

MODALIDAD ACADÉMICA

Asignatura	Ingeniería de Software de Fuentes Abiertas/Libre	
Carrera	INGENIERÍA EN SISTEMAS DE INFORMACIÓN	
Ciclo Lectivo	2021	
Vigencia del programa	<i>Desde el ciclo lectivo 2021</i>	
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	<i>Dr. Ricardo Medel</i>	
Área de Conocimiento	<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 <input checked="" type="checkbox"/> Asignatura Electiva	
Carga horaria semanal	<i>6 horas semanales</i>	
Anual / cuatrimestral	<i>Cuatrimstral (1° cuatrimestre)</i>	
Contenidos Mínimos, según Diseño Curricular-Ordenanza 1150 (sólo para asignaturas curriculares, no electivas)	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"> N/A 	<ul style="list-style-type: none"> N/A
Objetivos generales de la Asignatura	<ul style="list-style-type: none"> Describir adecuadamente los conceptos fundamentales del software de fuentes abiertas/libre, su historia y el estado actual de situación. Ser capaz de participar activamente de una comunidad de software de fuentes abiertas/libre. Aplicar a la gestión de comunidades de software de fuentes abiertas/libre las características de los procesos de desarrollo de software de este tipo. 	
<u>Programa Analítico</u>		
Unidad Nro. 1: CONCEPTOS E HISTORIA DEL SOFTWARE DE FUENTES ABIERTAS/LIBRE		
<p>Resultados de Aprendizaje: Reconocer los principales eventos de la historia del software que incidieron en la formación del movimiento de software libre y de la iniciativa del software de fuentes abiertas, a fin de avizorar cómo la configuración actual del ecosistema del software puede modificarse en el futuro. Explicar las similitudes y diferencias entre las definiciones de software libre y software de fuentes abiertas a fin de poder utilizar el tipo de software más adecuado según el contexto de desempeño profesional.</p>		

Contenidos: Historia de la computación en general y del software en particular. Surgimiento del software privativo. Proyecto GNU: el nacimiento del movimiento del software libre. Definición de software libre. La Iniciativa Open Source (OSI). Definición de software de fuentes abiertas. Comparación entre software libre, software de fuentes abiertas, shareware y freeware. Historia del software de fuentes abiertas/libre en Latinoamérica en general y en Argentina en particular.

Bibliografía Obligatoria:

- Gomez Gomez, V.M. (2005) Aspectos Históricos del Software Libre en América Latina.
- González Barahona, J., Seoane Pascual, J., Robles, G. (2007) Introducción al software libre. Fundació per a la Universitat Oberta de Catalunya. ISBN: 84-9788-028-5
- Medel, R. y otros (2018) Manual de la Asignatura Ingeniería de Software de Fuentes Abiertas/Libre.

Bibliografía Complementaria:

- Manresa, R.G. (compilador) (2020) La odisea del software libre.
- Red Hat (2021) The State of Enterprise Open Source.
- Williams, S. (2002) Free as in Freedom: Richard Stallman's Crusade for Free Software. O'Reilly.
- Zanotti, A. (2011) Reescribiendo Tecnologías: Aproximaciones al movimiento del software libre y su difusión en Argentina.

Evaluación: Se evaluará al estudiante según su participación en las discusiones y ejercicios en clase, y su participación en las discusiones de foros del Aula Virtual, donde se presentarán temáticas relacionadas a la historia del software y comparaciones entre diferentes tipos de software y su aplicación en diversos contextos.

Unidad Nro. 2: LICENCIAS DE SOFTWARE Y PROPIEDAD INTELECTUAL

Resultados de Aprendizaje: Listar las formas posibles de protección del software como producto intelectual, sus ventajas y desventajas en cada contexto, a fin de elegir la protección más adecuada según el contexto en que se desempeñe el futuro profesional. Confeccionar una grilla comparativa de las diferentes licencias libres y su compatibilidad con el propósito de seleccionar el subconjunto de ellas más adecuado para cada caso.

Contenidos: Propiedad intelectual y derechos de autor. Herramientas legales para protección de la propiedad intelectual, patentes, licencias, registro como obra. Estado de la propiedad intelectual del software en el mundo en general y en Argentina en particular. Clasificación y comparación entre licencias de software privativas y libres, estrictas y permisivas. Tablas de compatibilidad de licencias.

Bibliografía Obligatoria:

- González Barahona, J., Seoane Pascual, J., Robles, G. (2007) Introducción al software libre. Fundació per a la Universitat Oberta de Catalunya. ISBN: 84-9788-028-5
- St. Laurent, A.M. (2004) Understanding Open Source and Free Software Licensing. O'Reilly. ISBN: 0-596-00581-4

Bibliografía Complementaria:

- Raymond, E.S. (1997) La catedral y el bazar. O'Reilly.
- Wheeler, D.A. (2014) Make Your Open Source Software GPL-Compatible. Or Else.

Evaluación: Se evaluará al estudiante según su participación en las discusiones y ejercicios en clase, y su participación en las discusiones de foros del Aula Virtual, donde se presentarán temáticas relacionadas a la protección de la propiedad intelectual en general y de las licencias de software en particular. Se evaluará también la adecuada comprensión de la/s licencia/s de software que rigen la comunidad elegida por cada

estudiante según lo presentado en el primer informe escrito.

Unidad Nro. 3: GESTIÓN Y CALIDAD DE PROYECTOS DE SOFTWARE LIBRE

Resultados de Aprendizaje: Establecer los factores que afectan la gestión de un proyecto de software libre basado en el desarrollo por comunidad y seleccionar las herramientas digitales adecuadas para su gestión. Definir el modelo de gobernanza más adecuado para una comunidad de software libre. Gestionar el cambio y las expectativas cambiantes de los integrantes de la comunidad de software libre.

Contenidos: Modelos de desarrollo de software de fuentes abiertas/libre. Tipos de participación en comunidades de software de fuentes abiertas/libre. Gestión de proyectos distribuidos globalmente basados en comunidades de software de fuentes abiertas/libre. Gobernanza y comunicación en comunidades de software de fuentes abiertas/libre, problemáticas comunes en la gestión y herramientas digitales para su administración. Problemáticas que lleva al "fork" de proyectos.

Bibliografía Obligatoria:

- Fogel, K. (2007) Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre. O'Reilly.

Bibliografía Complementaria:

- Hernández Gómez, L. (2011) Herramientas software para el trabajo científico colaborativo. Trabajo de Fin de Máster, Universidad de Salamanca.
- Collins-Sussman, B., Fitzpatrick, B.W., Pilato, C.. (2004) Control de versiones con Subversion. O'Reilly.
- Loeliger, J., McCullough, M. (2012) Version Control with Git, 2nd Edition. O'Reilly.
- O'Sullivan, B. (2009) Mercurial: The Definitive Guide. O'Reilly.
- Berg, A. (2012) Jenkins Continuous Integration Cookbook. Packt Publishing.
- Black, R. (2011) Managing the Testing Process, 3rd Edition. Wiley.
- Nelson-Smith, S. (2013) Test-driven Infrastructure with Chef, 2nd Edition. O'Reilly.

Evaluación: Se evaluará al estudiante según su participación en las discusiones y ejercicios en clase, y su participación en las discusiones de foros del Aula Virtual, donde se presentarán temáticas relacionadas a la gestión de proyectos de software libre basados en comunidades. Se evaluará también la adecuada comprensión de las herramientas de gestión utilizadas en la comunidad elegida por cada estudiante según lo presentado en el segundo informe escrito.

Unidad Nro. 4: MODELOS DE NEGOCIOS CON SOFTWARE LIBRE

Resultados de Aprendizaje: Reconocer los modelos de negocios de las diferentes comunidades de software libre a fin de valorar la sustentabilidad de los proyectos. Proponer modelos de negocios viables para proyectos de software libre propios o de la comunidad.

Contenidos: Comparación de los modelos de negocios tradicionales y los basados en software de fuentes abiertas/libre. Planes de negocios y startups basadas en software libre. Cooperativismo en la industria del desarrollo de software y servicios informáticos. Casos de estudio de negocios exitosos y fracasos basados en software libre.

Bibliografía Obligatoria:

- Krishnamurthy, S. (2003) An Analysis of Open Source Business Models. University of Washington.
- Fogel, K. (2007) Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre. O'Reilly.

- González Barahona, J., Seoane Pascual, J., Robles, G. (2007) Introducción al software libre. Fundació per a la Universitat Oberta de Catalunya. ISBN: 84-9788-028-5

Bibliografía Complementaria:

- Goikolea, M. (2014) Modelos de negocios en Software Libre: Un nuevo paradigma. Innovation & Entrepreneurship Business School.

Evaluación: Se evaluará al estudiante según su participación en las discusiones y ejercicios en clase, y su participación en las discusiones de foros del Aula Virtual, donde se presentarán temáticas relacionadas a los modelos de negocios de proyectos de software libre y de fuentes abiertas. Se evaluará también la adecuada comprensión del modelo de negocio propuesto en la comunidad elegida por cada estudiante según lo presentado en el segundo informe escrito.

Unidad Nro. 5: SISTEMAS DE OPERATIVOS LIBRES

Resultados de Aprendizaje: Describir las raíces de los sistemas operativos libres desde la creación de Unix y sus derivaciones en sistemas libres y privativos a fin de recomendar el uso de cada sistema en el contexto adecuado. Seleccionar la distribución Linux o versión BSD más adecuada a cada situación en el contexto laboral.

Contenidos: Historia del sistema operativo UNIX y sus derivados. Historia y actualidad de los principales sistemas operativos libres: GNU/Linux, familia BSD, ReactOS, etc. Sistemas operativos monolíticos: el kernel y las herramientas de sistema. El concepto de distribuciones (distros). Sistemas operativos no libres derivados de sistemas libres: Android y iOS.

Bibliografía Obligatoria:

- Kroah-Hartman, G. (2006) Linux Kernel in a Nutshell. O'Reilly Media. ISBN : 0-596-10079-5
- Raymond, E.S. (1997) La catedral y el bazar. O'Reilly.

Bibliografía Complementaria:

- López, J.M. (2017) UNIX: el padre de los sistemas operativos actuales. Blogthinkbig.com.
- Isaac (2019) Una historia de UNIX: historia de Unix. ArchiTecnologia.es.

Evaluación: Se evaluará al estudiante según su participación en las discusiones y ejercicios en clase, y su participación en las discusiones de foros del Aula Virtual, donde se presentarán temáticas relacionadas a los diferentes sistemas operativos libres.

Unidad Nro. 6: DISEÑO E IMPLEMENTACIÓN DE SOFTWARE LIBRE

Resultados de Aprendizaje: Establecer los factores que afectan el desarrollo de un proyecto de software libre y seleccionar las herramientas digitales adecuadas para el desarrollo, sostén de infraestructura y comunicaciones entre integrantes del equipo. Definir el modelo de desarrollo más adecuado para cada proyecto de software libre. Establecer las reglas de participación de la comunidad de modo de asegurar la continuidad de los proyectos y obtener el mejor rendimiento de cada participante.

Contenidos: Herramientas de desarrollo distribuido: comunicaciones, control de versiones, infraestructura de alojamiento, control y seguimiento de errores (bugs/issues), testing automatizado e integración continua. Comparación de guías de estilo de diferentes comunidades de software fuentes abiertas/libre. Integración, gobernanza y códigos de conducta de desarrolladores en un proyecto de software de fuentes abiertas/libre.

Bibliografía Obligatoria:

- Fogel, K. (2007) Producir software de código abierto: Cómo llevar a buen puerto un proyecto de

código libre. O'Reilly.

- Ferreira, C., Souza, C., Pinto, G., Steinmacher, I., Meirelles, P. (2018) When Students Become Contributors: Leveraging OSS Contributions in Software Engineering Courses. Proceedings of XXXII BRAZILIAN SYMPOSIUM ON SOFTWARE ENGINEERING, Sao Carlos, Brazil, September 17–21, 2018 (SBES 2018). <https://doi.org/10.1145/3266237.3266250>
- Murphy, C., Buffardi, K., Dehlinge, J., Lambert, L., Veilleux, N. (2017) Community Engagement with Free and Open Source Software. SIGCSE'17, March 8–11, 2017, Seattle, WA, USA. DOI: <http://dx.doi.org/10.1145/3017680.3017682>

Bibliografía Complementaria:

- Martin, R.C. (2008) Clean Code. ISBN-10: 0132350882

Evaluación: Se evaluará al estudiante según su participación en las discusiones y ejercicios en clase, y su participación en las discusiones de foros del Aula Virtual, donde se presentarán temáticas relacionadas a las metodologías y modelos de desarrollo de proyectos de software libre basados en comunidades distribuidas globalmente. Se evaluará también la adecuada comprensión del modelo de desarrollo y las reglas (escritas o no) de participación en la comunidad elegida por cada estudiante según lo presentado en el tercer informe escrito.

Unidad Nro. 7: SOFTWARE LIBRE Y SU FILOSOFÍA EN OTROS ÁMBITOS

Resultados de Aprendizaje: Reconocer la aplicación de la filosofía del software libre en diferentes ámbitos y cómo afecta los resultados de las organizaciones e instituciones que la utilizan. Implementar software libre y su filosofía en diferentes ámbitos más allá del desarrollo de software. Formar una opinión propia sobre el uso del software libre en el Estado y la educación basada en los antecedentes históricos y los requerimientos de cada contexto específico.

Contenidos: Conceptos de soberanía tecnológica (a nivel regional/nacional/provincial/municipal) y apropiación de tecnología (a nivel organizacional y personal). Impacto del software libre en la apropiación de tecnología en el mundo en general y en países periféricos en particular. El software libre y el "derecho a reparar". El software libre como movimiento social. La filosofía del software libre en otros ámbitos: educación, Estado y gobierno, movimientos sociales, cooperativismo, medios de comunicación alternativos y hardware libre. El software libre y su relación con el voto electrónico.

Bibliografía Obligatoria:

- Stallman, R.M. (2001) Free Software, Free Society: Selected Essays of Richard M. Stallman. Free Software Foundation.
- Zúñiga, L. (2006) Voces libres de los campos digitales: Una investigación social sobre el Software Libre en América Latina y el Caribe. ISBN: 9977-12-888-X

Bibliografía Complementaria:

- Diéguez, G., Gasparín, J.M., Sánchez, J., Schejtman, L. (2015) Escenarios y perspectivas del gobierno electrónico en América Latina y el Caribe. Documento de trabajo N° 132. CIPPEC.
- da Rosa, F.; Heinz, F. (2007) Guía práctica sobre software libre: su selección y aplicación local en América Latina y el Caribe. UNESCO. ISBN: 92-9089-103-3
- Benítez-Eyzaguirre, L. (2019) Ciberfeminismo y apropiación tecnológica en América Latina. Virtualis - Revista de Cultura Digital. Vol. 10, Núm. 18.
- Zúñiga, L., Xhardez, V. y otros (2006) Software Libre en América Latina y el Caribe. Documento base: Marco conceptual y metodológico. Análisis de datos cualitativos. Primer año: septiembre 2003 a septiembre 2004. Bellanet Internacional. Oficina para América Latina y el Caribe.

Evaluación: Se evaluará al estudiante según su participación en las discusiones y ejercicios en clase, y su

participación en las discusiones de foros del Aula Virtual, donde se presentarán temáticas relacionadas al impacto de la filosofía del software libre en diferentes contextos, realizando análisis de casos y proponiendo nuevas aplicaciones.

<p>Metodología de enseñanza y aprendizaje / Mediación Pedagógica (Planificar estrategias centradas en el aprendizaje activo del estudiante)</p>	<p>Clases participativas</p> <ul style="list-style-type: none"> • Antes de cada clase el cuerpo docente indicará qué conceptos se cubrirán en dicha clase y se seleccionará uno de los grupos de estudiantes previamente organizados a fin de que dirija la discusión en clase. • Durante la clase el grupo seleccionado utilizará medios audiovisuales para guiar la discusión sobre el o los temas establecidos. Los estudiantes del grupo y el cuerpo docente promoverán la participación de todos los estudiantes en la discusión a fin de relacionar los conceptos teóricos con las experiencias prácticas en el desarrollo y aplicación de software que ya poseen los estudiantes. • El cuerpo docente también propondrá actividades durante la clase de modo de aplicar los conceptos analizados. <p>Práctica en una comunidad de software libre Durante el cuatrimestre cada estudiante, bajo supervisión del cuerpo docente, participará activamente en una comunidad de software libre ya existente y en la cual no haya participado previamente. 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 estudiante 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, pensando en la Evaluación como proceso continuo de recolección de evidencias)</p>	<p>La evaluación de los estudiantes se compone de una combinación de los siguientes tres componentes:</p> <ul style="list-style-type: none"> • Participación en clase: El cuerpo docente hará un seguimiento de la participación de cada estudiante en las discusiones durante las clases. Además de considerar el trabajo realizado por el grupo en la presentación del tema asignado, se tomará nota de las participaciones orales durante las clases y revisando el "chat" del aula virtual, como forma de obtener evidencia del aprendizaje. Para esto, es requerido que cada estudiante identifique correctamente su perfil en Zoom y que antes de intervenir oralmente indique su nombre. • Participación en foros del Aula Virtual: El cuerpo docente propondrá varios temas para discutir en los foros del Aula Virtual y los estudiantes deberán participar según lo requerido. • Informes escritos de participación en comunidad: Cada estudiante deberá presentar en forma individual informes escritos sobre distintos aspectos de su experiencia de participación en la comunidad de software libre elegida. • Informe oral de participación en comunidad: Cada estudiante realizará una presentación oral reportando la experiencia completa de participación comunidad de software libre seleccionada, a modo de

	informe final integrador.
Criterios de evaluación (los cuales serán tenidos en cuenta en las correcciones)	<ul style="list-style-type: none"> • La participación de los estudiantes en clase se evaluará teniendo en cuenta la calidad del trabajo realizado por el grupo en la presentación del tema asignado, y en base a la cantidad y calidad de las intervenciones de cada estudiante durante las clases (a viva voz o en chat). • La participación de los estudiantes en los foros del Aula Virtual será en base a un análisis de su participación teniendo en cuenta tanto la cantidad como a la calidad de las intervenciones. El análisis se realizará en base a lo novedoso del aporte o enfoque y cuán fundamentado está en los conceptos vistos y analizados durante la asignatura. • Los informes escritos sobre la participación del estudiante en la comunidad de software libre de su elección serán evaluados respecto al apego a los aspectos formales (longitud, ortografía, estructura) y a la calidad del contenido en cuanto a la completitud, profundidad y corrección del análisis de la comunidad de software donde se realizó la intervención, utilizando los conceptos vistos en clase. • La presentación oral de cada estudiante sobre su participación comunidad de software libre seleccionada, será evaluada tanto en los aspectos de completitud y corrección de la información presentada respecto de los aspectos analizados en cada comunidad, como al manejo adecuado del tiempo y de las herramientas de apoyo audiovisual.
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)	Esta asignatura no permite obtener la condición de Regular.
Promoción: condiciones (Aclarar si hubiera promoción de alguna parte de la asignatura, las condiciones y si tiene duración, con el mayor detalle posible)	Esta asignatura no permite obtener la Promoción.
Aprobación Directa: condiciones. (la calificación será la nota 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 exigirá)	<p>La calificación del estudiante en la asignatura será el promedio ponderado de las notas de los componentes mencionados en la sección Sistema de Evaluación.</p> $\text{Calificación} = C * 0,2 + A * 0,2 + P * 0,3 + F * 0,3$ <p>donde C es la nota por participación en clase, A es la nota por participación en los foros del Aula Virtual, P es la nota promedio de los informes parciales y F es la nota promedio de los informes finales escrito y oral.</p> <p>El estudiante estará en condiciones de Aprobación Directa si su Calificación es igual o mayor a 7 (siete).</p>

correlativas aprobadas)	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.
Modalidad de examen final (Describir las características metodológicas del examen final para los distintos estados del estudiante)	<p>Libre: cada estudiante que desee rendir libre debe ponerse en contacto con el cuerpo docente al menos una semana antes de la fecha del examen, realizar una participación en una comunidad de software libre de su elección y presentar un informe escrito el día anterior a la fecha de examen correspondiente. Si el informe es aprobado, el estudiante podrá realizar el examen, en el cual se le realizarán preguntas sobre conceptos y aplicaciones de dichos conceptos de la temática de la asignatura.</p> <p>Regular, Promoción: la asignatura no permite estos estados.</p> <p>Aprobación Directa: cada estudiante se deberá presentar con su libreta universitaria a fin de que el docente registre allí la nota correspondiente a la Aprobación Directa.</p>
Actividades en laboratorio	Las actividades prácticas que requiere la participación de los estudiantes en la comunidad de software libre de su elección se realizarán en forma remota, fuera de los horarios de clase de la asignatura.
Cantidad de horas prácticas totales (en el aula)	N/A
Cantidad de horas teóricas totales (en el aula)	60
Cantidad de horas estimadas totales de trabajo (extra áulicas).	30
Horas/año totales de la asignatura (en el aula).	90
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-)	N/A
Descripción de los prácticos	<p>Las actividades serán realizadas en el marco de una comunidad de software libre preexistente, a elección del estudiante. El tipo de actividad específica a realizar por el estudiante en dicha comunidad serán pautadas previamente con el profesor. Una actividad que todos los estudiantes realizarán es el análisis, con las herramientas conceptuales brindadas en la asignatura, sobre la gestión de la comunidad, estructura y herramientas utilizadas, modelo de negocios y estándares de desarrollo. La participación en sí del estudiante en la comunidad variará dependiendo del estudiante y la comunidad, pero pueden ser:</p> <ul style="list-style-type: none"> • Codificación de nuevas funcionalidades • Codificación para la corrección de errores

	<ul style="list-style-type: none"> • Corrección de errores de configuración del software desarrollado o de las herramientas utilizadas para su desarrollo • Detección de errores en el software desarrollado • Creación de y/o mejoras en la documentación del código, manuales de operación y manuales de usuario • Traducción de documentación, manuales o interfaces de usuario • Mejoras en los protocolos de gestión de la comunidad • Desarrollo o mejora de material educativo basado en el software desarrollado <p>Esta lista se da a modo de sugerencia pero de ninguna manera restringe las actividades posibles de realizarse en la comunidad a las indicadas aquí.</p>																																							
<p>Cronograma de actividades de la asignatura (contemplando las fechas del calendario 2021 y para cada unidad)</p>	<table border="1"> <thead> <tr> <th>Fecha</th> <th>Clase</th> <th>Tema</th> </tr> </thead> <tbody> <tr> <td>20/03/2021</td> <td>1° clase</td> <td>Inducción a la asignatura.</td> </tr> <tr> <td>27/03/2021</td> <td>2° clase</td> <td>Conceptos e historia del soft libre.</td> </tr> <tr> <td>10/04/2021</td> <td>3° clase</td> <td>Propiedad intelectual. Licencias software.</td> </tr> <tr> <td>17/04/2021</td> <td>4° clase</td> <td>Gestión de proyectos de SL.</td> </tr> <tr> <td>24/04/2021</td> <td>5° clase</td> <td>Modelos de negocios con SL.</td> </tr> <tr> <td>08/05/2021</td> <td>6° clase</td> <td>Sistemas operativos de SL.</td> </tr> <tr> <td>15/05/2021</td> <td>7° clase</td> <td>Clase de repaso y consulta.</td> </tr> <tr> <td>29/05/2021</td> <td>8° clase</td> <td>Diseño e implementación de SL.</td> </tr> <tr> <td>05/06/2021</td> <td>9° clase</td> <td>SL en otros ámbitos.</td> </tr> <tr> <td>12/06/2021</td> <td>10° clase</td> <td>Clase de cierre y consulta.</td> </tr> <tr> <td>19/06/2021</td> <td>11° clase</td> <td>Presentaciones orales (1° sesión).</td> </tr> <tr> <td>26/06/19</td> <td>12° clase</td> <td>Presentaciones orales (2° sesión).</td> </tr> </tbody> </table>	Fecha	Clase	Tema	20/03/2021	1° clase	Inducción a la asignatura.	27/03/2021	2° clase	Conceptos e historia del soft libre.	10/04/2021	3° clase	Propiedad intelectual. Licencias software.	17/04/2021	4° clase	Gestión de proyectos de SL.	24/04/2021	5° clase	Modelos de negocios con SL.	08/05/2021	6° clase	Sistemas operativos de SL.	15/05/2021	7° clase	Clase de repaso y consulta.	29/05/2021	8° clase	Diseño e implementación de SL.	05/06/2021	9° clase	SL en otros ámbitos.	12/06/2021	10° clase	Clase de cierre y consulta.	19/06/2021	11° clase	Presentaciones orales (1° sesión).	26/06/19	12° clase	Presentaciones orales (2° sesión).
Fecha	Clase	Tema																																						
20/03/2021	1° clase	Inducción a la asignatura.																																						
27/03/2021	2° clase	Conceptos e historia del soft libre.																																						
10/04/2021	3° clase	Propiedad intelectual. Licencias software.																																						
17/04/2021	4° clase	Gestión de proyectos de SL.																																						
24/04/2021	5° clase	Modelos de negocios con SL.																																						
08/05/2021	6° clase	Sistemas operativos de SL.																																						
15/05/2021	7° clase	Clase de repaso y consulta.																																						
29/05/2021	8° clase	Diseño e implementación de SL.																																						
05/06/2021	9° clase	SL en otros ámbitos.																																						
12/06/2021	10° clase	Clase de cierre y consulta.																																						
19/06/2021	11° clase	Presentaciones orales (1° sesión).																																						
26/06/19	12° clase	Presentaciones orales (2° sesión).																																						
<p>Propuesta para la atención de consultas y mail de contacto.</p>	<p>El estudiante podrá realizar consultas a través del sistema de chat del Aula Virtual o por medio de las casillas de email:</p> <ul style="list-style-type: none"> • rmedel@frc.utn.edu.ar • ricardo.h.medel@gmail.com 																																							
<p>Plan de integración con otras asignaturas</p>	<p>N/A</p>																																							
<p>Bibliografía Obligatoria</p>	<ul style="list-style-type: none"> • Gomez Gomez, V.M. (2005) Aspectos Históricos del Software Libre en América Latina. 																																							

	<ul style="list-style-type: none"> • González Barahona, J., Seoane Pascual, J., Robles, G. (2007) Introducción al software libre. Fundació per a la Universitat Oberta de Catalunya. ISBN: 84-9788-028-5 • Medel, R. y otros (2018) Manual de la Asignatura Ingeniería de Software de Fuentes Abiertas/Libre. • St. Laurent, A.M. (2004) Understanding Open Source and Free Software Licensing. O'Reilly. ISBN: 0-596-00581-4 • Fogel, K. (2007) Producir software de código abierto: Cómo llevar a buen puerto un proyecto de código libre. O'Reilly. • Krishnamurthy, S. (2003) An Analysis of Open Source Business Models. University of Washington. • Kroah-Hartman, G. (2006) Linux Kernel in a Nutshell. O'Reilly Media. ISBN : 0-596-10079-5 • Raymond, E.S. (1997) La catedral y el bazar. O'Reilly. • Ferreira, C., Souza, C., Pinto, G., Steinmacher, I., Meirelles, P. (2018) When Students Become Contributors: Leveraging OSS Contributions in Software Engineering Courses. Proceedings of XXXII BRAZILIAN SYMPOSIUM ON SOFTWARE ENGINEERING, Sao Carlos, Brazil, September 17–21, 2018 (SBES 2018). https://doi.org/10.1145/3266237.3266250 • Murphy, C., Buffardi, K., Dehlinge, J., Lambert, L., Veilleux, N. (2017) Community Engagement with Free and Open Source Software. SIGCSE'17, March 8–11, 2017, Seattle, WA, USA. DOI: http://dx.doi.org/10.1145/3017680.3017682 • Stallman, R.M. (2001) Free Software, Free Society: Selected Essays of Richard M. Stallman. Free Software Foundation. • Zúñiga, L. (2006) Voces libres de los campos digitales: Una investigación social sobre el Software Libre en América Latina y el Caribe. ISBN: 9977-12-888-X • Martin, R.C. (2008) Clean Code. ISBN-10: 0132350882
<p>Bibliografía Complementaria</p>	<ul style="list-style-type: none"> • Manresa, R.G. (compilador) (2020) La odisea del software libre. • Red Hat (2021) The State of Enterprise Open Source. • Williams, S. (2002) Free as in Freedom: Richard Stallman's Crusade for Free Software. O'Reilly. • Zanotti, A. (2011) Reescribiendo Tecnologías: Aproximaciones al movimiento del software libre y su difusión en Argentina. • Raymond, E.S. (1997) La catedral y el bazar. O'Reilly. • Wheeler, D.A. (2014) Make Your Open Source Software GPL-Compatible. Or Else. • Hernández Gómez, L. (2011) Herramientas software para el trabajo científico colaborativo. Trabajo de Fin de Máster, Universidad de Salamanca. • Collins-Sussman, B., Fitzpatrick, B.W., Pilato, C.. (2004) Control de versiones con Subversion. O'Reilly. • Loeliger, J., McCulloug, M. (2012) Version Control with Git, 2nd Edition. O'Reilly. • O'Sullivan, B. (2009) Mercurial: The Definitive Guide. O'Reilly. • Berg, A. (2012) Jenkins Continuous Integration Cookbook. Packt Publishing. • Black, R. (2011) Managing the Testing Process, 3rd Edition. Wiley.

- Nelson-Smith, S. (2013) Test-driven Infrastructure with Chef, 2nd Edition. O'Reilly.
- Goikolea, M. (2014) Modelos de negocios en Software Libre: Un nuevo paradigma. Innovation & Entrepreneurship Business School.
- López, J.M. (2017) UNIX: el padre de los sistemas operativos actuales. Blogthinkbig.com.
- Isaac (2019) Una historia de UNIX: historia de Unix. ArchiTecnologia.es.
- Diéguez, G., Gasparín, J.M., Sánchez, J., Schejtman, L. (2015) Escenarios y perspectivas del gobierno electrónico en América Latina y el Caribe. Documento de trabajo N° 132. CIPPEC.
- da Rosa, F.; Heinz, F. (2007) Guía práctica sobre software libre: su selección y aplicación local en América Latina y el Caribe. UNESCO. ISBN: 92-9089-103-3
- Benítez-Eyzaguirre, L. (2019) Ciberfeminismo y apropiación tecnológica en América Latina. Virtualis - Revista de Cultura Digital. Vol. 10, Núm. 18.
- Zúñiga, L., Xhardez, V. y otros (2006) Software Libre en América Latina y el Caribe. Documento base: Marco conceptual y metodológico. Análisis de datos cualitativos. Primer año: septiembre 2003 a septiembre 2004. Bellanet Internacional. Oficina para América Latina y el Caribe.

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>sábados de 8 a 10:30 hs.</i>	<i>Ricardo Medel</i>	<i>N/A</i>	<i>N/A</i>

Firma: