



Asignatura	INGENIERIA DE SOFTWARE	
Ciclo Lectivo	2016	
Vigencia del programa	Ciclo lectivo 2016 – Segundo Cuatrimestre Primer Cuatrimestre (Contra-cuatrimestre) Ciclo Lectivo 2017	
Plan	2008	
Nivel	<input type="checkbox"/> 1er. Nivel <input type="checkbox"/> 2do. Nivel <input type="checkbox"/> 3er. Nivel <input checked="" type="checkbox"/> 4to. Nivel <input type="checkbox"/> 5to. Nivel	
Coordinador/ Director de la Cátedra	Ing. Judith Meles	
Área	<input type="checkbox"/> Programación <input type="checkbox"/> Computación <input checked="" type="checkbox"/> Sistemas de Información <input type="checkbox"/> Gestión Ingenieril <input type="checkbox"/> Modelos <input type="checkbox"/> Complementaria	
Carga horaria semanal	6 horas	
Anual/ cuatrimestral	Cuatrimestral	
Contenidos Mínimos (según Diseño Curricular- Ordenanza 1150)	⇒ Componentes de un proyecto de Sistemas de Información. ⇒ Gestión de Configuración de Software. ⇒ Modelos de Calidad de Software. Aseguramiento de la Calidad. ⇒ Métricas de Software. ⇒ Auditoría y Peritaje	
Correlativas para Cursarla	Regulares	Aprobadas
	<ul style="list-style-type: none">• Probabilidad y Estadística• Diseño• Gestión de Datos	<ul style="list-style-type: none">• Análisis de Sistemas• Sintaxis y Semántica del Lenguaje• Paradigma de Programación
Correlativas para Rendirla	Regulares	Aprobadas
		<ul style="list-style-type: none">• Probabilidad y Estadística• Diseño• Gestión de Datos
Objetivos de la Asignatura	⇒ Reconocer la importancia de los conceptos relacionados con la Ingeniería de Software y sus técnicas y herramientas relacionadas. ⇒ Identificar los procesos de desarrollo y los modelos de procesos más adecuados para el desarrollo de software en cada situación particular. ⇒ Introducir el uso de métodos ágiles para el desarrollo y la gestión de proyectos de software. ⇒ Conocer los componentes de un proyecto de ingeniería de software.	



	<ul style="list-style-type: none">⇒ Conocer los estándares asociados a la calidad del proceso de desarrollo de software y de los productos de software.⇒ Conocer los componentes de los planes de aseguramiento de la calidad y de los planes de prueba.⇒ Presentar la disciplina de Gestión de Configuración y su importancia para el desarrollo de software.⇒ Emplear métricas que se aplican al desarrollo de software.⇒ Aplicar los elementos de un proceso de prueba ("testing") como parte integral del Aseguramiento de Calidad del producto.⇒ Integrar por medio de casos prácticos concretos los conocimientos adquiridos en la parte teórica, empleando así las técnicas y herramientas de aplicación de la ingeniería de software.
--	--

Programa Analítico

Unidad Nro. 1: Ingeniería de Software en Contexto

Objetivos específicos:

- ☐ Introducir los componentes de un proyecto de desarrollo de software.
- ☐ Comprender los motivos que ocasionaron la llamada "crisis del software".
- ☐ Comprender la utilidad de los ciclos de vida en el contexto del desarrollo de software y su influencia en la Administración de Proyectos de Software.

Contenidos:

- ☐ Introducción a la Ingeniería del Software. ¿Qué es?
- ☐ Estado Actual y Antecedentes. La Crisis del Software.
- ☐ Disciplinas que conforman la Ingeniería de Software.
- ☐ Ejemplos de grandes proyectos de software fallidos y exitosos.
- ☐ Ciclos de vida (Modelos de Proceso) y su influencia en la Administración de Proyectos de Software.
- ☐ Procesos de Desarrollo Empíricos vs. Definidos.
- ☐ Ciclos de vida (Modelos de Proceso) y Procesos de Desarrollo de Software
- ☐ Ventajas y desventajas de c/u de los ciclos de vida. Criterios para elección de ciclos de vida en función de las necesidades del proyecto y las características del producto.

Bibliografía:

- ☐ **Sommerville, Ian** - INGENIERÍA DE SOFTWARE - Novena Edición (Editorial Addison-Wesley Año 2011). **Capítulo 1**
- ☐ **Pressman, Roger** - INGENIERÍA DE SOFTWARE, UN ENFOQUE PRÁCTICO. Séptima Edición - Editorial McGraw Hill – Año 2010. **Capítulo 1**
- ☐ **Steve Mc Connell.**, DESARROLLO Y GESTIÓN DE PROYECTOS INFORMÁTICOS (Editorial McGraw Hill – Año 1996). Capítulo 7.



- ☐ SEBOK V3.0 (Software Engineering Body of Knowledge)- IEEE 2014
- ☐ **Papers:**
- ☐ **Orphans Preferred** (<http://www.stevemcconnell.com/psd/07-orphanspreferred.htm>)
- ☐ **No Silver Bullet**
(<http://www.virtualschool.edu/mon/SoftwareEngineering/BrooksNoSilverBullet.html>)
- ☐ **Software's Ten Essentials**
(<http://www.stevemcconnell.com/ieeesoftware/10Essentials.pdf>)
- ☐ **Royce, Winston** – Managing the development of large systems – IEEE Wescon, Agosto 1970.
- ☐ <http://martinfowler.com/articles/newMethodology.html> **Fowler, Martin** – The new methodology

Evaluación:

La evaluación de contenidos se hará en el primer parcial. También se evaluará a los estudiantes con una actividad de investigación y exposición oral.

Unidad Nro. 2: Proyectos de Desarrollo de Software basado en procesos definidos

Objetivos específicos:

- ☐ Introducir los componentes de un proyecto de desarrollo de software.
- ☐ Valorar la relación existente entre el Proceso, el Proyecto y el Producto de Software a construir.
- ☐ Introducir los conceptos fundamentales a cerca de la problemática de administrar proyectos de software basado en procesos definidos.
- ☐ Saber utilizar las técnicas de medición y estimación de software.
- ☐ Comprender la importancia de medir para obtener visibilidad de los proyectos de desarrollo de software.
- ☐ Reconocer la importancia de las métricas para la mejora de procesos, proyectos y productos.

Contenidos

- ☐ Componentes de un Proyecto de Sistemas de Información.
- ☐ Vinculo proceso-proyecto-producto en la gestión de un proyecto de desarrollo de software.
- ☐ Estimación de software, técnicas de estimación de software basadas en procesos definidos
- ☐ Métricas, Medidas e Indicadores
- ☐ Métricas de Producto, Proyecto y Proceso.
- ☐ Aplicación de las métricas en Administración de Proyectos.
- ☐ Monitoreo y Control de Proyectos

Bibliografía:

- ☐ **Sommerville, Ian** - INGENIERÍA DE SOFTWARE - Novena Edición (Editorial Addison-Wesley Año 2011). **Capítulo 22**
- ☐ **Pressman, Roger** - INGENIERÍA DE SOFTWARE, UN ENFOQUE PRÁCTICO. Séptima Edición - Editorial McGraw Hill – Año 2010. **Capítulo 21 al 25**
- ☐ **Brooks, Frederick** -THE MYTHICAL MAN-MONTH (ANNIVERSARY ED.), 1995 Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA ©1995 Capítulos 1 al 3



- ☐ **McConnell, Steve**, Software Estimation: Demystifying the Black Art (Editorial Microsoft Press – Año 2006) Capítulo 1 al 4

Evaluación:

La evaluación de contenidos se hará en el primer parcial como en los ejercicios prácticos de resolución en el aula integrando los contenidos de toda la materia y de materias anteriores de la carrera.

Unidad Nro. 3: Gestión Ágil de Proyectos

Objetivos específicos:

- ☐ Presentar conceptos relacionados a las metodologías ágiles en general y a una metodología de gestión ágil de proyectos de software en particular.
- ☐ Introducir la filosofía y los principios de las métricas en ambientes ágiles.
- ☐ Analizar comparativamente los enfoques de gestión tradicionales basados en procesos definidos con los enfoques de gestión basados en procesos empíricos.
- ☐ Desarrollar capacidades que permitan gestionar proyectos con frameworks Ágiles
- ☐ Introducir el manejo ágil de Requerimientos.
- ☐ Crear user stories, una técnica para identificación de requerimientos de usuario.

Contenidos

- ☐ Manifiesto Ágil.
- ☐ Introducción al Desarrollo Ágil.
- ☐ Requerimientos en ambientes ágiles - User Stories
- ☐ Enfoque Ágil de Gestión de Proyectos con SCRUM
- ☐ Métricas Ágiles

Bibliografía:

- ☐ **Cohn, Mike** – Agile Estimation and Planning – Editorial Prentice Hall 2006 – Capítulo 16
- ☐ <http://www.scrumguides.org/download.html>
- ☐ <http://www.romanpichler.com/blog/grooming-the-product-backlog/>
- ☐ <http://guide.agilealliance.org/guide/backlog-grooming.html>
- ☐ **Dean Leffingwell and Pete Behrens** – A user story primer (2009)
- ☐ **Manifiesto Ágil** <http://agilemanifesto.org/iso/es/>
- ☐ <http://people10.com/blog/software-sizing-for-agile-transformation>

Evaluación:

La evaluación de contenidos se hará en el primer parcial como en los ejercicios prácticos de resolución en el aula integrando los contenidos de toda la materia y de materias anteriores de la carrera.

Unidad Nro. 4: Gestión de Configuración del Software



Objetivos específicos:

- ☐ Reconocer la importancia de la Gestión de Configuración de Software.
- ☐ Describir las actividades de planificación de la Gestión de Configuración de Software.
- ☐ Conocer los componentes de un Plan de Gestión de Configuración de Software.
- ☐ Discutir el uso de algunas herramientas utilizadas para la Gestión de Configuración de Software.

Contenidos.

- ☐ Conceptos Introductorias de la Gestión de Configuración.
- ☐ Versiones, variantes, release.
- ☐ Planificación de la Gestión de Configuración de Software.
- ☐ Actividades relacionadas con la Gestión de Configuración.
- ☐ El rol de las líneas base y su administración.
- ☐ Elementos de configuración del Software.
- ☐ Identificación de Objetos en la Configuración de Software.
- ☐ Gestión de Configuración en ambientes ágiles

Bibliografía:

- ☐ **Bersoff, Edgard** – Elements of Software Configuration Management
- ☐ **Software Program Manager Network** - The Little Book of Software Configuration Management, (AirLie Software Council, 1998)
- ☐ **Pressman, Roger** - INGENIERÍA DE SOFTWARE, UN ENFOQUE PRÁCTICO. Séptima Edición - Editorial McGraw Hill – Año 2010. **Capítulo 22**
- ☐ **Sommerville, Ian** - INGENIERÍA DE SOFTWARE - Novena Edición (Editorial Addison-Wesley Año 2011). **Capítulo 27**
- ☐ http://www.scmpatterns.com/pubs/hass_sidebar.html - Agile SCM
- ☐ <http://www.scmpatterns.com/pubs/crossroads-mirror/agileoct03.pdf>
- ☐ <https://www.cmcrossroads.com/article/defining-agile-scm-past-present-future-2008?page=0%2C1>

Evaluación:

La evaluación de contenidos se hará en el segundo parcial como en los ejercicios prácticos de resolución en el aula en los que se integrarán los contenidos de toda la materia y de materias anteriores de la carrera

Unidad Nro. 5: Aseguramiento de Calidad de Proceso y de Producto

Objetivos específicos:

- ☐ Identificar las principales tendencias respecto a la calidad y su forma de incorporarla.
- ☐ Conocer la importancia y la influencia que tienen el trabajo para y con calidad en las organizaciones.
- ☐ Analizar críticamente los principales modelos de calidad de software existentes en el mercado.
- ☐ Identificar técnicas y herramientas para realizar aseguramiento de calidad de software.
- ☐ Descubrir la importancia de la prueba del sistema como una herramienta que controlará la calidad del producto construido.



- ☐ Reconocer y utilizar las diferentes técnicas (auditorías, revisión e inspecciones de software) relacionadas con el aseguramiento de la calidad del proceso y del producto.
- ☐ Conocer técnicas y herramientas para realizar pruebas e inspecciones de software.
- ☐ Poder planificar actividades relacionadas al aseguramiento de calidad de software e insertarlas en el proyecto.

Contenidos

- ☐ Conceptos generales sobre calidad.
- ☐ Importancia de trabajar para y con Calidad. Ventajas y Desventajas.
- ☐ Actividades relacionadas con el Aseguramiento de la Calidad del Software.
- ☐ Principales Modelos de Calidad existentes (CMMI – SPICE – ISO) y sus métodos de evaluación.
- ☐ Lineamientos para la implementación de modelos de calidad en las organizaciones.
- ☐ Calidad de Producto: Planificación de pruebas para el software- Niveles y tipos de pruebas para el software. Técnicas y herramientas para probar software. Técnicas y Herramientas para la realización de revisiones técnicas del software.
- ☐ Testing en ambientes Ágiles.
- ☐ Diferentes tipos de Auditorías: Auditorías de Proyecto y Auditorías al Grupo de Calidad.
- ☐ Proceso de Auditorías: Responsabilidades. Preparación y ejecución. Reporte y seguimiento.

Bibliografía:

- ☐ **Sommerville, Ian** - INGENIERÍA DE SOFTWARE - Novena Edición (Editorial Addison-Wesley Año 2002). **Capítulo 24 y 26**
- ☐ **Pressman, Roger** - INGENIERÍA DE SOFTWARE, UN ENFOQUE PRÁCTICO. Séptima Edición - Editorial McGraw Hill – Año 2010. **Capítulo 14 al 16**
- ☐ **Myers, Glenford**- El arte de Probar el Software. (Editorial El Ateneo, 1983).- **Capítulos 2 al 6**
- ☐ IEEE STD 1028-1997 STANDARD FOR SOFTWARE REVIEWS
- ☐ IEEE STD 1012-1998 (REVISION OF IEEE STD 1012-1986) IEEE STANDARD FOR SOFTWARE VERIFICATION AND VALIDATION
- ☐ [HTTP://TESTOBSESSED.COM/WP-CONTENT/UPLOADS/2011/04/AGILETESTINGOVERVIEW.PDF](http://testobsessed.com/wp-content/uploads/2011/04/agile-testing-overview.pdf)
- ☐ [HTTP://WWW.AMBYSOFT.COM/ESSAYS/AGILETESTING.HTML](http://www.ambysoft.com/essays/agile-testing.html)

Evaluación:

La evaluación de contenidos se hará en el segundo parcial como en los ejercicios prácticos de resolución en el aula y del desarrollo de un Trabajo Práctico en el que se integrarán los contenidos de toda la materia y de materias anteriores de la carrera.



Metodología de enseñanza y aprendizaje	<p>Destacando el hecho que el currículo no solo se manifiesta en la especificación de una serie de contenidos en un programa, sino por el contrario, abarca cuestiones mucho más profundas tales como: bibliografía, priorización de algunos contenidos sobre otros, proceso de enseñanza – aprendizaje, formas de evaluación, entre otras; es que se considera importante poner de manifiesto algunos de estos aspectos con el propósito de mejorar el nivel académico y fomentar la integración de la cátedra, sin interferir, por supuesto, en la libertad de cada uno de los docentes que la integren.</p> <p>La selección de los contenidos incluidos en el programa se realizó considerando la integración de esta nueva asignatura al resto de las asignaturas de la carrera, lo que fundamenta en gran medida la priorización y el nivel de profundidad elegido para cada tema.</p> <p>Dentro de las cuestiones que se expondrán para el desarrollo de la Metodología se tendrán en cuenta los siguientes aspectos:</p> <p>Dictado de la materia: el contenido temático está organizado lógicamente y situado coherentemente según su grado de dificultad de manera que permita al estudiante ir asimilando los contenidos propios de la materia en forma gradual y a la vez integrar los contenidos de otras asignaturas.</p> <p>Para el desarrollo del programa se considerará un sistema de clases que combine: clases explicativas, clases prácticas tipo taller, la realización de ejercicios prácticos grupales y el uso del laboratorio. A través de estas diferentes formas organizativas de la enseñanza se proponen los siguientes objetivos educativos:</p> <ul style="list-style-type: none">⇒ Transmitir los conocimientos a través de un proceso de enseñanza-aprendizaje que permita la apropiación de los contenidos.⇒ Desarrollar el hábito de la lectura, el análisis y la interpretación de textos, invitando a los alumnos a trabajar con las fuentes bibliográficas originales, posibilitando que elaboren sus propias interpretaciones y realicen sus propias conclusiones.⇒ Promover el espíritu investigativo para buscar siempre la verdad auténtica y la rigurosidad de la ciencia en la búsqueda de las soluciones a las situaciones de aprendizaje que se propongan.⇒ Valorar el uso de bibliografía como fuente original de los conceptos desarrollados en la asignatura.⇒ Fomentar la habilidad para aplicar los conocimientos adquiridos a situaciones concretas.
---	---



Sistema de evaluación	<p>Para obtener la aprobación de la asignatura se proponen diferentes actividades, algunas para desarrollar individualmente y otras para desarrollar en forma grupal.</p> <p><u>Condiciones de Regularidad</u></p> <p>Nivel Grupal:</p> <p>1. <i>Ejercicios Prácticos</i>: en cada una de las unidades se le planteará a los estudiantes la realización de uno o más ejercicios de este tipo, cuyo objetivo es lograr la apropiación de los contenidos abordados en cada unidad. Debe presentar el 100 % de los trabajos prácticos y aprobar el 70 % de los mismos como mínimo. En caso de no alcanzar el porcentaje mínimo de aprobación, cada uno de los integrantes del grupo quedará en condición de <i>libre</i>.</p>
Condiciones de regularidad	<p>Además de lo descripto en el apartado anterior respecto de los Ejercicios Prácticos, para obtener la regularidad el estudiante deberá rendir y aprobar 2 (dos) parciales teórico-prácticos, teniendo la posibilidad de realizar 1 (un) recuperatorio al final del cuatrimestre. La aprobación se obtiene con nota mayor o igual a 4 (cuatro), tanto en los parciales como en los Recuperatorios. La nota 4 (cuatro) se corresponde con el 60 % de los contenidos evaluados en el parcial.</p> <p>Debe tener al menos un 60 % del puntaje asignado a cada parte (teórico-práctico) además de cumplir con el puntaje mínimo, para que el parcial se considere aprobado.</p> <p>Tanto los parciales como los Recuperatorios se toman de forma unificada para todos los cursos que conforman la cátedra.</p> <p>Respecto a los recuperatorios, si el estudiante tiene 45 puntos o más en la parte práctica, quedará eximido de hacer esa parte en el recuperatorio, debiendo recuperar únicamente la parte teórica.</p>
Condiciones, duración y características de la promoción (si corresponde)	No corresponde
Modalidad de examen final	<p>Para obtener la aprobación de la materia el estudiante deberá aprobar el examen final en el que se evaluarán todos los contenidos del <i>último programa vigente</i> para la asignatura. Se evaluarán aspectos teóricos de la materia.</p> <p>El examen final se aprueba con nota mínima no menor a 4 (cuatro), correspondiendo también al 60 % de los contenidos evaluados, al igual que en los exámenes parciales.</p> <p><u>Importante:</u></p> <p>La cátedra tomará los exámenes finales en forma conjunta para todos los alumnos, esto permitirá la nivelación e integración de todos los cursos que la conforman.</p>



Actividades en laboratorio	Práctica deseable: Utilización de un software para gestión de configuración, para métricas y testing.				
Horas/año totales de la asignatura (horas cátedra)	96 horas				
Cantidad de horas prácticas totales (horas cátedra)	La carga horaria afectada a la formación práctica se corresponde con el 50 % de las horas totales de la asignatura, es decir 48 horas. De este 50 % de carga horaria al menos el 15 % se empleará específicamente en la Resolución de Problemas de Ingeniería. Este 15 % de tiempo se traduce en términos de clases en 5 clases de 3 módulos cada una.				
Cantidad de horas teóricas totales (horas cátedra)	48 horas				
Tipo de formación práctica (marque la que corresponde y si es asignatura curricular -no electiva-)	<input type="checkbox"/> Formación experimental <input checked="" 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 afectadas a la formación práctica indicada (horas cátedra)	La carga horaria afectada a la formación práctica se corresponde con el 50 % de las horas totales de la asignatura. De este 50 % de carga horaria al menos el 15 % se empleará específicamente en la Resolución de Problemas de Ingeniería. Este 15 % de tiempo se traduce en términos de clases en 5 clases de 3 módulos cada una.				
Descripción de los prácticos	La parte práctica de la materia está basada en el desarrollo de ejercicios prácticos de resolución de problemas de ingeniería. Sobre estos ejercicios prácticos se trabajará cada uno de los contenidos prácticos que se desarrollan en la materia. Adicionalmente se entregará a los alumnos Ejercicios Prácticos resueltos, sobre los que se explicará en clase la forma de resolución de los mismos.				
Criterios de evaluación	<p>Sobre los prácticos que la cátedra utilizará se evaluarán los siguientes aspectos, vinculados al cumplimiento de los objetivos de la asignatura:</p> <ol style="list-style-type: none"> 1. Trabajo acorde a las consignas presentadas 2. Que resuelva correctamente el problema que el proyecto presenta y cumpla los objetivos definidos para éste 3. Consistencia de cada uno de los modelos que se van desarrollando 4. Aspecto formal de la presentación del práctico 5. Cumplimiento de la fecha acordada 6. Integración del grupo en la realización del trabajo <p>La escala de notas utilizada para la corrección de parciales y parte práctica de los exámenes finales se corresponde con la siguiente escala:</p> <table border="1"> <tr> <td>Puntajes sobre 100 puntos</td><td>Nota</td></tr> <tr> <td>Hasta 19 puntos</td><td>1 (uno)</td></tr> </table>	Puntajes sobre 100 puntos	Nota	Hasta 19 puntos	1 (uno)
Puntajes sobre 100 puntos	Nota				
Hasta 19 puntos	1 (uno)				



		20 – 59 puntos	2 (dos)		
		60 - 64 puntos	4 (cuatro)		
		65- 69 puntos	5 (cinco)		
		70 -76 puntos	6 (seis)		
		77- 82 puntos	7 (siete)		
		83- 89 puntos	8 (ocho)		
		90 -95 puntos	9 (nueve)		
		96 – 100 puntos	10 (diez)		
Descripción de presentación de los prácticos (aspectos formales)	Cada uno de los Ejercicios Prácticos de Aplicación estará descripto considerando los siguientes aspectos: 1. Tema que cubre el Ejercicio Práctico 2. Objetivo del Ejercicio. 3. Propósito del Ejercicio 4. Entradas requeridas para su ejecución 5. Salidas esperadas 6. Consigna asociada Ejercicio Práctico 7. Instrucciones, si correspondiera.				
Cronograma de actividades de la asignatura, contemplando las fechas del calendario 2016, 2do. Cuatrimestre y para cada unidad.	Semana	Clase 1	Clase 2		
	01/08	Presentación de la materia, de los docentes- U1 Teórico: Introducción a la Ingeniería de Software	U1 Teórico Ciclo de vida y procesos Procesos Definidos y Empíricos		
	08/08	U2: Teórico de Componentes de un proyecto de software Estimaciones Tradicionales	U2: Teórico Métricas Tradicionales		
	15/08	Práctico de Métricas Tradicionales	U3: Teórico de Métodos Ágiles - Manifiesto Ágil		
	22/08	Práctico de Métricas Tradicionales	U3: Teórico de Requerimientos Ágiles		
	29/08	Práctico de User Stories	U3:Teórico de Scrum		
	05/09	Práctico de User Stories	U3: Teórico de Scrum- Monitoreo y Métricas Ágiles		
	12/09	Práctico de Scrum- Práctico de Estimaciones con Poker Planning	Práctico de Métricas		
	19/09	Ejercicio Práctico tipo parcial	Clase de Consulta		
	Primer Parcial 24/09				
	26/09	U4 Teórico: Administración de Configuración de Software Práctico de CM	Teórico CMMi - Foco en nivel II		
	03/10	Retrospectiva Entrega parcial	Práctico de SCM		
	10/10	Teórico Práctico de Calidad de Producto: Testing de caja negra: Festival de folklore	Teórico de Testing: Proceso de Testing		
	17/10	Teórico Práctico de Calidad de Producto - Testing caja blanca: Festival del folklore	Teórico Práctico de Calidad de Producto - Testing caja blanca: Festival del folklore		
	24/10	Práctico de Testing	Teórico de Auditorías de Software		
	01/11	Ejercicio Práctico tipo parcial	Clase de Consulta		
	Segundo Parcial 05/11				
	07/11	Actividad de Cátedra a Definir			
	14/11	Regularización y Recuperatorios de parciales			
	Propuesta para la atención de consultas y mail de contacto	La cátedra tiene planificada una clase de consulta la clase previa a cada uno de los parciales, inclusive los recuperatorios. Los horarios de consulta deberán convenirlos en cada curso con sus docentes.			



	A todo efecto la cátedra dispone de la siguiente dirección de correo para que los alumnos se puedan comunicar directamente con el Coordinador de Cátedra: jmeles@gmail.com .
Plan de integración con otras asignaturas	<p>Esta materia utilizará en gran medida lo aprendido en ASI (Análisis de Sistemas) en el segundo nivel, en DSI (Diseño de Sistemas), directa continuadora de los contenidos de la materia ASI, en Gestión de Datos y las materias del área de programación, dado que se asume que el alumno ya maneja las herramientas y técnicas necesarias para construir un software. En este sentido, el foco principal de la Asignatura Ingeniería de Software es fortalecer las disciplinas denominadas “protectoras”, que transversalmente van apoyando el desarrollo del producto desde el inicio hasta el final, me refiero a las disciplinas de Gestión de Configuración, Aseguramiento de Calidad, Verificación y Validación y Administración de Proyectos.</p> <p>Además, se prevé la integración con (ARE) Administración de Recursos, que complementará a Ingeniería de Software con el dictado de contenidos vinculados a Auditoría Informática y Peritaje.</p>



Bibliografía Obligatoria	<ul style="list-style-type: none">❖ Sommerville, Ian - INGENIERÍA DE SOFTWARE - Novena Edición (Editorial Addison-Wesley Año 2011).❖ Steve Mc Connell., DESARROLLO Y GESTIÓN DE PROYECTOS INFORMÁTICOS (Editorial McGraw Hill – Año 1996).❖ Pressman, Roger - INGENIERÍA DE SOFTWARE, UN ENFOQUE PRÁCTICO. –(Sexta Edición -Editorial McGraw Hill – Año 2005)❖ Myers, Glenford- El arte de Probar el Software. (Editorial El Ateneo, 1983).- <p>Otras fuentes:</p> <ul style="list-style-type: none">❖ Orphans Preferred (http://www.stevemcconnell.com/psd/07-orphanspreferred.htm)❖ No Silver Bullet (http://www.virtualschool.edu/mon/SoftwareEngineering/BrooksNoSilverBullet.html)❖ Software's Ten Essentials (http://www.stevemcconnell.com/ieeesoftware/10Essentials.pdf)❖ http://www.scrumguides.org/download.html❖ Dean Leffingwell and Pete Behrens – A user story primer (2009)❖ Manifiesto Ágil http://agilemanifesto.org/iso/es/❖ http://pgpubu.blogspot.com.ar/2007/01/tecnica-de-estimacin-wideband-delphi.html❖ http://people10.com/blog/software-sizing-for-agile-transformation❖ Bersoff, Edgard – Elements of Software Configuration Management – Sitio: http://portal.acm.org❖ Software Program Manager Network - The Little Book of Software Configuration Management, (AirLie Software Council, 1998)- Sitio: http://www.spmn.com
---------------------------------	--



Bibliografía Complementaria	<ul style="list-style-type: none">❖ Cohn, Mike – Agile Estimation and Planning – Editorial Prentice Hall 2006 – Capítulos 4, 6 y 7.❖ McConnell, Steve, Software Estimation: Demystifying the Black Art (Editorial Microsoft Press – Año 2006).❖ Brooks, Frederick The mythical man-month (anniversary ed.), 1995 Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA ©1995❖ CMMI para Desarrollo en Español: http://cmmiinstitute.com/assets/Spanish%20Technical%20Report%20CMMI%20V%201%203.pdf❖ SPICE PROJECT, CONSOLIDATED PRODUCT. SOFTWARE PROCESS ASSESSMENT – PART 1: CONCEPTS AND INTRODUCTORY GUIDE. VERSION 1.00. Site de SPICE: www.esi.es/Projects/SPICE❖ McFeeley, Bob - IDEAL: A USER GUIDE FOR SOFTWARE PROCESS IMPROVEMENT – CMU/SEI-96-HB-001. WWW.SEI.CMU.EDU❖ Sitio de la IEEE: http://www.ieee.org❖ IEEE STD 730 STANDARD FOR SOFTWARE QUALITY ASSURANCE PLANS❖ IEEE STD 1028-1997 STANDARD FOR SOFTWARE REVIEWS❖ IEEE STD 1012-1998 (REVISION OF IEEE STD 1012-1986) IEEE STANDARD FOR SOFTWARE VERIFICATION AND VALIDATION❖ Cohn, Mike - User Stories Applied – Editorial Addison Wesley 2004❖ http://www.infoq.com/articles/roadmap-agile-documentation❖ http://www.romanpichler.com/blog/grooming-the-product-backlog/❖ http://guide.agilealliance.org/guide/backlog-grooming.html❖ Royce, Winston – Managing the development of large systems – IEEE Wescon, Agosto 1970.❖ http://martinfowler.com/articles/newMethodology.html Fowler, Martin – The new methodology❖ SEBOK V3.0 (Software Engineering Body of Knowledge)- IEEE 2014																														
Distribución de docentes por curso	<table><tr><th>Curso</th><th>Día y Horas</th><th>Turno</th><th>Profesor</th><th>J.T.P.</th><th>Ayudantes</th></tr><tr><td>4K1</td><td>Mar 3-4-5-6 Jue 4-5</td><td>M</td><td>Meles, Judith</td><td>Massano, Cecilia</td><td>Robles, Joaquín</td></tr><tr><td>4K2</td><td>Mar 1-2-3-4 Vie 3-4</td><td>T</td><td>Meles, Judith</td><td>Massano, Cecilia</td><td></td></tr><tr><td>4K3</td><td>Mie 3-4-5-6 Vie 5-6</td><td>N</td><td>Battistelli, Daniel</td><td>Covaro, Laura</td><td></td></tr><tr><td>4K4</td><td>Mar 3-4-5-6 Vie 1-2</td><td>N</td><td>Battistelli, Daniel</td><td>Covaro, Laura</td><td>Robles, Joaquín</td></tr></table>	Curso	Día y Horas	Turno	Profesor	J.T.P.	Ayudantes	4K1	Mar 3-4-5-6 Jue 4-5	M	Meles, Judith	Massano, Cecilia	Robles, Joaquín	4K2	Mar 1-2-3-4 Vie 3-4	T	Meles, Judith	Massano, Cecilia		4K3	Mie 3-4-5-6 Vie 5-6	N	Battistelli, Daniel	Covaro, Laura		4K4	Mar 3-4-5-6 Vie 1-2	N	Battistelli, Daniel	Covaro, Laura	Robles, Joaquín
Curso	Día y Horas	Turno	Profesor	J.T.P.	Ayudantes																										
4K1	Mar 3-4-5-6 Jue 4-5	M	Meles, Judith	Massano, Cecilia	Robles, Joaquín																										
4K2	Mar 1-2-3-4 Vie 3-4	T	Meles, Judith	Massano, Cecilia																											
4K3	Mie 3-4-5-6 Vie 5-6	N	Battistelli, Daniel	Covaro, Laura																											
4K4	Mar 3-4-5-6 Vie 1-2	N	Battistelli, Daniel	Covaro, Laura	Robles, Joaquín																										

Firma:

Aclaración: