Martin, Beck, Kent, Brant, John, Opdyke, William, Roberts, Don. “Refactoring: Improving the Design of Existing Code”, 1999. ISBN-10: 0201485672
- Gamma, Erich, Helm, Richard, Johnson, Ralph, Vlissides, John. “Design patterns : elements of reusable object-oriented software”, 1994. ISBN-10: 0201633612
- Sokol, Francisco Zigmund, Aniche, Mauricio Finavaro, Gerosa, Marco Aurélio. “Does the Act of Refactoring Really Make Code Simpler? A Preliminary Study”, Workshop Brasileiro de Métodos Ágeis, 2013.



#### **EVALUACIÓN:**

Este temario se incluye en los ejercicios prácticos a entregar y en el examen parcial integrador. Asimismo, su comprensión será evaluada por la participación del estudiante en un proyecto de software de fuentes abiertas/libre real.

#### **UNIDAD N° 4: Mantenimiento y calidad en proyectos de Software Libre**

##### **OBJETIVOS ESPECÍFICOS:**

Desarrollar en el estudiante los conceptos de calidad e integración continua aplicados al desarrollo de software de fuentes abiertas/libre.

Brindar al estudiante conocimiento práctico sobre las herramientas de automatización de las tareas de calidad en proyectos de software de fuentes abiertas/libre.

##### **CONTENIDOS:**

Recepción y revisión de *patches*. Tests automatizados. *Triaging* de bugs. Herramientas de *build* e integración continua (Jenkins, Hudson).

##### **BIBLIOGRAFÍA:**

- Burns, Ed, Prakash, Winston. “Hudson Continuous Integration in Practice”, 2013. IS “BN-10: 0071804285
- Berg, Alan. “Jenkins Continuous Integration Cookbook”, Packt Publishing, 2012.
- Smart, John Ferguson. “Jenkins: The Definitive Guide”, 2011. ISBN-10: 1449305350
- Nelson-Smith, Stephen. “Test-driven Infrastructure with Chef, 2nd Edition”, O’Reilly, 2013.
- Black, Rex. “Managing the Testing Process, 3rd Edition”, Wiley, 2011.

#### **EVALUACIÓN:**

Este temario se incluye en los ejercicios prácticos a entregar y en el examen parcial integrador. Asimismo, su comprensión será evaluada por la participación del estudiante en un proyecto de software de fuentes abiertas/libre real.

#### **UNIDAD N° 5: Aspectos sociales del Software Libre**

##### **OBJETIVOS ESPECÍFICOS:**

Desarrollar en el estudiante la capacidad de analizar cómo el software afecta las relaciones sociales y económicas en la sociedad.

Brindar al estudiante herramientas que le permitan tomar decisiones sobre el uso adecuado del software de fuentes abiertas/libre en el contexto de su profesión.

##### **CONTENIDOS:**

El software de fuentes abiertas/libre como movimiento social. Software de fuentes abiertas/libre en la educación. Software de fuentes abiertas/libre en el estado. Modelos de negocios tradicionales y basados en software de fuentes abiertas/libre. Planes de negocios y *startups* basadas en software de fuentes abiertas/libre. Cooperativismo en el mercado del software.

##### **BIBLIOGRAFÍA:**

- D’Elia Branco, Marcelo. “El Software Libre y sus perspectivas para el desarrollo en América Latina y el Caribe”, <http://www.bellanet.org>.
- Rosa, Fernando da; Heinz, Federico. “Guía práctica sobre software libre: su selección y aplicación local en América Latina y el Caribe”, UNESCO, 2007. ISBN: 92-9089-103-3



- Stallman, Richard M. "Free Software, Free Society: Selected Essays of Richard M. Stallman", Free Software Foundation, 200
- Zúñiga, Lena. "Voces libres de los campos digitales: Una investigación social sobre el Software Libre en América Latina y el Caribe", 2006. ISBN: 9977-12-888-X
- Salas, Margarita. "Género y software libre en América Latina: Un estudio de caso", <http://www.sulabatsu.com/voces/Documentos/genero.pdf>
- Wheeler, David A. "Open Source Software (OSS or FLOSS) and the U.S. Department of Defense (DoD)", 2009. <http://www.dwheeler.com/essays/dod-oss.pdf>

#### **EVALUACIÓN:**

Este temario se incluye en los ejercicios prácticos a entregar y en el informe sobre la participación del estudiante en un proyecto de software de fuentes abiertas/libre real.

