

MODALIDAD ACADÉMICA

Asignatura	Ingeniería de Software de Fuentes Abiertas/Libre	
Carrera	INGENIERÍA EN SISTEMAS DE INFORMACIÓN	
Ciclo Lectivo	2018	
Vigencia del programa	Desde el ciclo lectivo 2018	
Plan	2008	
Nivel	<input type="checkbox"/> 1er. Nivel <input type="checkbox"/> 2do. Nivel <input type="checkbox"/> 3er. Nivel <input type="checkbox"/> 4to. Nivel <input checked="" type="checkbox"/> 5to. Nivel	
Coordinador de la Cátedra	Dr. Ricardo Medel	
Área de Conocimiento	<input checked="" type="checkbox"/> Programación <input type="checkbox"/> Computación <input type="checkbox"/> Sistemas de Información <input checked="" type="checkbox"/> Gestión Ingenieril <input type="checkbox"/> Modelos <input type="checkbox"/> Complementaria	
Carga horaria semanal	6 horas semanales	
Anual/ cuatrimestral	Cuatrimestral (1° cuatrimestre)	
Contenidos Mínimos, según Diseño Curricular-Ordenanza 1150 (sólo para asignaturas curriculares)	N/A	
Correlativas para cursar (según Diseño Curricular-Ordenanza 1150)	Regulares	Aprobadas
	<ul style="list-style-type: none"> Al menos 1 materia del conjunto PAVII-DLC 	<ul style="list-style-type: none"> Diseño de Sistemas
Correlativas para rendir (según Diseño Curricular-Ordenanza 1150)	Regulares	Aprobadas
	<ul style="list-style-type: none"> Al menos 1 materia del conjunto PAVII-DLC Ingeniería de Software de Fuentes Abiertas/Libre (Electiva) 	<ul style="list-style-type: none"> Diseño de Sistemas
Objetivos de la Asignatura	<u>Generales</u> <ul style="list-style-type: none"> Entender los conceptos fundamentales, historia y actualidad del software de fuentes abiertas/libre. Comprender los factores que afectan la gestión de proyectos de software de fuentes abiertas/libre y cómo enfrentarlos. Adquirir los conocimientos necesarios para participar de un proyecto de software de fuentes abiertas/libre. <u>Específicos</u> <ul style="list-style-type: none"> Tener los conocimientos necesarios que permitan evaluar la conveniencia de utilizar software de fuentes abiertas/libre en la práctica profesional. Saber utilizar herramientas conceptuales y de software que permiten participar en proyectos de desarrollo de software de fuentes abiertas/libre. Entender los conceptos de calidad e integración continua aplicados al 	

	<p>desarrollo de software de fuentes abiertas/libre y utilizar herramientas apropiadas.</p> <ul style="list-style-type: none"> • Obtener la capacidad de analizar cómo el software afecta las relaciones sociales y económicas en la sociedad. • Poder definir un modelo de negocios basado en software de fuentes abiertas/libre y el modelo cooperativo de creación de valor.
--	---

Programa Analítico

UNIDAD N° 1: Conceptos básicos e historia del software de fuentes abiertas/libre

OBJETIVOS ESPECÍFICOS:

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

CONTENIDOS:

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.

BIBLIOGRAFÍA OBLIGATORIA:

- 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
- Gomez Gomez, Víctor Manuel. “Aspectos Históricos del Software Libre en América Latina”, 2005.

BIBLIOGRAFÍA COMPLEMENTARIA:

- Williams, Sam. “Free as in Freedom: Richard Stallman's Crusade for Free Software”, O'Reilly, 2002.
- Raymond, Eric S., “La catedral y el bazar”, O'Reilly, 1997.
- St. Laurent, Andrew M. “Understanding Open Source and Free Software Licensing”, O'Reilly, 2004, ISBN: 0-596-00581-4.
- Wheeler, David A. “Make Your Open Source Software GPL-Compatible. Or Else.”, 2014, <http://www.dwheeler.com/essays/gpl-compatible.html>

EVALUACIÓN:

Según la metodología definida para esta asignatura, este temario se evaluará por participación en clase del alumno, ejercicios prácticos a entregar y durante el examen parcial.

UNIDAD N° 2: Licencias de software

OBJETIVOS ESPECÍFICOS:

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, permitiéndole tomar decisiones sobre el uso adecuado de dicho software.

CONTENIDOS:

Licencias de software: privativas, permisivas, recíprocas totales, recíprocas parciales. Legislación sobre software de fuentes abiertas/libre.

BIBLIOGRAFÍA OBLIGATORIA:

- 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
- St. Laurent, Andrew M. “Understanding Open Source and Free Software Licensing”, O'Reilly, 2004,

ISBN: 0-596-00581-4.

BIBLIOGRAFÍA COMPLEMENTARIA:

- Raymond, Eric S., “La catedral y el bazar”, O’Reilly, 1997.
- Wheeler, David A. “Make Your Open Source Software GPL-Compatible. Or Else.”, 2014, <http://www.dwheeler.com/essays/gpl-compatible.html>

EVALUACIÓN:

Según la metodología definida para esta asignatura, este temario se evaluará por participación en clase del alumno, ejercicios prácticos a entregar y durante el examen parcial.

UNIDAD N°3: Gestión y calidad de proyectos de software de fuentes abiertas/libre

OBJETIVOS ESPECÍFICOS:

Desarrollar en el estudiante una comprensión de los factores que afectan la gestión de proyectos de software de fuentes abiertas/libre y cómo enfrentarlos. 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.

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:

Tipos de comunicación en proyectos de software de fuentes abiertas/libre, ventajas y limitaciones Sistemas de control de versión. Infraestructura para alojamiento de proyectos en la web. Distintas formas de participación en comunidades de software de fuentes abiertas/libre. Sistemas de control y seguimiento de bugs. Recepción y revisión de patches. Tests automatizados. Triaging de bugs. Herramientas de build e integración continua.

BIBLIOGRAFÍA OBLIGATORIA:

- Fogel, Karl. “Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre”, O’Reilly, 2007.
- Loeliger, Jon, McCullough, Matthew. “Version Control with Git, 2nd Edition”, O’Reilly, 2012.
- Black, Rex. “Managing the Testing Process, 3rd Edition”, Wiley, 2011.

BIBLIOGRAFÍA COMPLEMENTARIA:

- 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.
- 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.
- Berg, Alan. “Jenkins Continuous Integration Cookbook”, Packt Publishing, 2012.
- Burns, Ed, Prakash, Winston. “Hudson Continuous Integration in Practice”, 2013. ISBN-10: 0071804285
- 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:

Según la metodología definida para esta asignatura, este temario se evaluará por participación en clase del alumno, ejercicios prácticos a entregar y durante el examen parcial.

UNIDAD N° 4: Modelos de negocios con software de fuentes abiertas/libre

OBJETIVOS ESPECÍFICOS:

Desarrollar en el estudiante la habilidad de comprender los conceptos básicos requeridos para analizar modelos de negocios basados en software de fuentes abiertas/libre, asegurar su viabilidad y poder valorar el aporte al mercado. Brindar al estudiante los conocimientos necesarios para crear nuevos negocios en esta área.

CONTENIDOS:

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. Casos de estudio de negocios exitosos y fracasos basados en software de fuentes abiertas/libre.

BIBLIOGRAFÍA OBLIGATORIA:

- Fogel, Karl. “Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre”, O’Reilly, 2007.
- 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

BIBLIOGRAFÍA COMPLEMENTARIA:

- D’Elia Branco, Marcelo. “El Software Libre y sus perspectivas para el desarrollo en América Latina y el Caribe”, <http://www.bellanet.org>.

EVALUACIÓN:

Según la metodología definida para esta asignatura, este temario se evaluará por participación en clase del alumno, ejercicios prácticos a entregar y durante el examen parcial.

UNIDAD N° 5: Sistemas de operativos de fuentes abiertas/libre

OBJETIVOS ESPECÍFICOS:

Brindar al estudiante conocimiento sobre los principales sistemas de software de fuentes abiertas/libres utilizados en la industria del software en la actualidad.

Permitir al estudiante obtener conocimiento práctico sobre el software de base y sistemas complejos que permiten el desarrollo de un completo ecosistema de software de fuentes abiertas/libre.

CONTENIDOS:

Lenguajes de programación libres (GCC, Clang/LLVM, Python). Sistemas operativos libres/de fuentes abiertas (GNU/Linux, FreeBSD, ReactOS). Bases de datos de fuentes abiertas/libres (MySQL, PostgreSQL, MongoDB).

BIBLIOGRAFÍA OBLIGATORIA:

- Kroah-Hartman, Greg. “Linux Kernel in a Nutshell”, O’Reilly Media, 2006. ISBN : 0-596-10079-5
- Redmond, Eric, Wilson, Jim R. “Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement”, 2012.

BIBLIOGRAFÍA COMPLEMENTARIA:

- Gough, Brian J. “An Introduction to GCC”, 2004. ISBN: 0-9541617-9-3
- Lattner, Chris. “The Design of LLVM”, Dr. Dobb's Journal, Mayo de 2012.
- Lutz, Mark. “Learning Python (5th ed.)”, O’Reilly Media, 2013. ISBN 978-0-596-15806-4
- Dyer, Russell J. T. “MySQL in a Nutshell”, O’Reilly Media, 2008.

EVALUACIÓN:

Según la metodología definida para esta asignatura, este temario se evaluará por participación en clase del alumno, ejercicios prácticos a entregar y durante el examen parcial.

UNIDAD N° 6: Diseño e implementación de software de fuentes abiertas/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 OBLIGATORIA:

- Fogel, Karl. “Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre”, O’Reilly, 2007.

BIBLIOGRAFÍA COMPLEMENTARIA:

- 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:

Según la metodología definida para esta asignatura, este temario se evaluará por participación en clase del alumno, ejercicios prácticos a entregar y durante el examen parcial.

UNIDAD N° 7: Software de fuentes abiertas/libres en otros ámbitos

OBJETIVOS ESPECÍFICOS:

Desarrollar en el estudiante los conceptos básicos requeridos para analizar y valorar los aportes de la filosofía del software libre aplicada en otros ámbitos: educación, estado, movimientos sociales, cooperativismo. Desarrollar en el estudiante la capacidad de analizar cómo el software afecta las relaciones sociales y económicas en la sociedad.

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.

BIBLIOGRAFÍA OBLIGATORIA:

- Stallman, Richard M. “Free Software, Free Society: Selected Essays of Richard M. Stallman”, Free Software Foundation, 2001.
- 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

BIBLIOGRAFÍA COMPLEMENTARIA:

- 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
- 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:

Según la metodología definida para esta asignatura, este temario se evaluará por participación en clase del alumno, ejercicios prácticos a entregar y durante el examen parcial.

<p>Metodología de enseñanza y aprendizaje</p>	<p>Clases participativas teórico-prácticas Durante las clases un grupo previamente seleccionado de alumnos guiará la discusión sobre los temas pre-definidos. Todos los alumnos participarán aportando su conocimiento, tratando de aclarar sus dudas y aportando para llegar a conclusiones generales. Con este tipo de clases se busca interrelacionar los fundamentos teóricos con las experiencias prácticas en el desarrollo y aplicación de software de fuentes abiertas/libre y abrir la discusión a nuevas visiones de la problemática.</p> <p>Proyecto de Aplicación El alumno participará activamente en un proyecto de software de fuentes abiertas/libre ya existente, realizado por una comunidad externa a la asignatura. Este tipo de actividad está orientada a ejercitar en un ambiente real los temas discutidos durante las clases de la asignatura. Debido a la complejidad de insertarse en tan poco tiempo en una comunidad de software de fuentes abiertas/libre, la participación del alumno puede ser bien como colaborador en la documentación, como tester o como desarrollador.</p>
<p>Sistema de evaluación (Nombrar y describir cada una de las diferentes instancias de evaluación)</p>	<p>La evaluación de los alumnos se compone de los siguientes cuatro componentes:</p> <ol style="list-style-type: none"> 1. Participación en clase. El docente evaluará cada clase la magnitud y calidad de la participación de cada alumno. 2. Ejercicios prácticos o ensayos de realización obligatoria, con entrega pautada. 3. Examen parcial integrador y/o su instancia recuperatoria. 4. Presentación (oral y escrita) sobre su participación en el proyecto de software libre mencionado anteriormente (Proyecto de Aplicación). <p>Como se indica en el punto 3, habrá un examen de recuperación en el caso de que el alumno no alcanzara el mínimo esperado en el examen parcial integrador.</p>
<p>Regularidad: condiciones (Describir las condiciones necesarias para regularizar. Se sugiere incluir la aclaración que el estudiante en condición de regular puede rendir en el plazo de un ciclo lectivo sin control de correlativas aprobadas)</p>	<p>La asignatura no permite la Regularidad.</p>
<p>Promoción: condiciones (Aclarar si hubiera promoción de alguna parte de la asignatura, las condiciones y si tiene duración, con el mayor detalle posible)</p>	<p>La asignatura no permite la Promoción.</p>
<p>Aprobación Directa: condiciones. (la calificación será la nota)</p>	<p>La calificación del alumno en la materia será el promedio de las notas de los cuatro componentes mencionados en la sección Sistema de Evaluación.</p>

registrada como Nota Final en Autogestión) (Se sugiere incluir la aclaración que el estudiante, en esta condición, puede registrar su nota en examen en el plazo de un ciclo lectivo, sin control de correlativas aprobadas, y después de ello se le exigirán correlativas aprobadas)	El alumno estará en condiciones de Aprobación Directa si su calificación es igual o mayor a 7 (siete).
Modalidad de examen final (Describir las características metodológicas del examen final para los distintos estados del estudiante)	N/A
Actividades en laboratorio	N/A
Horas/año totales de la asignatura (hs. cátedra)	90
Cantidad de horas prácticas totales (hs. cátedra)	30
Cantidad de horas teóricas totales (hs. cátedra)	60
Tipo de formación práctica (sólo si es asignatura curricular -no electiva-)	<input type="checkbox"/> Formación experimental <input type="checkbox"/> Resolución de problemas de ingeniería <input type="checkbox"/> Actividades de proyecto y diseño <input type="checkbox"/> Prácticas supervisadas en los sectores productivos y /o de servicios
Cantidad de horas cátedras afectadas a la formación práctica indicada en el punto anterior (sólo si es asignatura curricular -no electiva-)	
Descripción de los prácticos	<p>Las actividades prácticas constarán de dos aspectos:</p> <ul style="list-style-type: none"> • Realización de prácticos o ensayos sobre temas de la asignatura a ser entregados en fechas pautadas oportunamente. • Actividades realizadas en el marco de un proyecto existente de software libre a elección del alumno, las que serán pautadas previamente con el profesor.
Criterios generales (los cuales serán tenidos en cuenta en las correcciones)	<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"> • Precisión Conceptual • Capacidad de análisis y síntesis • Criterios para transferir los conocimientos adquiridos a situaciones prácticas • Cumplimiento de los requerimientos de la actividad • El lenguaje técnico utilizado sea el adecuado a la actividad • La creatividad de la presentación de la actividad

<p>Cronograma de actividades de la asignatura (contemplando las fechas del calendario 2018 y para cada unidad)</p>	<table border="1"> <thead> <tr> <th>Fecha</th> <th>Clase</th> <th>Tema</th> </tr> </thead> <tbody> <tr> <td>24/3/2018</td> <td>Feriado</td> <td>-</td> </tr> <tr> <td>31/03/2018</td> <td>Semana Santa</td> <td>-</td> </tr> <tr> <td>07/4/2018</td> <td>1° clase</td> <td>Intro</td> </tr> <tr> <td>14/04/18</td> <td>2° clase</td> <td>Unidad 1</td> </tr> <tr> <td>21/04/18</td> <td>3° clase</td> <td>Unidad 2</td> </tr> <tr> <td>28/04/18</td> <td>Feriado largo</td> <td>-</td> </tr> <tr> <td>05/05/18</td> <td>4° clase</td> <td>Unidad 3</td> </tr> <tr> <td>12/05/18</td> <td>5° clase</td> <td>Unidad 4</td> </tr> <tr> <td>19/05/18</td> <td>6° clase</td> <td>Unidad 5</td> </tr> <tr> <td>26/05/18</td> <td>Feriado largo</td> <td>-</td> </tr> <tr> <td>02/06/18</td> <td>7° clase</td> <td>Unidad 6</td> </tr> <tr> <td>09/06/18</td> <td>8° clase</td> <td>Parcial</td> </tr> <tr> <td>16/06/18</td> <td>9° clase</td> <td>Recup/Repaso</td> </tr> <tr> <td>23/06/18</td> <td>10° clase</td> <td>Unidad 7</td> </tr> <tr> <td>30/06/18</td> <td>11° clase</td> <td>Exposiciones</td> </tr> </tbody> </table>	Fecha	Clase	Tema	24/3/2018	Feriado	-	31/03/2018	Semana Santa	-	07/4/2018	1° clase	Intro	14/04/18	2° clase	Unidad 1	21/04/18	3° clase	Unidad 2	28/04/18	Feriado largo	-	05/05/18	4° clase	Unidad 3	12/05/18	5° clase	Unidad 4	19/05/18	6° clase	Unidad 5	26/05/18	Feriado largo	-	02/06/18	7° clase	Unidad 6	09/06/18	8° clase	Parcial	16/06/18	9° clase	Recup/Repaso	23/06/18	10° clase	Unidad 7	30/06/18	11° clase	Exposiciones
Fecha	Clase	Tema																																															
24/3/2018	Feriado	-																																															
31/03/2018	Semana Santa	-																																															
07/4/2018	1° clase	Intro																																															
14/04/18	2° clase	Unidad 1																																															
21/04/18	3° clase	Unidad 2																																															
28/04/18	Feriado largo	-																																															
05/05/18	4° clase	Unidad 3																																															
12/05/18	5° clase	Unidad 4																																															
19/05/18	6° clase	Unidad 5																																															
26/05/18	Feriado largo	-																																															
02/06/18	7° clase	Unidad 6																																															
09/06/18	8° clase	Parcial																																															
16/06/18	9° clase	Recup/Repaso																																															
23/06/18	10° clase	Unidad 7																																															
30/06/18	11° clase	Exposiciones																																															
<p>Propuesta para la atención de consultas y mail de contacto.</p>	<p>El estudiante podrá contactar al coordinador de la cátedra de la siguiente manera: En la oficina 4 del Edificio Maders lunes y viernes de 18:00 a 20:00, o bien a través de la casilla de email: ricardo.h.medel@gmail.com</p>																																																
<p>Plan de integración con otras asignaturas</p>	<p>Esta asignatura requiere de los conocimientos desarrollados en las siguientes asignaturas:</p> <ul style="list-style-type: none"> Diseño de Sistemas 																																																
<p>Bibliografía Obligatoria</p>	<ul style="list-style-type: none"> 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 Gomez Gomez, Víctor Manuel. "Aspectos Históricos del Software Libre en América Latina", 2005. St. Laurent, Andrew M. "Understanding Open Source and Free Software Licensing", O'Reilly, 2004, ISBN: 0-596-00581-4. Fogel, Karl. "Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre", O'Reilly, 2007. Loeliger, Jon, McCullough, Matthew. "Version Control with Git, 2nd Edition", O'Reilly, 2012. Black, Rex. "Managing the Testing Process, 3rd Edition", Wiley, 2011. Kroah-Hartman, Greg. "Linux Kernel in a Nutshell", O'Reilly Media, 2006. ISBN : 0-596-10079-5 Redmond, Eric, Wilson, Jim R. "Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement", 2012. Stallman, Richard M. "Free Software, Free Society: Selected Essays of Richard M. Stallman", Free Software Foundation, 2001. 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 Raymond, Eric S., "La catedral y el bazar", O'Reilly, 1997. 																																																
<p>Bibliografía Complementaria</p>	<ul style="list-style-type: none"> Williams, Sam. "Free as in Freedom: Richard Stallman's Crusade for Free Software", O'Reilly, 2002. Wheeler, David A. "Make Your Open Source Software GPL-Compatible. Or Else.", 2014, http://www.dwheeler.com/essays/gpl-compatible.html 																																																

- 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.
- 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.
- Berg, Alan. “Jenkins Continuous Integration Cookbook”, Packt Publishing, 2012.
- Burns, Ed, Prakash, Winston. “Hudson Continuous Integration in Practice”, 2013. ISBN-10: 0071804285
- Smart, John Ferguson. “Jenkins: The Definitive Guide”, 2011. ISBN-10: 1449305350
- Nelson-Smith, Stephen. “Test-driven Infrastructure with Chef, 2nd Edition”, O’Reilly, 2013.
- D’Elia Branco, Marcelo. “El Software Libre y sus perspectivas para el desarrollo en América Latina y el Caribe”, <http://www.bellanet.org>.
- Gough, Brian J. “An Introduction to GCC”, 2004. ISBN: 0-9541617-9-3
- Lattner, Chris. “The Design of LLVM”, Dr. Dobb's Journal, Mayo de 2012.
- Lutz, Mark. “Learning Python (5th ed.)”, O’Reilly Media, 2013. ISBN 978-0-596-15806-4
- Dyer, Russell J. T. “MySQL in a Nutshell”, O’Reilly Media, 2008.
- 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.
- 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
- 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>

Distribución de docentes

<i>Curso</i>	<i>Turno</i>	<i>Día y Horas</i>	<i>Profesor</i>	<i>JTP</i>	<i>Ayudante</i>
<i>Único</i>	<i>Mañana</i>	<i>Sábado 8-10:30 hs.</i>	<i>Medel</i>	-	-

Firma:

Aclaración: