

Carrera: Ingeniería en Sistemas de Información**Asignatura:** Ingeniería y Calidad de Software**Planificación a partir del Ciclo Lectivo 2025**

1. Datos administrativos de la asignatura			
Nivel en la carrera	4	Duración	Cuatrimestral
Plan	2023		
Bloque curricular:	Tecnologías Aplicadas		
Carga horaria presencial semanal (hs. cátedra):	6 horas	Carga Horaria total (hs. reloj):	72
Carga horaria no presencial semanal (hs. reloj) (si correspondiese)	N/A	% horas no presenciales (hs. reloj) (si correspondiese)	N/A

2. Presentación, Fundamentación
<p>Ingeniería y Calidad de Software forma parte del Área de Desarrollo de Software; creada en el marco de la adecuación del plan de Estudios de la carrera de Ingeniería en Sistemas de Información, está situada en el 4to. Nivel y es una asignatura cuatrimestral.</p> <p>La Ingeniería de Software es una disciplina muy amplia cuyos contenidos están distribuidos en varias de las asignaturas de la carrera; del área de la que forma parte, los estudiantes adquieren competencias para poder implementar software. Del área de Sistemas de Información, las asignaturas Sistemas y Procesos de Negocio (Primer Nivel), Análisis de Sistemas de Información (Segundo Nivel) y Diseño de Sistemas de Información (Tercer Nivel), se encargan de desarrollar contenidos de disciplinas denominadas “técnicas”, es decir: el conocimiento del Sistema de Negocio u Organización donde los sistemas de información se van a insertar, cómo determinar necesidades que encuadrarán el problema que debe resolverse, el planteo de una solución a esos problemas que se vea reflejada en modelos arquitectónicos y de diseño. Luego la asignatura de cuarto nivel, Administración de Sistemas de Información, introduce otros aspectos que también integran la ingeniería de software vinculados a la gestión eficiente de los recursos necesarios para la ejecución de un proyecto de software.</p> <p>El rol de Ingeniería y Calidad de Software focaliza en el desarrollo de otras disciplinas denominadas de “Gestión” y de “Soporte” para completar con ello la formación integral del futuro profesional informático. Este bagaje de conocimientos será la entrada a la Asignatura Proyecto en la cual el estudiante tendrá la oportunidad de aplicarlo en una situación real las competencias adquiridas.</p> <p>Además se profundizará en el uso de métodos, técnicas y herramientas eficientes, que posibiliten la construcción de software de “calidad”, que satisfaga las necesidades y expectativas de los diferentes interesados.</p> <p>Es ya bien comprendido que para crear un producto de software que satisfaga las necesidades de los interesados, no es suficiente con “programar”, los aspectos de gestión y soporte vinculados al desarrollo de software son tan importantes como la correcta determinación de los requerimientos, el análisis, el diseño, la verificación y validación.</p>

Ingeniería y Calidad de Software completa la formación del Graduado Tecnológico en estas disciplinas que dan soporte al desarrollo de un sistema de información que cumpla con todas las características de integridad y calidad requeridas. Es necesario destacar que los contenidos aquí abordados tienen vigencia en la realidad laboral actual. De allí la importancia de esta asignatura en la formación del graduado en Ingeniería en Sistemas de Información.

Durante sus primeros años de cursado, el futuro graduado va adquiriendo conocimientos y habilidades de la profesión y también de disciplinas organizacionales, computacionales, matemáticas, etc.. Aquí tiene la posibilidad de abordar la profesión desde otro ángulo, desde una perspectiva diferente que le abrirá las puertas al conocimiento de otros roles importantes involucrados en la construcción de software.

El futuro profesional, deberá desarrollar a lo largo de esta asignatura, habilidades y destrezas necesarias para administrar un proyecto de software y para garantizar la calidad del producto resultante de ese proyecto.

Si recordamos que el perfil que se pretende es el de un profesional informático capaz de ANALIZAR, DISEÑAR, ADMINISTRAR.... Sistemas de Información, entonces, es de vital importancia el reconocimiento de la incidencia de esta materia en la formación de este profesional tecnológico.

Se orientará la asignatura de manera que proporcione al ahora estudiante, futuro graduado tecnológico, todas las competencias necesarias para actuar en el mercado global actual.

En su orientación se incluirán actividades tendientes a fortalecer su capacidad de resolución de problemas, de toma de decisiones y de presentación de soluciones creativas; como así también su capacidad para adaptarse a los cambios que impone el medio, cambios que pueden ser originados por diversos aspectos tecnológicos o de otro tipo.

3. Relación de la asignatura con las competencias de egreso de la carrera

En la tabla siguiente se establece la relación de la asignatura con las competencias de egreso: Específicas, Genéricas Tecnológicas y Genéricas Sociales, Políticas y Actitudinales de la carrera. Se incluyen las competencias de egreso a las que tributa, aportes reales y significativos de la asignatura, y en qué nivel (no aporta, bajo, medio, alto).

Competencias	Nivel
Competencias genéricas tecnológicas (CG):	
CG.1. Identificación, formulación y resolución de problemas de ingeniería en sistemas de información/informática.	Medio
CG.2. Concepción, diseño y desarrollo de proyectos de Ingeniería en Sistemas de Información/Informática	Medio
CG.3. Gestión, planificación, ejecución y control de proyectos de Ingeniería en Sistemas de Información/Informática.	No aporta
CG.4. Utilización de técnicas y herramientas de aplicación de Ingeniería en Sistemas de Información/Informática.	Medio
CG.5. Generación de desarrollos tecnológicos y/o innovaciones tecnológicas.	Medio
Competencias genéricas sociales, políticas y actitudinales (CG)	
CG.6. Fundamentos para el desempeño en equipos de trabajo.	Alto

CG.7. Fundamentos para una comunicación efectiva.	Alto
CG.8. Fundamentos para una actuación profesional ética y responsable.	No aporta
CG.9. Fundamentos para evaluar y actuar en relación con el impacto social de su actividad profesional en el contexto global y local.	No aporta
CG.10. Aprender en forma continua y autónoma.	Medio
CG.11. Fundamentos para el desarrollo de una actitud profesional emprendedora	No aporta
Competencias Específicas de la carrera	
CE1.1. Especificar, proyectar y desarrollar sistemas de información para concebir soluciones tecnológicas que permitan resolver situaciones en las organizaciones mediante el empleo de metodologías de sistemas y tecnologías asociadas a los sistemas de información.	Alto
CE1.2. Especificar, proyectar y desarrollar sistemas de comunicación de datos, evaluando posibles soluciones tecnológicas disponibles para dar soporte a los sistemas de información en lo referido al procesamiento y comunicación de datos.	No aporta
CE1.3. Especificar, proyectar y desarrollar software para la elaboración de soluciones informáticas con el propósito de resolver problemas estratégicos y operativos, así como de servicios y de negocios, en el marco de una actividad económica que sea social y ambientalmente sustentable.	Medio
CE2.1. Proyectar y dirigir lo referido a seguridad informática para seleccionar y aplicar técnicas, herramientas, métodos y normas, garantizando la seguridad y privacidad de la información procesada y generada por los sistemas de información.	No aporta
CE3.1. Establecer métricas y normas de calidad de software para medir, evaluar, controlar y monitorear el rendimiento, impulsando mejoras de acuerdo a técnicas y normas vigentes definidas por los organismos de estandarización.	Medio
CE4.1. Certificar el funcionamiento, condición de uso o estado de sistemas de información, sistemas de comunicación de datos, software, seguridad informática y calidad de software para asegurar la generación de los resultados deseados en función de restricciones de tiempo y recursos establecidos.	Medio
CE5.1. Dirigir y controlar la implementación, operación y mantenimiento de sistemas de información, sistemas de comunicación de datos, software, seguridad informática y calidad de software, a los fines de alcanzar los objetivos fijados por la organización.	Medio

CE.6.1. Asesorar y capacitar a organizaciones, empresas, organismos públicos o privados en la adquisición, instalación y uso, en lo que respecta a sistemas de información, sistemas de comunicación de datos, software, seguridad informática y calidad de software, a los fines de un uso correcto de los sistemas intervenientes.	No aporta
CE.7.1. Realizar pericias, tasaciones y arbitrajes relacionados con su actividad profesional, respetando marcos normativos y jurídicos con el objeto de asesorar a las partes o a los tribunales de Justicia.	No aporta

4. Contenidos Mínimos

Software e Ingeniería de Software
 Disciplinas de la Ingeniería de Software
 Gestión de Configuración de Software
 Modelos de Calidad de Software
 Aseguramiento de Calidad del Producto de Software. Validación y Verificación
 Enfoques en el desarrollo de software
 Despliegue de Software
 Métricas y Estimaciones de Software
 Auditorías de Software
 Plan de Desarrollo y mantenimiento de software

5. Objetivos establecidos en el DC

- Comprender los componentes de un proyecto de Ingeniería de Software.
- Aplicar estándares internacionales de certificación en Ingeniería de Software.
- Emplear métricas de software que se aplican al desarrollo de software.
- Aplicar técnicas y herramientas de auditoría de software.

6. Resultados de aprendizaje

Los siguientes resultados de aprendizaje se promueven en el desarrollo de la asignatura

Identificador de RA	Redacción
RA 1	Categorizar las disciplinas que conforman la Ingeniería de Software y su aporte para obtener un producto de software de calidad.
RA 2	Formular un plan de desarrollo y/o mantenimiento de software considerando todos sus elementos constitutivos para aplicarlo en el contexto de la creación de un producto de software.
RA 3	Calcular métricas para obtener visibilidad en el contexto de proyectos de desarrollo de software.

RA 4	Utilizar frameworks Lean-Ágiles para gestionar productos de software acordes a las expectativas de los involucrados, desarrollando mecanismos para la comunicación efectiva en los equipos de trabajo.
RA 5	Reconocer el aporte de la disciplina de Gestión de Configuración de Software para construir productos de software de calidad fortaleciendo el trabajo colaborativo entre los involucrados.
RA 6	Diferenciar las prácticas continuas (integración continua, entrega continua y despliegue continuo) para utilizarlas en el desarrollo de productos de software de calidad, mediante casos prácticos de uso.
RA 7	Valorar los principales modelos de calidad de software existentes para elegir el más adecuado en un proyecto de desarrollo de software, fortaleciendo el aprendizaje continuo, mediante el trabajo colaborativo.
RA 8	Utilizar técnicas y herramientas (testing, auditorías, revisión e inspecciones de software) relacionadas con el aseguramiento de la calidad del proceso y del producto con la finalidad de entregar un producto de software de calidad y fortalecer el trabajo profesional ético y responsable.
RA 9	Reconocer las diferentes formas de transferencia de un producto de software al ambiente de producción, en un entorno de despliegue específico con el objetivo de privilegiar el vínculo del equipo con los usuarios/clientes.

7. Relación de los RA y las competencias

En la tabla siguiente se indica con X la tributación de cada Resultado de Aprendizaje con las competencias de egreso: específicas, genéricas tecnológicas, sociales, políticas y actitudinales de la carrera.

RA	CE1.1	CE1.2	CE1.3	CE2.1	CE3.1	CE4.1	CE5.1	CE6.1	CE7.1	CG1	CG2	CG3	CG4	CG5	CG6	CG7	CG8	CG9	CG10	CG11
RA 1	X	-	X	-	X	X	-	-	X	X	X	X	X	X	X	X	-	-	X	-
RA 2	X	-	X	-	-	X	-	-	X	X	X	X	X	X	X	-	-	X	-	-
RA 3	X	-	X	-	X	X	-	-	X	X	X	X	X	X	X	-	-	X	-	-
RA 4	X	-	X	-	X	X	-	-	X	X	X	X	X	X	X	-	-	X	-	-
RA 5	X	-	X	-	X	X	-	-	X	X	X	X	X	X	X	-	-	X	-	-
RA 6	X	-	X	-	X	X	-	-	X	X	X	X	X	X	X	-	-	X	-	-
RA 7	X	-	X	-	-	X	X	-	-	X	X	X	X	X	X	-	-	X	-	-
RA 8	X	-	X	-	-	X	X	-	-	X	X	X	X	X	X	-	-	X	-	-
RA 9	X	-	X	-	-	X	X	-	-	X	X	X	X	X	X	-	-	X	-	-

8. Asignaturas correlativas previas

Para cursar y rendir debe tener cursadas:

- Asignatura/s:
 - Bases de Datos
 - Desarrollo de Software
 - Diseño de Sistemas de Información

Para cursar y rendir debe tener aprobada:

- Asignatura/s:
 - Sintaxis y Semántica de los Lenguajes
 - Paradigmas de Programación

9. Asignaturas correlativas posteriores

Indicar las asignaturas correlativas posteriores:

- Asignatura/s:
 - Proyecto Final

10. Programa analítico

Este programa analítico contempla los contenidos mínimos, previstos en el DC vigente, y aquellos que se consideran necesarios para desarrollar los resultados de aprendizaje propuestos.

Unidad N°: 1

Título: Ingeniería de Software en Contexto

Contenidos:

- La disciplina de Ingeniería del Software. ¿Qué comprende la ingeniería de software?
- 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 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
- Criterios para elección de ciclos de vida para proyectos de desarrollo de software.
- Vinculo proceso-proyecto-producto en la gestión de un proyecto de desarrollo de software.

Carga horaria por Unidad: 12 horas

Unidad N°: 2

Título: Gestión Lean-Ágil de Productos de Software

Contenidos:

- Gestión de Proyectos basados en procesos definidos. Planificación para el desarrollo y el mantenimiento de productos de software. Métricas y Estimaciones de Software
- Manifiesto Ágil/Filosofía Lean
- Requerimientos en ambientes lean ágil
- Requerimientos en ambientes ágiles - User Stories
- Estimaciones en ambientes ágiles
- Frameworks SCRUM a nivel equipo y escala - Métricas Ágiles
- Framework Kanban en el contexto del desarrollo de software – Métricas Lean
- Gestión de Productos de Software – Planificación de Productos – Herramientas para Definición de Productos de Software
 - Lean UX
 - Design Thinking

Carga horaria por Unidad: 36 horas

Unidad N°: 3

Título: Gestión del Software como producto

Contenidos:

- Conceptos que conforman la disciplina Gestión de Configuración de Software.
- 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 ítems de configuración en la Configuración de un software.
- Gestión de Configuración en ambientes ágiles
- Prácticas continuas (Continuous Integration /Continuous Delivery/ Continuous deployment)
- Despliegue de producto. Estrategias para el despliegue de productos.

Carga horaria por Unidad: 18

Unidad N°: 4

Título: Aseguramiento de Calidad de Proceso y de Producto

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).
- Auditorías al Software: Auditorías de Proyecto y Auditorías al Grupo de Calidad.
- Proceso de Auditorías
- 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.

Carga horaria por Unidad: 30 horas

Carga horaria por tipo de formación práctica de toda la asignatura

Tipo de formación práctica	Horas reloj
Formación experimental	10
Análisis y resolución de problemas de ingeniería y estudios de casos	35
Formulación, análisis y desarrollo de proyectos.	10

Bibliografía Obligatoria:

- Anderson, David J.(2011). Kanban. Editorial Blue Hole.
- Anderson, David J. & Carmichael, Andy (2016). Kanban Esencial Condensado. Editorial LeanKanban University.
- Toledo, Federico. (2014) Introducción a las Pruebas de Sistemas de Información. Editorial Abstracta.
- Jorgensen, Paul. (2014). Software Testing – A Craftsman's Approach. Editorial CRC Press.
- Crispin, Lisa & Gregory Janet (2008) Agile Testing – A Practical Guide for Testing and Agile Teams. Editorial O' Really Media.
- McConnell, Steve - SOFTWARE ESTIMATION: DEMYSTIFYING THE BLACK ART (Editorial Microsoft Press – Año 2006).
- Pressman, R. (2010). *Ingeniería del Software. Un enfoque práctico.* (7ma Ed.). Mc Graw - Hill Interamericana.
- Sommerville, I. (2011). *Ingeniería de Software* (Novena ed.). Mexico: Addison- Wesley.

Bibliografía optativa y otros materiales a utilizar en la asignatura:

- Cohn, Mike (2004) User Stories Applied – Editorial Addison Wesley.
- Cohn, Mike (2006). Agile Estimation and Planning – Editorial Prentice Hall.

- M. Shahin, M. Ali Babar, and L. Zhu, (2017). Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices”, IEEE Access.
- Knibber Henric (2011). Lean from the trenches – Un example of Kanban for large software project. Editado por Key Keppler.
- Rossel Sander. (2017). Continuous Integration, Delivery and Deployment, Editorial Packt
- Schneider Jonny (2017). Understanding Design Thinking, Lean and Agile – Editorial O'Reilly.
- Mc Connell, Steve. (1996). Desarrollo y Gestión de Proyectos Informáticos. Editorial McGraw Hill.
- No Silver Bullet
(<http://www.virtualschool.edu/mon/SoftwareEngineering/BrooksNoSilverBullet.html>)
- Leffingwell, Dean and Behrens Pete (2009). A user story primer Whitepaper.
- Manifiesto Ágil <http://agilemanifesto.org/iso/es/>
- The Scrum Guide 2020 - <http://www.scrumguides.org/download.html>
- The Nexus Scrum Guide 2020 - <https://www.scrum.org/resources/nexus-guide>
- <http://pgpubu.blogspot.com.ar/2007/01/tcnica-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 (1998) The Little Book of Software Configuration Management. AirLie Software Council- Sitio: <http://www.spmn.com>
- Gothelf, Jeff – Lean UX: Applying Lean Principles to Improve User Experience – Editorial O'Reilly, 2013
- Brooks, Frederick The mythical man-month (anniversary ed.) (1995) Addison-Wesley Longman Publishing Co., Inc.
- 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
- SEBOK V 1.9.1 (Software Engineering Body of Knowledge)- IEEE 2018 -
https://www.sebokwiki.org/wiki/Download_SEBoK_PDF

11. Metodología de enseñanza

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 integran.

La selección de los contenidos incluidos en el programa se realizó considerando la integración de esta 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.

Dentro de las cuestiones que se expondrán para el desarrollo de la Metodología se tendrán en cuenta los siguientes aspectos:

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, privilegiando el desarrollo de las competencias genéricas transversalmente.

A través de estas diferentes formas organizativas de la enseñanza se proponen los siguientes objetivos educativos:

- Desarrollar el hábito de la lectura, el análisis y la interpretación de textos, posibilitando que elaboren sus propias interpretaciones y elaboren 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, así como también el trabajo en equipo, la comunicación de las ideas en forma oral y escrita, considerando la ética profesional en el cumplimiento de su compromiso.

Además la catedra definió lo siguiente:

- Un espacio integrado en la UV donde se compartirá todo el material necesario y se propondrán tareas para trabajos prácticos. Cada Estudiante se matricula en el curso que está inscripto.
- Quedan a disposición de los estudiantes los links de todas las clases grabadas del ciclo lectivo 2021, que están subidas en el canal de YouTube de la cátedra.

Respecto a las estrategias de enseñanza se utilizarán tanto clases expositivas, como clases invertidas, así como también actividades con gamification. Trabajos grupales con Aprendizaje Basado en Problemas, Estudio de Casos, y técnicas de exposición de los trabajos realizados.

También se utilizarán recursos como mapas mentales y mapas conceptuales, cuadros sinópticos y comparativos para afianzar los conceptos abordados en las clases y para las evaluaciones.

Estas estrategias mediarán para que los estudiantes logren los resultados de aprendizaje propuestos en la asignatura, abordando los contenidos propuestos en cada una de las unidades temáticas.

12. Recomendaciones para el estudio

Las clases plantean y desarrollan los conceptos y su aplicación a los casos prácticos utilizando ejemplos; para dar continuidad al aprendizaje abordando los temas y afianzando los saberes privilegiando el “hacer”. Dada esta estrategia las recomendaciones son:

- Asistir a las clases con las consignas pedidas cumplidas, leer bibliografía, llevar material o cualquier otra recomendación, según sea el caso o analizado.
- Hacer al menos los ejercicios que la cátedra pone a disposición.
- Analizar en forma personal e individual que necesidades de realización de ejercicios prácticos adicionales tiene cada estudiante. Los ejercicios prácticos están disponibles en el aula virtual de la cátedra.
- No quedarse con dudas sobre ningún tema, consultar a los docentes en el momento de la clase o solicitar una clase de consulta.
- Participar activamente en las actividades grupales propuestas.
- Utilizar las soluciones propuestas para contrastarlas con las propias, luego de resolver los ejercicios.
- Estudiar para los parciales utilizando la bibliografía recomendada.

13. Metodología de evaluación

El modelo de enseñanza basado en competencias implica la aplicación de metodologías e instrumentos de evaluación que permiten conocer, a docentes y estudiantes, el nivel de desarrollo de las competencias que aborda la asignatura.

Se trabajará en forma combinada con las diferentes formas de evaluación: diagnóstica, formativa, de autoevaluación y sumativas que se utilizarán durante el cursado.

Las evaluaciones diagnósticas principalmente en las clases que introducen cada tema.

Las evaluaciones formativas están centradas en la observación de la participación del estudiante en la clase en las distintas dinámicas que se proponen y en la conformación y evolución del portfolio grupal.

Para la evaluación sumativa se plantea la realización de 2 parciales, entre 8 y 10 ejercicios prácticos grupales y 2 trabajos de investigación grupales.



Facultad Regional Córdoba

A continuación, se detallan todos los Resultados de Aprendizajes con sus contenidos a desarrollar para alcanzarlos, la mediación pedagógica, metodologías y estrategias de evaluación, tiempo en horas reloj.

Resultados de Aprendizaje	Contenidos según programa	Mediación Pedagógica	Metodología y Estrategias de Evaluación	Tiempos en hora reloj
RA 1	<ul style="list-style-type: none"> • La disciplina de Ingeniería del Software. ¿Qué comprende la ingeniería de software? • Estado Actual y Antecedentes. La Crisis del Software. • Disciplinas que conforman la Ingeniería de Software. • Ejemplos de grandes proyectos de software fallidos y exitosos. 	<p>Estrategias:</p> <ul style="list-style-type: none"> • Clase magistral participativa • Preguntas exploratorias • Mapas conceptuales • Estudio de casos <p>Actividades</p> <ul style="list-style-type: none"> • Analizar el material bibliográfico indicado para cada tema • Debates • Desarrollar preguntas al cierre de cada clase. 	<p>Instrumentos</p> <ul style="list-style-type: none"> • Parcial • Preguntas diagnósticas <p>Criterios de Evaluación:</p> <ul style="list-style-type: none"> • Utiliza el lenguaje técnico adecuado • Categoriza las disciplinas que conforman la ingeniería de software • Organiza los errores frecuentes y conocidos de los proyectos de desarrollo de software 	4 horas áulicas 2 horas extra áulicas
RA 2	<ul style="list-style-type: none"> • Ciclos de vida y su influencia en la Administración de Proyectos de Software. • Criterios para elección de ciclos de vida para 	<p>Estrategias</p> <ul style="list-style-type: none"> • Clase magistral participativa • Aula invertida • Estudio de casos • Preguntas exploratorias • Mapas mentales 	<p>Instrumentos</p> <ul style="list-style-type: none"> • Parcial • Portfolio grupal. • Guías de Ejercicios 	4 horas áulicas 2 horas extra áulicas

	<ul style="list-style-type: none"> proyectos de desarrollo de software. Vinculo proceso-proyecto-producto en la gestión de un proyecto de desarrollo de software. Gestión de Proyectos basados en procesos definidos. Planificación para el desarrollo y el mantenimiento de productos de software. 	Actividades <ul style="list-style-type: none"> Analizar el material bibliográfico indicado para cada tema Identificar elementos que conforman un plan de desarrollo de software Elaborar esquemas de representación de conceptos adquiridos en un mapa mental Formular planes de desarrollo de software 	Criterios de Evaluación <ul style="list-style-type: none"> Utiliza lenguaje técnico adecuado. Diferencia ciclos de vida adecuados para utilizar en cada situación. Construye un plan de desarrollo y/o mantenimiento de software adecuado al contexto propuesto.
RA 3	<ul style="list-style-type: none"> Métricas y Estimaciones de Software Métricas Ágiles Métricas Lean 	Actividades <ul style="list-style-type: none"> Estrategias Clase magistral participativa Clase invertida Estudio de casos Aprendizaje basado en problemas Gamification 	Instrumentos <ul style="list-style-type: none"> Parcial Guía de Ejercicios Portfolio grupal Dinámicas grupales Criterios de Evaluación <ul style="list-style-type: none"> Formula métricas adecuadas al contexto que se les presenta. Interpreta la información que brindan los indicadores construidos para un proyecto. Distingue las métricas adecuadas

		<ul style="list-style-type: none"> Participar en las clases, respondiendo preguntas, planteando dudas. Elaborar cuadros comparativos. 	<ul style="list-style-type: none"> para cada caso en particular. Estima el tamaño de un producto de software. 	
RA 4	<ul style="list-style-type: none"> Requerimientos en ambientes ágiles - User Stories Estimaciones en ambientes ágiles Frameworks SCRUM a nivel equipo y escala Framework Kanban en el contexto del desarrollo de software Gestión de Productos de Software – Planificación de Productos – Herramientas para Definición de Productos de Software Lean UX- Design Thinking 	<p>Estrategias</p> <ul style="list-style-type: none"> Clase magistral participativa Clase invertida Estudio de casos Aprendizaje basado en problemas Gamificación Resolución de ejercicios <p>Actividades</p> <ul style="list-style-type: none"> Analizar el material bibliográfico indicado para cada tema Realizar las dinámicas propuestas Resolver los ejercicios de la guía de ejercicios Participar en las clases, respondiendo preguntas, planteando dudas. Elaborar un trabajo y exponerlo 	<p>Instrumentos</p> <ul style="list-style-type: none"> Parcial Portfolio grupal Guías de Ejercicios Exposiciones grupales Elaboración de pósters científicos Rúbricas <p>Criterios de Evaluación</p> <ul style="list-style-type: none"> Preparar un póster respetando los lineamientos definidos para crear un póster científico Identificar historias de usuario Diseñar un proceso utilizando lineamientos de Kanban Argumenta sobre la conveniencia del uso de los diferentes enfoques lean ágil para desarrollar software Crea visiones de productos de 	<p>22 horas áulicas 8 horas extra áulicas</p>

			software utilizando herramientas de design thinking
RA 5	<ul style="list-style-type: none"> • Conceptos que conforman la disciplina • Gestión de Configuración de Software. • 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 ítems de configuración en la Configuración de un software. 	<p>Estrategias</p> <ul style="list-style-type: none"> • Clase magistral participativa • Clase invertida • Estudio de casos • Aprendizaje basado en problemas • Resolución de ejercicios <p>Actividades</p> <ul style="list-style-type: none"> • Analizar el material bibliográfico indicado para cada tema • Realizar las dinámicas propuestas • Resolver los ejercicios de la guía de ejercicios • Participar en las clases, respondiendo preguntas, planteando dudas. • Analizar productos de software para SCM existentes en el mercado para seleccionar la que utilizarán durante el cursado. 	<p>Instrumentos</p> <ul style="list-style-type: none"> • Parcial • Portfolio grupal. • Guías de Ejercicios • Exposición de trabajos realizados <p>Criterios de Evaluación</p> <ul style="list-style-type: none"> • Usa lenguaje técnico adecuado. • Utilizar una herramienta de gestión de configuración para administrar sus ítems de configuración. • Justifica los ítems de configuración que incluye en una línea base • Planifica la gestión de configuración de software para un proyecto <p>8 horas áulicas 6 horas extra áulicas</p>

	<ul style="list-style-type: none"> Gestión de Configuración en ambientes ágiles 		
RA 6	<ul style="list-style-type: none"> Prácticas continuas (Continuous Integration /Continuous Delivery/ Continuous deployment) 	<p>Estrategias</p> <ul style="list-style-type: none"> Clase magistral participativa Clase invertida Aprendizaje basado en investigación <p>Actividades</p> <ul style="list-style-type: none"> Analizar el material bibliográfico indicado para cada tema Participar en las clases, respondiendo preguntas, planteando dudas. Analizar productos de software existentes en el mercado para la implementación de prácticas continuas 	<p>Instrumentos</p> <ul style="list-style-type: none"> Parcial Portfolio grupal Exposición de trabajos realizados <p>Criterios de Evaluación</p> <ul style="list-style-type: none"> Compara las tres prácticas continuas, distinguiendo sus propósitos diferencias. Evaluá estrategias para realizar despliegue continuo de software Explica el contexto necesario para aplicar prácticas continuas.
RA 7	<ul style="list-style-type: none"> Conceptos generales sobre calidad. Importancia de trabajar para y con Calidad. Ventajas y Desventajas. 	<p>Estrategias</p> <ul style="list-style-type: none"> Clase magistral participativa Clase invertida <p>Actividades</p>	<p>Instrumentos</p> <ul style="list-style-type: none"> Parcial Debates <p>Criterios de Evaluación</p>

	<ul style="list-style-type: none"> Actividades relacionadas con el Aseguramiento de la Calidad del Software. Principales Modelos de Calidad existentes (CMMI – SPICE – ISO). 	<ul style="list-style-type: none"> Analizar el material bibliográfico indicado para cada tema Participar en las clases, respondiendo preguntas, planteando dudas. Prepararse para participar en los debates 	<ul style="list-style-type: none"> Debate sobre los principales modelos de calidad aplicables al software. Explica las actividades implicadas en el aseguramiento de calidad de software 	
RA 8	<ul style="list-style-type: none"> Auditorías al Software: Auditorías de Proyecto y Auditorías al Grupo de Calidad. Proceso de Auditorías 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. 	<p>Estrategias</p> <ul style="list-style-type: none"> Clase magistral participativa Clase invertida Estudio de casos Dinámicas <p>Actividades</p> <ul style="list-style-type: none"> Analizar el material bibliográfico indicado para cada tema Realizar las dinámicas propuestas Resolver los ejercicios de la guía de ejercicios Participar en las clases, respondiendo preguntas, planteando dudas. Elaborar cuadros comparativos. 	<p>Instrumentos</p> <ul style="list-style-type: none"> Parcial Portfolio grupal. Casos prácticos de Estudio Exposiciones grupales <p>Criterios de Evaluación</p> <ul style="list-style-type: none"> Diseña casos de prueba Prueba una funcionalidad de un producto de software Realizar una revisión técnica de código Diferencia el testing ágil del testing tradicional Compara las diferentes técnicas para asegurar la calidad de un 	<p>14 horas áulicas 8 horas extra áulicas</p>

			producto de software.
RA 9	<ul style="list-style-type: none"> • Despliegue de producto. • Estrategias para el despliegue de productos. 	<p>Estrategias</p> <ul style="list-style-type: none"> • Clase magistral participativa • Clase invertida • Aprendizaje basado en investigación <p>Actividades</p> <ul style="list-style-type: none"> • Analizar el material bibliográfico indicado para cada tema • Realizar las dinámicas propuestas • Participar en las clases, respondiendo preguntas, planteando dudas. • Investigar sobre el tema asignado y preparar una exposición. 	<p>Instrumentos</p> <ul style="list-style-type: none"> • Parcial • Portfolio grupal. • Exposiciones grupales con Pecha Kucha • Rúbricas <p>Criterios de Evaluación</p> <ul style="list-style-type: none"> • Compara las diferentes estrategias de despliegue • Explica las ventajas y desventajas de los enfoques de despliegue de productos de software

14. Condiciones de aprobación

Para regularizar, el estudiante deberá:

- Rendir y aprobar 2 (dos) parciales.
- Puede acceder a 1 (un) recuperatorio de cualquiera de ellos. Los parciales recuperatorios se tomarán a fin del cuatrimestre.
- En caso de recuperar se considera la mejor nota obtenida por el estudiante.
- Cumplir con las siguientes condiciones respecto a los trabajos grupales:
- **Ejercicios Prácticos:** en cada una de las unidades se 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 la totalidad de los ejercicios prácticos propuestos 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 *libre*.
- **Trabajos de Investigación Grupales:** se propondrá a los estudiantes la realización de dos (2) trabajos de este tipo. Debe presentar y aprobar la totalidad de los trabajos propuestos con opción a recuperar 1 de ellos, por cualquier motivo que se lo requiera. En caso de no lograr el criterio de aprobación establecido, cada uno de los integrantes del grupo quedará en condición de *libre*.

La nota mínima de aprobación para los parciales es un 4 (cuatro),

La escala de notas para aprobación de parciales es la siguiente:

Nota	Porcentaje	Situación
1		No aprueba
2		No aprueba
3		No aprueba
4	55 % - 57 %	Aprueba
5	58% - 59 %	Aprueba
6	60 % - 68 %	Aprueba
7	69 % - 77%	Aprueba
8	78% - 86%	Aprueba
9	87% - 95 %	Aprueba
10	96% - 100 %	Aprueba

Aprobación Directa:

Los estudiantes podrán obtener la aprobación directa de la asignatura si:

- Obtienen notas mayores o iguales a 7 (siete) en los parciales y trabajos de investigación.
- Puede optar por recuperar parciales para obtener 7 o más, con el mismo criterio que los parciales para obtener la regularidad y en las mismas fechas. La nota que se considera es la mayor obtenida.
- 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
- Cumplir con las condiciones explicadas respecto de los ejercicios grupales; con notas mayores o iguales a 7 en al menos el 70 % de los ejercicios prácticos evaluables.

15. Modalidad de examen

De no obtener aprobación directa, el estudiante deberá aprobar el examen final.

Al momento de la inscripción al examen final, el sistema de inscripción le asigna aleatoriamente un tema, basado en los contenidos de las unidades temáticas de la materia. Este tema será el primer tema que el estudiante exponga en su coloquio, de no alcanzar nivel satisfactorio en su exposición, el examen se da por finalizado con la no aprobación del estudiante.

Luego de la exposición satisfactoria del primer tema, los docentes le asignarán dos temas más para que el estudiante desarrolle. Finalizado el coloquio se le informará la nota.

En esta instancia se evaluarán todos los contenidos del *último programa vigente* para la asignatura.

El examen final se aprueba con nota 6 (seis) o superior, correspondiendo al 60 % de los contenidos evaluados.

La cátedra tomará los exámenes finales en forma conjunta para todos los estudiantes, esto permitirá la nivelación e integración de todos las comisiones que la conforman.

Escala de Notas:

Nota	Porcentaje	Situación
1		Insuficiente
2		Insuficiente
3		Insuficiente
4		Insuficiente
5		Insuficiente
6	60 % - 68 %	Aprobado
7	69 % - 77%	Bueno
8	78% - 86%	Muy Bueno
9	87% - 95 %	Distinguido
10	96% - 100 %	Sobresaliente

16. Recursos necesarios

- Aulas limpias, con conexiones eléctricas suficientes y acceso a Internet, con el tamaño adecuado a la cantidad de estudiantes inscriptos.
- Proyectores multimedia.
- Software para soporte a prácticas continuas.
- Software para dar soporte a la administración de configuración de software.
- Software para gestión de proyectos lean ágil.