<b>Metodología de enseñanza y aprendizaje</b>	<p>Clases teóricas</p> <p>Clases que se desarrollarán en forma expositiva con el objeto de abordar y desarrollar la temática específica vinculada a los fundamentos de la asignatura. Puede invitarse a personalidades reconocidas en la comunidad de software de fuentes abiertas/libre como expositores de temáticas específicas.</p> <p>Clases teórico-prácticas</p> <p>Estas clases permitirán articular aspectos teóricos de la asignatura con actividades prácticas relacionadas con la temática abordada. Con este tipo de clases se buscará interrelacionar los fundamentos teóricos con las experiencias prácticas en el desarrollo y aplicación de software de fuentes abiertas/libre.</p> <p>Proyecto de Aplicación</p> <p>El alumno participará activamente en un proyecto de software de fuentes abiertas/libre ya existente a su elección. Este tipo de actividad está orientada a ejercitar en un ambiente real los temas teóricos y prácticos abordados en la asignatura.</p>
<b>Sistema de evaluación</b>	<p>La evaluación de los alumnos se compone de los siguientes tres componentes mayores, sumados a su asistencia y participación activa en las clases.</p> <ol style="list-style-type: none"><li>1. Ejercicios prácticos de realización obligatoria, con entrega pautada.</li><li>2. Examen parcial integrador.</li><li>3. Proyecto de aplicación (participación en proyecto de software libre).</li></ol> <p>Además la materia cuenta con un examen de recuperación en el caso de que el alumno no alcanzara el mínimo esperado en alguna de las instancias antes mencionadas.</p> <p>El porcentaje de aprobación de la materia es 80% para cada uno los tres componentes mencionados anteriormente.</p>
<b>Condiciones de regularidad</b>	<p>Sistema de Promoción total: Todo aquel estudiante que cumpla con el sistema de evaluación promocionará la asignatura completamente.</p>
<b>Modalidad de examen final</b>	<p>La regularidad de la materia implica la promoción total de la misma por lo cual no está planteado un examen final.</p>
<b>Actividades en laboratorio</b>	<p>Las actividades prácticas deberán ser realizadas asistidas por computadoras con diverso software libre, según el tema a tratar.</p>



<b>Horas/año totales de la asignatura</b>	90										
<b>Cantidad de horas prácticas totales</b>	50										
<b>Cantidad de horas teóricas totales</b>	40										
<b>Tipo de formación práctica</b> (marque la que corresponde si es asignatura curricular -no electiva-)	<input type="checkbox"/> Formación experimental <input type="checkbox"/> Resolución de problemas de ingeniería <input checked="" type="checkbox"/> Actividades de proyecto y diseño <input type="checkbox"/> Prácticas supervisadas en los sectores productivos y /o de servicios										
<b>Cantidad de horas afectadas a la formación práctica indicada en el punto anterior</b>	35 Resolución de problemas de Ingeniería.										
<b>Descripción de los prácticos</b>	<p>Las actividades prácticas semanales constarán de una guía de ejercicios que el alumno debe realizar solo o en grupo, según se establezca previamente. Algunos de estos ejercicios serán de entrega obligatoria, y aportarán a la nota del componente N° 1 del sistema de evaluación.</p> <p>Las actividades prácticas realizadas en el marco de un proyecto existente de software libre a elección del alumno serán pautadas previamente con el profesor, quien a su vez evaluará su ejecución en base a un informe del alumno (componente N° 3 del sistema de evaluación).</p>										
<b>Criterios de evaluación de los prácticos</b>	<p>Se evalúa la calidad de presentación de las resoluciones y la capacidad analítica del alumno o grupo para arribar a las conclusiones.</p> <p>Algunos de los conceptos evaluados como parte de las actividades descriptas anteriormente son:</p> <ul style="list-style-type: none"><li>Precisión Conceptual</li><li>Capacidad de análisis y síntesis</li><li>Criterios para transferir los conocimientos adquiridos a situaciones prácticas</li><li>Cumplimiento de los requerimientos de la actividad</li><li>El lenguaje técnico utilizado sea el adecuado a la actividad</li><li>La creatividad de la presentación de la actividad</li></ul>										
<b>Descripción de la presentación de los prácticos</b>	Los trabajos prácticos serán entregados en formato digital al docente. No existe un formato predefinido de presentación ya que la presentación del trabajo forma parte de los criterios de evaluación de los mismos.										
<b>Cronograma de actividades de la asignatura,</b> incluyendo semanas previstas para cada unidad	<table border="1"><thead><tr><th>Semana N°</th><th>Unidad N°</th><th>Horas Teóricas</th><th>Horas Prácticas</th><th>Contenido</th></tr></thead><tbody><tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr></tbody></table>	Semana N°	Unidad N°	Horas Teóricas	Horas Prácticas	Contenido					
Semana N°	Unidad N°	Horas Teóricas	Horas Prácticas	Contenido							



	1	1	3	3	Historia del software. Surgimiento del software privativo. Proyecto GNU: el nacimiento del software libre. Conceptos y definiciones de software libre. La Iniciativa Open Source (OSI): conceptualización del software de fuentes abiertas. Historia del software de fuentes abiertas/libre en Latinoamérica.
	2	1	3	3	Licencias de software: privativas, permisivas, recíprocas totales, recíprocas parciales. Legislación sobre software de fuentes abiertas/libre.
	3	2	3	3	Distintas formas de participación en comunidades de software de fuentes abiertas/libre. Sistemas de reputación en proyectos de software de fuentes abiertas/libre y meritocracia. Sistemas de control de versión y su impacto sobre el modelo de contribuciones (SVN, Git, Mercurial, Bazaar).
	4	2	3	3	Tipos de comunicación en proyectos de software de fuentes abiertas/libre, ventajas y limitaciones (email, IRC, foros, blogs). Sistemas de seguimiento de Bugs (Bugzilla, Redmine).
	5	3	3	3	El código auto-documentado. Guías de estilo. Filosofía Unix de Diseño.
	6	3	3	3	<i>Refactoring</i> . Patrones de diseño.
	7	-	1	5	Selección del proyecto de software de fuentes abiertas/libre donde participar.
	8	4	3	3	Recepción y revisión de <i>patches</i> .
	9	4	3	3	Tests automatizados. <i>Triaging</i> de bugs..
	10	4	3	3	Herramientas de <i>build</i> e integración continua
	11	-	3	3	Semana del Parcial Integrador (Consulta y Examen)
	12	-	1	5	Participación en el proyecto de software de fuentes abiertas/libre seleccionado.
	13	5	4	2	El software de fuentes abiertas/libre como movimiento social. Software de fuentes abiertas/libre en la educación. Software de fuentes abiertas/libre en el estado.



	14	5	4	2	Modelos de negocios tradicionales y basados en software de fuentes abiertas/libre. Planes de negocios y <i>startups</i> basadas en software de fuentes abiertas/libre. Cooperativismo en el mercado del software.
	15	-	0	6	Presentación de resultados de participación en proyectos.
	<b>Crédito Horario Total</b> <b>90</b>		<b>40</b>	<b>50</b>	
	<p><b>Nota:</b> La distribución de horas teórico-práctica puede variar ya que la temática y enfoque aplicado de la materia hace que ambos contenidos sean altamente solapados en algunas de las unidades temáticas.</p>				
<b>Descripción de metodología propuesta de consultas y cronograma de consultas</b>	<p>El estudiante podrá contactar al coordinador de la cátedra de la siguiente manera: De lunes a viernes de 09:00 a 21:00 Hs. Al TE 0351-153065199 o con la casilla de mail: <a href="mailto:ricardo.h.medel@gmail.com">ricardo.h.medel@gmail.com</a> o en la oficina 4 del Edificio Maders de lunes a viernes de 18:00 a 20:00.</p>				
<b>Plan de integración con otras asignaturas</b>	<p>Esta asignatura requiere de los conocimientos desarrollados en las siguientes asignaturas:</p> <ul style="list-style-type: none"> <li>-</li> </ul>				
<b>Bibliografía Obligatoria</b>	<ul style="list-style-type: none"> <li>González Barahona, Jesús, Seoane Pascual, Joaquín, Robles, Gregorio. "Introducción al software libre", Fundació per a la Universitat Oberta de Catalunya, 2003. ISBN: 84-9788-028-5</li> <li>Raymond, Eric S., "La catedral y el bazar", O'Reilly, 1997.</li> <li>Gomez Gomez, Víctor Manuel. "Aspectos Históricos del Software Libre en América Latina", 2005.</li> <li>Fogel, Karl. "Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre", O'Reilly, 2007.</li> <li>Martin, Robert C. "Clean Code", 2008. ISBN-10: 0132350882</li> <li>Gamma, Erich, Helm, Richard, Johnson, Ralph, Vlissides, John. "Design patterns : elements of reusable object-oriented software", 1994. ISBN-10: 0201633612</li> <li>Black, Rex. "Managing the Testing Process, 3rd Edition", Wiley, 2011.</li> <li>D'Elia Branco, Marcelo. "El Software Libre y sus perspectivas para el desarrollo en América Latina y el Caribe", <a href="http://www.bellanet.org">http://www.bellanet.org</a>.</li> <li>Rosa, Fernando da; Heinz, Federico. "Guía práctica sobre software libre: su selección y aplicación local en América Latina y el Caribe", UNESCO, 2007. ISBN: 92-9089-103-3</li> </ul>				
<b>Bibliografía Complementaria</b>	<ul style="list-style-type: none"> <li>Williams, Sam. "Free as in Freedom: Richard Stallman's Crusade for Free Software", O'Reilly, 2002.</li> </ul>				





	<ul style="list-style-type: none"> <li>• St. Laurent, Andrew M. “Understanding Open Source and Free Software Licensing”, O’Reilly, 2004, ISBN: 0-596-00581-4.</li> <li>• Wheeler, David A. “Make Your Open Source Software GPL-Compatible. Or Else.”, 2014, <a href="http://www.dwheeler.com/essays/gpl-compatible.html">http://www.dwheeler.com/essays/gpl-compatible.html</a></li> <li>• Hernández Gómez, Lorena. “Herramientas software para el trabajo científico colaborativo”, Trabajo de Fin de Máster, Universidad de Salamanca, 2011.</li> <li>• Collins-Sussman, Ben, Fitzpatrick, Brian W., Pilato, C. Michael. “Control de versiones con Subversion”, O’Reilly, 2004.</li> <li>• Loeliger, Jon, McCullough, Matthew. “Version Control with Git, 2nd Edition”, O’Reilly, 2012.</li> <li>• O’Sullivan, Bryan. “Mercurial: The Definitive Guide”, O’Reilly, 2009.</li> <li>• Gyerik, Janos. “Bazaar Version Control”, Packt Publishing, 2013.</li> <li>• Lesyuk, Andriy. “Mastering Redmine”, Packt Publishing, 2013.</li> <li>• Fowler, Martin, Beck, Kent, Brant, John, Opdyke, William, Roberts, Don. “Refactoring: Improving the Design of Existing Code”, 1999. ISBN-10: 0201485672</li> <li>• Sokol, Francisco Zigmund, Aniche, Mauricio Finavaro, Gerosa, Marco Aurélio. “Does the Act of Refactoring Really Make Code Simpler? A Preliminary Study”, Workshop Brasileiro de Métodos Ágeis, 2013.</li> <li>• Burns, Ed, Prakash, Winston. “Hudson Continuous Integration in Practice”, 2013. IS</li> <li>• “BN-10: 0071804285</li> <li>• Berg, Alan. “Jenkins Continuous Integration Cookbook”, Packt Publishing, 2012.</li> <li>• Smart, John Ferguson. “Jenkins: The Definitive Guide”, 2011. ISBN-10: 1449305350</li> <li>• Nelson-Smith, Stephen. “Test-driven Infrastructure with Chef, 2nd Edition”, O’Reilly, 2013.</li> <li>• Stallman, Richard M. “Free Software, Free Society: Selected Essays of Richard M. Stallman”, Free Software Foundation, 200</li> <li>• Zúñiga, Lena. “Voces libres de los campos digitales: Una investigación social sobre el Software Libre en América Latina y el Caribe”, 2006. ISBN: 9977-12-888-X</li> <li>• Salas, Margarita. “Género y software libre en América Latina: Un estudio de caso”, <a href="http://www.sulabatsu.com/voces/Documentos/genero.pdf">http://www.sulabatsu.com/voces/Documentos/genero.pdf</a></li> <li>• Wheeler, David A. “Open Source Software (OSS or FLOSS) and the U.S. Department of Defense (DoD)”, 2009. <a href="http://www.dwheeler.com/essays/dod-oss.pdf">http://www.dwheeler.com/essays/dod-oss.pdf</a></li> </ul>																					
<b>Distribución de docentes por curso</b>	<table border="1"> <thead> <tr> <th><i>Curso</i></th> <th><i>Turno</i></th> <th><i>Día y Horas</i></th> <th><i>Profesor</i></th> <th><i>JefeTrab. Práct.</i></th> <th><i>Ayudante</i></th> <th><i>Ayudante ad honorem</i></th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td colspan="7" style="height: 100px;"></td> </tr> </tbody> </table>	<i>Curso</i>	<i>Turno</i>	<i>Día y Horas</i>	<i>Profesor</i>	<i>JefeTrab. Práct.</i>	<i>Ayudante</i>	<i>Ayudante ad honorem</i>														
	<i>Curso</i>	<i>Turno</i>	<i>Día y Horas</i>	<i>Profesor</i>	<i>JefeTrab. Práct.</i>	<i>Ayudante</i>	<i>Ayudante ad honorem</i>															



Firma:

Aclaración: