Ingeniería y Calidad De Software

Planificación Ciclo lectivo 2026

1. Datos administrativos de la asignatura			
Departamento:	Sistemas de Información	Carrera	Ing. en Sistemas de Información
Asignatura:	**Ingeniería y Calidad de Software**		
Nivel de la carrera	4to Año	Duración	Cuatrimestral
Plan	2023 Ord. 1877	Modalidad	Presencial
Bloque Curricular:	Tecnologías Aplicadas		
Área	Desarrollo de Software	RTF	6
Horas cátedra semanal:	3	Horas reloj anual total:	72
Carga horaria no presencial semanal	----------	% horas no presenciales	----------
Profesor/es Titular/Asociado/Adjunto:	Mg. Vanzetti Juan José	Dedicación:	Simple
Auxiliar/es de 1°/JTP	Ing. Toselli. Laura	Dedicación:	Exclusiva
Competencias	Específicas	Genéricas	
	CE1.1 – CE1.3 – CE3.1 – CE4.1 – CE5.1	CG3 – CG4 – CG6 - CG7	

2. Presentación, Fundamentación

En el contexto actual, la ingeniería de software desempeña un papel fundamental en la concepción y desarrollo de soluciones tecnológicas que impulsan el progreso en diversas áreas. Es necesario comprender que el desarrollo de software no se limita únicamente a la codificación, sino que implica un enfoque sistemático y disciplinado que abarca todas las etapas del ciclo de vida del software.

La aplicación de técnicas y metodologías de ingeniería de software permite gestionar eficientemente los proyectos, garantizando la calidad, la eficiencia y la robustez de las soluciones desarrolladas. Además, fomenta la colaboración y la comunicación efectiva entre los miembros del equipo, promoviendo así un ambiente propicio para la innovación y el éxito en la entrega de software.

Es importante reconocer que la ingeniería de software se basa en principios sólidos, como la planificación, la gestión de requisitos, el diseño estructurado, la implementación rigurosa, las pruebas exhaustivas y el mantenimiento continuo. Estos principios aseguran que los sistemas de software sean confiables, seguros y escalables, brindando soluciones que se ajusten a las necesidades de las organizaciones y los usuarios

En resumen, la ingeniería de software es esencial para enfrentar los desafíos actuales en el desarrollo de software, permitiendo concebir soluciones tecnológicas efectivas y de calidad que impulsen el progreso en nuestras vidas y en las organizaciones.

Relación de la asignatura con el perfil de egreso.

Dado que el ingeniero de sistemas en información debe ser un profesional de sólida formación analítica que le permite la resolución de problemas, esta asignatura le aportará diversos conocimientos para desarrollar tareas profesionales.

Relación de la asignatura con los alcances del título.

Con los conocimientos obtenidos el alumno podrá aplicarlos en la especificación, gestión, desarrollo tanto de sistemas de información software, como así también en lo concerniente al establecimiento de normas de calidad para el desarrollo y gestión del software.

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

Se detalla en la siguiente tabla, la relación de la asignatura con las competencias de egreso específicas, genéricas tecnológicas, sociales, políticas y actitudinales de la carrera. Indicando a cuáles competencias de egreso tributa y en qué nivel (No Tributa, 1=bajo, 2=medio, 3=alto).

Competencias	Nivel
Competencias genéricas tecnológicas (CG):	
CG. 1. Identificar, formular y resolver problemas de ingeniería.,	No Tributa
CG. 2. Concebir, diseñar y desarrollar proyectos de ingeniería.	No Tributa
CG. 3. Gestionar, planificar, ejecutar y controlar proyectos de ingeniería.	2
CG. 4. Utilizar de manera efectiva las técnicas y herramientas de aplicación en la ingeniería.	2
CG. 5. Contribuir a la generación de desarrollos tecnológicos y/o innovaciones tecnológicas.	No Tributa
Competencias genéricas sociales, políticas y actitudinales (CG):	
CG. 6. Desempeñarse de manera efectiva en equipos de trabajo.	2
CG. 7. Comunicarse con efectividad	2
CG. 8. Actuar con ética, responsabilidad profesional y compromiso social, considerando el impacto económico, social y ambiental de su actividad en el contexto local y global.	No Tributa
CG. 9. Aprender en forma continua y autónoma.	No Tributa
CG. 10. Actuar con espíritu emprendedor	No Tributa
Competencias Específicas (CE):	
CE. 1.1. 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	2
CE. 1.2. 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 Tributa
CE. 1.3. 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.	2
CE. 2.1. 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 Tributa

CE. 3.1. 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.	3
CE. 4.1. 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.	2
CE. 5.1. 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.	2
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 intervinientes.	No Tributa
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 Tributa

4. Propósito

Elaborar productos de software de alta calidad mediante un proceso respaldado por métodos y herramientas, con el objetivo de desarrollar aplicaciones informáticas que se adapten eficazmente a las necesidades de las organizaciones. Además, se busca dirigir y coordinar el desarrollo de aplicaciones complejas, interviniendo en todas las etapas del ciclo de vida del producto. Asimismo, se requiere habilidad para estimar las distintas dimensiones de un proyecto y determinar los plazos de desarrollo.

5. Objetivos establecidos en el Diseño Curricular

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

6. Contenidos Mínimos establecidos en el Diseño Curricular

- 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.
- Auditoría de Software.
- Plan de desarrollo y mantenimiento de software

7. Resultados de aprendizaje

Se describen los Resultados de aprendizaje (RA) a promover en el desarrollo de la asignatura.

Identificador del RA	Redacción
RA1	Aplica los componentes y conceptos fundamentales de la Ingeniería de Software, gestionando proyectos mediante metodologías de planificación, ejecución y control, asegurando el cumplimiento de los objetivos establecidos conforme a la norma IEEE 1058 y las mejores prácticas de frameworks agile.
RA2	Utiliza técnicas de estimación, métricas y auditoría para evaluar el tamaño, esfuerzo y complejidad del proyecto, garantizando visibilidad y calidad, mediante herramientas que identifiquen mejoras y gestionen riesgos, conforme a normas y estándares internacionales.
RA3	Selecciona las actividades clave de la gestión de configuración de software y gestiona el despliegue de productos, utilizando las mejores prácticas y herramientas, para asegurar la integridad, control y entrega eficaz del software.
RA4	Evalúa los principales modelos de calidad de software, para aplicarlos de manera efectiva en contextos de desarrollo, considerando modelos como ISO, CMMI, McCall y Boehm.
RA5	Implementa las actividades de verificación y validación en el desarrollo de software, asegurando que el producto cumpla con las necesidades del usuario mediante técnicas de testing y retroalimentación continua, conforme a estándares como IEEE 829-2008, ISO/IEC/IEEE 29119 y el Marco de Certificación ISTQB.

RA6	Evalúa competencias transversales en la ejecución de actividades que fomenten la comunicación efectiva y potencien el pensamiento creativo, considerando la empatía, resolución de conflictos y trabajo en equipo.
RA7	Desarrolla el plan de mantenimiento de software, asegurando su evolución y calidad a lo largo del ciclo de vida mediante la planificación, ejecución y monitoreo de actividades clave, integrando retroalimentación y mejora continua conforme a las mejores prácticas y normas internacionales.

8. Asignaturas correlativas previas

Para cursar y rendir

Cursada:

- Desarrollo de Software.
- Diseño de Sistemas de Información.
- Bases de Datos.

Aprobada:

- Sintaxis y Semántica de los Lenguajes.
- Paradigmas de Programación.

9. Asignaturas correlativas posteriores

- Proyecto Final

10. Programa analítico, Contenidos temáticos

Ejes temáticos	Contenidos
Ingeniería de Software.	Concepto de Software. Naturaleza, Evolución, Importancia y Problemas del Software. Características. Mitos. Crisis del Software. Definición de Ingeniería de Software. Evolución. El proceso de desarrollo de software. Características. Ciclo de vida del software. Evaluación del proceso. Ventajas y desventajas. Modelos de proceso software. Objetivo de los modelos.Standard ISO/IEC 12207.Desarrollo ágil, #noprojects, design thinking Componentes de un proyecto de Sistemas de información. Vinculo proceso-proyecto-producto en la gestión de un proyecto de desarrollo de software

Estimación y Métricas de proceso y proyecto	Estimación de software: métodos basados en líneas de código, puntos de función y modelos paramétricos. Introducción a métricas de software: productividad, calidad, y métricas de proceso. Auditoría de software: fundamentos y procedimientos. Identificación de riesgos en proyectos. Herramientas de estimación y seguimiento: uso práctico en contextos reales. Estándares internacionales relacionados con métricas y auditoría (ISO/IEC 25000, IEEE 12207).
Gestión de configuración de software	Gestión de Cambios. Fundamentos de la gestión de configuración de software (SCM). Concepto. Versiones, reléase. Patrones de rama. Línea Base Procesos clave de la SCM: identificación, control, auditoría y reportes de configuración. Herramientas de gestión de configuración. Gestión del despliegue de software: planificación, ejecución y control. Automatización del despliegue: herramientas CI/CD. Buenas prácticas en SCM y despliegue.
Calidad del Software	Calidad: concepto y visiones Garantía de calidad (SQA). Modelos de calidad: McCall, Boehm, ISO9126.SPICE Medición de la calidad. Mediciones de McCall. Concepto de defecto. Medidas de aspectos de la calidad: Medición de densidad de defectos, Mediciones de usabilidad, Mediciones de mantenimiento. Inspecciones de software. Introducción al CMMI. Nivel inicial. Niveles de madurez. Evolución del proceso. Estructura del CMMI. Áreas clave del proceso por nivel. Áreas clave del proceso: estructura. Usando el CMMI. Ejemplo de área clave de proceso
Verificación y Validación	Introducción a la verificación y validación de software (V&V). Principios y niveles de testing: unitario, integración, sistema y aceptación. Estrategias de testing: funcional, no funcional y regresión. Estándares de testing: IEEE 829-2008, ISO/IEC/IEEE 29119. Marco de Certificación ISTQB

	Diseño y ejecución de casos de prueba.Uso de herramientas de testing etc.
Habilidades Blandas	Habilidades de comunicación efectiva: técnicas y estrategias. Resolución de conflictos en equipos de desarrollo. Principios y dinámicas del trabajo en equipo. Empatía en el desarrollo de software. Herramientas colaborativas.
Mantenimiento	Conceptos de mantenimiento de software: tipos y objetivos. Actividades clave en el mantenimiento de software. Planificación del mantenimiento: estrategias y recursos. Control y monitoreo de actividades de mantenimiento. Integración de retroalimentación en el ciclo de vida del software. Normas internacionales aplicables al mantenimiento (ISO/IEC 14764).

Cuadro de Carga Horaria Práctica

Tipo de Formación Práctica	**Carga horaria**	
	Presencial	**A Distancia**
Instancias supervisadas de formación práctica (prácticas en diferentes ámbitos tales como aula, laboratorios, campo u otros)	32 hs	
Proyecto Integrador		
Práctica profesional supervisada		
Carga horaria total	32 hs	

11. Metodología de enseñanza

En la cátedra se realizan exposiciones orales para transmitir los contenidos teóricos de la materia. Durante estas exposiciones, se enfatizará la importancia de la lectura anticipada de los temas a tratar, lo que permitirá a los estudiantes tener un contexto previo y aprovechar al máximo la clase. Además, se fomentará la elaboración de mapas conceptuales para ayudar a los estudiantes a organizar y visualizar la información, así como la integración de los contenidos para una comprensión más completa. Durante las charlas se usarán herramientas como el Mentimeter, para afianzar la participación de los alumnos mediante diferentes formatos de participación para el alumno. Lo cual permitirá que jueguen e interactúen con la herramienta.

Se presentarán y analizarán publicaciones relevantes en el campo de la Ingeniería de Software. Esto permitirá a los estudiantes estar al tanto de los avances y tendencias en el área, así como desarrollar habilidades de búsqueda y análisis crítico de información.
Se utilizarán recursos audiovisuales, como videos y podcasts, relacionados con los temas de la materia.
Estos recursos proporcionarán ejemplos prácticos y casos de estudio que ayudarán a los estudiantes a visualizar y comprender los conceptos teóricos de manera más dinámica y aplicada.

Se proporcionarán guías prácticas que incluirán ejercicios y problemas relacionados con los contenidos de la materia. Estas actividades permitirán a los estudiantes aplicar los conocimientos teóricos adquiridos, resolver problemas prácticos y desarrollar habilidades prácticas en el campo de la Ingeniería de Software.
Se solicitará a los estudiantes que realicen un trabajo de investigación sobre un tema específico de la Ingeniería de Software. Esto les permitirá profundizar en un área de interés, aplicar métodos de investigación y desarrollar habilidades de búsqueda, análisis, presentación y escritura académica.

Durante el cursado de la materia, se desarrollará un trabajo práctico integrador que abarcará todos los temas estudiados. Este proyecto permitirá a los estudiantes aplicar de manera integrada los conocimientos teóricos y prácticos adquiridos, trabajando en equipo y desarrollando habilidades de gestión de proyectos.
La carga horaria se divide equitativamente entre las clases teóricas y prácticas, con un 50% para cada una. Las clases teóricas se centran en la adquisición gradual de conocimientos, mientras que las clases prácticas promueven la comprensión y aplicación de los conceptos a través de actividades prácticas y de resolución de problemas. En total, se asignan 96 horas para todas las actividades de formación, brindando un equilibrio adecuado entre la adquisición de conocimientos teóricos y el desarrollo de habilidades prácticas.

Resumen de Competencias - Resultados de Aprendizaje - Metodologías de Enseñanza de la asignatura.

COMPETENCIA - NIVEL DE TRIBUTACIÓN	RESULTADO DE APRENDIZAJE (RA)
CE1.1(2) – CE1.3(2) – CE3.1(3) – CE4.1(2) – CE5.1(2) - CG3(2) – CG4(2) – CG6(2) - CG7(2)	RA1 - RA2 - RA3 - RA4 -RA5 - RA6 – RA7

EJES TEMÁTICOS - CONTENIDOS

Ingeniería de Software - Estimación y Métricas de proceso y proyecto - Gestión de configuración de software - Calidad del Software - Verificación y Validación - Habilidades Blandas - Mantenimiento

ACTIVIDADES DEL ESTUDIANTE (FORMATIVAS)

El **estudiante** deberá realizar las siguientes actividades para alcanzar los **RA** propuestos:

- Lectura y análisis de material bibliográfico, diapositivas y videos.
- Participar activamente en clase.
- Resolver Casos de Estudios.
- Utilizar software de modelado, estimación y configuración como herramienta para la verificación y validación.
- Realizar el Trabajo Práctico Integrador.
- Responder cuestionarios de autoevaluación.
- Realizar Instancia de Evaluación de integración de saberes.
- Elaborar una exposición oral para presentar el TPI.

ESTRATEGIAS DE ENSEÑANZA

El **docente** diseñará las siguientes **actividades pedagógicas**:

- Clase expositiva dialogada
- Acompañamiento continuo.
- Diseño de casos de estudios de trabajos prácticos
- Diseñar el Trabajo Práctico Integrador.
- Diseñar cuestionarios de evaluación y autoevaluación
- Diseñar Instancia de evaluación individual de integración de saberes.
- Retroalimentación formativa.
- Coordinación de trabajo colaborativo.

12. Recomendaciones para el estudio

Antes de cada clase, es fundamental dedicar tiempo a la preparación. Se recomienda realizar una lectura anticipada de los temas que se abordarán, sumergiéndose en los conceptos básicos

y tomando notas de los puntos clave. Esta anticipación te permitirá llegar a clase con una base sólida de conocimientos y aprovechar al máximo la exposición del profesor.

Además, una estrategia efectiva para comprender y visualizar la estructura de la materia es crear mapas conceptuales. Estos mapas te ayudarán a identificar las relaciones entre los diferentes conceptos y a tener una visión general de la materia. Al elaborarlos, podrás organizar tus ideas de manera visual y comprender mejor cómo se interconectan los diferentes temas.

Durante las exposiciones de la cátedra, es importante participar activamente. La participación activa en clase no solo permitirá profundizar en los temas, sino que también enriquecerá las discusiones y facilitará un aprendizaje colaborativo.

Para fortalecer tus habilidades y aplicar los conceptos teóricos, es fundamental completar las guías prácticas y los ejercicios propuestos. Estas actividades prácticas te brindarán la oportunidad de poner en práctica lo aprendido, resolver problemas reales y desarrollar habilidades concretas en la materia.

13. Metodología de evaluación

EVALUACIÓN

Para llevar a cabo la evaluación desde el enfoque formativo es necesario que el docente incorpore en el aula estrategias de evaluación congruentes con las características y necesidades individuales de cada alumno y las colectivas del grupo. El cumplimento en la entrega en tiempo y forma de las actividades prácticas. En caso que las actividades contengan errores, los alumnos tendrán la posibilidad de ir corrigiéndolos a media que transcurre el cuatrimestre en entregas sucesivas. Al finalizar el cuatrimestre, los alumnos deberán haber concluido correctamente todas las actividades prácticas. Se evaluarán además a los alumnos en instancias parciales

Momentos: Evaluación Diagnóstica, Evaluación formativa o continua.

Instrumentos:

- 2 parciales teóricos y 1 parcial práctico
- Resolución de guías de trabajos prácticos.
- Resolución proyecto integrador
- Confección de videos y podcast
- Presentaciones

Criterios de:

Regularidad: Para obtener la regularidad de la materia el alumno deberá

- Tener el 75% de asistencias
- Tener completas y entregadas las guías de trabajos prácticos.
- Tener completo el proyecto integrador
- Tener aprobados los parciales prácticos y teóricos con nota promedio 6 o superior.

- Podrá recuperar una instancia de parcial práctico y uno teórico.
- Tener aprobados la entrega de videos, podcast y exposiciones.

Promoción: Para obtener la promoción directa de la materia el alumno deberá

- Tener el 75% de asistencias
- Tener completas y entregadas las guías de trabajos prácticos.
- Tener aprobados los parciales prácticos y teóricos con nota 8 o superior.
- Tener aprobados la entrega de videos, podcast y exposiciones.

Podrá recuperar una instancia de parcial práctico y uno teórico.

14. Cronograma de clases año 2026

Semana	Contenido / Actividad
Semana 1	Presentación de la materia. Encuesta Presentación materia. RA 1 Software. Naturaleza. Evolución. Descripción de las habilidades blandas de la ingeniería del software.
Semana 2	RA1: Crisis del software. Definición Ingeniería de software. Características internas/externas Proceso. Proceso de desarrollo de software. Características.
Semana 3	RA1: Modelos de proceso de desarrollo de software. Características. Comparación. Procesos Ágiles.
Semana 4	Gestión de proyecto. Áreas de gestión. Visión PMI. Generación de un plan.
Semana 5	RA3: Cambio. Concepto. Características. Gestión de cambios. Gestión de la configuración. Proceso.
Semana 6	RA3: Gestión de configuración estrategias. Integración Continua Estrategias.
Semana 7	Actividad Práctica. RA3: Gestión de configuración estrategias. Integración Continua Estrategias.
Semana 8	RA4: Calidad. Concepto. Objetivos. Actividad Práctica.
Semana 9	RA2: Estimación. Concepto. Objetivos Técnicas.
Semana 10	Actividad Práctica. RA2: Métrica. Concepto. Objetivos. Tipos. Programa de métricas.
Semana 11	RA 2: Auditoria.
Semana 12	RA 2: Auditoria. Actividad Práctica.
Semana 13	RA5: Testing. Concepto. Objetivos. - RA 5: Testing. Tipos de pruebas. Actividad Práctica.
Semana 14	RA7: Conceptos de mantenimiento de software Actividades clave en el Mantenimiento. Planificación del mantenimiento: estrategias y recursos. Control y monitoreo de actividades de mantenimiento. Actividad Práctica.

Semana 15	Integración de retroalimentación en el ciclo de vida del software. Actividad Evaluativa.
Semana 16	Recuperatorio

15. Recursos necesarios

- Recopilación de apuntes de la Cátedra
- Retroproyector, proyección de videos
- Mentimeter, Notion, Miro.

16. Referencias bibliográficas (citadas según Normas APA)

Bibliografía Obligatoria

- Pressman, R. (2021). *Ingeniería de software: Un enfoque práctico*. McGraw Hill.
- Sommerville, I. (2016). *Ingeniería de software* (10.ª ed.). Pearson - Addison Wesley.

Bibliografía Complementaria

- Echenique García, J. A. (s.f.). *Auditoría en informática* (2.ª ed.). McGraw Hill.

17. Función Docencia

- Dictado de las clases
- Elaboración de plan de trabajos de situaciones prácticas
- Actualización de material bibliográfico
- Aprobación de instrumentos de evaluación.
- Diseñar las clases para lograr trabajo colaborativo y desempeño de habilidades blandas.
- Capacitación continua en temas específicos como en estrategias de enseñanza por competencias.

18. Reuniones de asignatura y área

Reuniones periódicas de asignatura y se prevén dos reuniones de área en el año.
La cátedra queda sujeta a las reuniones y capacitaciones que el coordinador de área establezca o planifique.

19. Atención y orientación a las y los estudiantes

En el cronograma establecido por la cátedra se encuentran las fechas de las instancias de evaluación grupal e individual como así también fecha de recuperatorios por cada instancia de evaluación.

Las clases de consultas se coordinan con los estudiantes según sus necesidades.

Se enfatiza el uso de los recursos de comunicación asincrónica y sincrónica que proporciona la plataforma virtual (Moodle).

Se planifican encuentros individuales o grupales por medio de videoconferencia por las diversas plataformas.