
	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 1 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	

## DATOS INFORMATIVOS

**Código y Nombre de la Asignatura:** LTI\_05A\_INS Ingeniería del Software

**Número de horas componente docencia:** 2

**Número de horas componente de prácticas de aplicación y experimentación de los aprendizajes:** 0

**Número de horas de aprendizaje autónomo:** 4

**Número de créditos de asignatura:** 2

**Nivel al que pertenece la asignatura:** 5

**Periodo académico ordinario:** octubre 2025 – febrero 2026

### Estructura curricular:

Unidad de organización curricular					
<input type="checkbox"/>	Unidad Básica	<input checked="" type="checkbox"/>	Unidad profesional	<input type="checkbox"/>	Unidad de integración curricular

**Prerrequisito(s) código y nombre de la Asignatura:** LTI\_05U\_ESD Estructura de Datos

**Correquisito(s) código y nombre de la Asignatura:** Ninguno

**Número de sesiones:** 32 horas / 16 semanas

**Horario de clases:** jueves, 16h00 a 18h00

**Horario de tutorías:** lunes, 11h00 a 12h00



### Descripción y como aporta al perfil profesional

La asignatura Ingeniería de Software ha sido enriquecida con material Arizona State University (ASU) tomada del sílabo: SER-216 Software Enterprise Processes and Quality.

Esta asignatura proporciona a los estudiantes los fundamentos teóricos y prácticos de la ingeniería de software, enfocándose en procesos empresariales y calidad. El curso abarca desde los principios fundamentales y la evolución histórica del desarrollo de software hasta metodologías ágiles, DevOps, ingeniería de requerimientos, modelado UML, pruebas, y gestión de proyectos. Los estudiantes aprenderán a aplicar conceptos y técnicas modernas en el desarrollo profesional de software, con énfasis en la calidad, la ética profesional y las mejores prácticas de la industria.

Al finalizar la asignatura, el estudiante estará en capacidad de:

1. Aplicar metodologías ágiles y prácticas DevOps en proyectos de desarrollo de software empresarial.
2. Implementar procesos efectivos de ingeniería de requerimientos y modelado UML para sistemas complejos.

	<b>Universidad Internacional del Ecuador</b>	
	<b>SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD</b>	
	<b>MACROPROCESO GESTIÓN ACADÉMICA</b>	
<b>Versión: 1.2</b>	<b>PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE</b>	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
<b>Página 2 de 11</b>	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	

3. Diseñar arquitecturas de software utilizando patrones de diseño adecuados y principios de calidad.
4. Desarrollar estrategias de pruebas y mantenimiento que garanticen la calidad del software a lo largo de su ciclo de vida.
5. Gestionar proyectos de software aplicando técnicas de estimación, análisis de riesgos y trabajo en equipo.

#### **DATOS GENERALES DEL DOCENTE**

**Nombre del Profesor:** Charlie Alexander Cárdenas Toledo

**Grado Académico:** Magister en Ciencias y Tecnologías de la Computación

**e-mail:** chcardenasto@uide.edu.ec

**Teléfono fijo:** +593 22985600 Ext. 6008



**Teléfono móvil:** +593980762456

#### **LUGAR DONDE SE EMITE EL DOCUMENTO**


**Nombre del Coordinador/a Académico:** Mgs. Darío Javier Valarezo León



**Ubicación / Dirección del establecimiento educativo:** Calle Agustín Carrión Palacios entre Av. Salvador Bustamante Celi y Beethoven. Sector Jipiro


**Teléfono del establecimiento educativo:** +593 22985600 Ext. 6008



	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 3 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	



**CONTENIDOS DE LA ASIGNATURA** (Distribución por semana de la 1 a la 16 / 1 a 9 para idiomas. No se puede unir dos o tres semanas en una misma planificación)



UNIDAD	TEMA	SEMANA	NO. HORAS			RESULTADO DE APRENDIZAJE			HERRAMIENTA DE APRENDIZAJE	ACTIVIDAD CALIFICADA
			TEÓRICA	PRÁCTICA	AUTÓNOMO	Componente de docencia	Componente de práctica de aplicación y experimentación de los aprendizajes	Componente de aprendizaje autónomo		
Unidad 1: Fundamentos de la Ingeniería de Software	 Ingeniería de software hoy (nube, IA, producto) Ética, privacidad y uso responsable de IA Descubrimiento inicial del proyecto (contexto, stakeholders)	1	3	0	3	Explicar la ética y evolución del desarrollo de software	Analizar casos de implicaciones éticas en proyectos de software desde una perspectiva de responsabilidad social	Evaluar críticamente diferentes paradigmas de programación y su impacto en el desarrollo de software	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>GA-1.1:</b> Completar modulo 1 de ASC-5 (1) <b>PE-1.1:</b> dinámica de stakeholders y contexto del proyecto. (2) <b>TA-1.1:</b> visión de producto (1 pág) + mapa de stakeholders. (2)
Unidad 1: Fundamentos de la Ingeniería de Software	Reqs funcionales y no funcionales (rendimiento, seguridad, accesibilidad) Técnicas de elicitación (entrevistas, workshops, prototipos) Borrador de SRS / backlog inicial	2	3	0	3	Describir la crisis del software y modelos de procesos utilizando comunicación efectiva	Identificar soluciones a problemas derivados de la crisis del software mediante pensamiento crítico	Investigar autónomamente tendencias en lenguajes de programación	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>PE-1.2:</b> Taller de entrevistas y prototipo (2). <b>TA-1.2:</b> Borrador SRS (alcance, stakeholders, RF/RNF) (2) <b>GA-1.2:</b> Completar modulo 2 de ASC-5 (1) <b>GA:</b> Revisión Proyecto (0)



	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 4 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	



Unidad 2: Metodologías Ágiles	Historias de usuario, casos de uso y criterios de aceptación Validación (revisiones, protos, aceptación) Trazabilidad y administración de requerimientos	3	∞	0	∞	Explicar métodos ágiles y desarrollo dirigido por planes	Aplicar principios ágiles y técnicas de trabajo colaborativo en situaciones de desarrollo	Comparar críticamente enfoques de desarrollo ágil vs. planificado	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>PE-1.3:</b> Validación con prototipo clicable + checklist. Con IA(2) <b>TA-1.3:</b> Historias de usuario con criterios de aceptación (INVEST) (2). <b>GA-1.3:</b> Completar modulo 3 de ASC-5 (1)
Unidad 3: DevOps	Plan-driven vs ágil; Scrum/Kanban Grooming del backlog y planning Git básico y flujo de trabajo del equipo	4	∞	0	∞	Describir la cultura DevOps y su importancia en el ciclo de desarrollo	Implementar prácticas básicas de DevOps utilizando herramientas tecnológicas apropiadas	Analizar críticamente cómo escalar metodologías ágiles en diferentes contextos organizacionales	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>PE-1.4:</b> Simulación Scrum/Kanban y Git básico (issue + PR). (3) <b>TA-1.4:</b> Roadmap + tablero (épicas, DoR/DoD). (3) <b>GA-1.4:</b> Completar modulo 4 de ASC-5 (1)
Evaluación	Evaluación Diagnóstica	5	∞	0	∞	EVALUACIÓN SEMANA 1-4, PRESENTACIÓN PROYECTO - FASE I				<b>GA-1.5:</b> Evaluación Teórica (2) <b>GA-1.6:</b> Revisión Proyecto: SRS/backlog + planning y criterios de aceptación (6)
Unidad 4: Ingeniería de Requerimientos	 DDD ligero, ADRs Monolito modular vs microservicios; contratos y compatibilidad	6	∞	0	∞	Explicar requerimientos funcionales y no funcionales usando comunicación efectiva	Aplicar técnicas de elicitación de requerimientos y comunicación con stakeholders	Evaluar críticamente documentos de requerimientos considerando diversos contextos de usuario	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>GA-2.1:</b> Completar modulo 5 de ASC-5 (1) <b>TA-2.1:</b> 1 ADR + bosquejo de arquitectura (monolito modular vs microservicios).(2) <b>PE-2.1:</b> Modelado DDD ligero y ADR colaborativo. (2)


	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 5 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	



Unidad 5: Modelado	 Casos de uso, modelo de contexto Diagramas de clases para el dominio	7	∞	0	∞	Describir procesos de ingeniería de requerimientos con pensamiento sistemático	Implementar técnicas de validación de requerimientos utilizando pensamiento crítico	Desarrollar protocolos de validación mediante pensamiento visual y representaciones estructuradas	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>TA-2.2:</b> Diagramas: casos de uso + clases del dominio (v1). (2) <b>PE-2.2:</b> Revisión guiada de clases (consistencia con historias).(2) <b>GA-2.2:</b> Completar modulo 6 de ASC-5 (1) GA: Revisión Proyecto (0)
Unidad 6: Modelado de Software con UML I	Secuencia/actividad (escenarios clave) Componentes y despliegue orientado a cloud	8	∞	0	∞	Explicar modelos de contexto e interacción mediante representaciones visuales efectivas	Crear diagramas UML básicos transformando conceptos abstractos en modelos visuales	Construir modelos estructurales aplicando pensamiento computacional y visual	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>TA-2.3:</b> Diagrama de secuencia (escenario clave) + despliegue cloud (borrador). (2) <b>PE-2.3:</b> Componentes y despliegue orientado a cloud (contratos/compatibilidad). (2)
Unidad 4: Diseño y Arquitectura de Software con UML II	 Git avanzado (ramas cortas, PRs, code review) Platform Engineering / IDP y automatización básica	9	∞	0	∞	Describir decisiones arquitectónicas y patrones utilizando representaciones visuales avanzadas	Aplicar patrones arquitectónicos para resolver problemas de diseño complejos	Diseñar arquitecturas de aplicación considerando diversos contextos culturales y organizacionales	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>GA-2.3:</b> Completar modulo 7 de ASC-5 (1) <b>TA-2.4:</b> Estrategia Git (ramas cortas), plantilla de PR y CODEOWNERS. (3) <b>PE-2.4:</b> Pipeline mínimo (build + artefacto + revisión por PR). (3)
Evaluación	Evaluación Formativa	10	∞	0	∞	EVALUACIÓN SEMANA 6-9, PRESENTACIÓN PROYECTO - FASE II			Banco de preguntas Rúbricas de evaluación Guías de proyecto Tutor de IA	<b>GA-2.4:</b> Evaluación Teórica (3) <b>GA-2.5:</b> Revisión proyecto: modelos/ADR/pipeline (6)

	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 6 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	

Unidad 6: Diseño Orientado a Objetos con UML III	CI/CD moderno (pipelines como código, artefactos, feature flags) Seguridad: SAST/DAST/SCA, SBOM, firma, supply chain (SLSA)	11	∞	0	∞	Explicar patrones de diseño y su representación visual en UML	Implementar patrones de diseño en código aplicando pensamiento computacional	Demostrar autonomía en el aprendizaje de patrones de diseño avanzados	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>PE-3.1:</b> SAST/DAST/SCA en CI, SBOM y firma de artefacto (demo). (3) <b>TA-3-1:</b> SBOM + verificación de dependencias (3)
Unidad 7: Pruebas y DevOps	 Contenedores, Docker, Kubernetes/GitOps (visión práctica) Observabilidad: logs, métricas, trazas (OpenTelemetry), SLO/SLA	12	∞	0	∞	Describir pruebas de desarrollo y entrega continua utilizando comunicación técnica efectiva	Implementar pruebas unitarias y de integración aplicando pensamiento computacional y resolución de problemas	Investigar autónomamente herramientas avanzadas como Docker y Kubernetes	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>PE-3.2:</b> Contenedor + despliegue (local/K8s) y observabilidad (OTel: logs, métricas, trazas). <b>TA-3.2:</b> <i>runbook SRE con 2 SLO y 1 alerta crítica.</i> (3) GA: Revisión Proyecto (0)
Unidad 8: Calidad de Software	 Pirámide de pruebas, contratos, mutación y desempeño TDD/ATDD en el proyecto	13	∞	0	∞	Explicar pruebas de usuario y estándares de calidad evaluando fuentes de información relevantes	Aplicar métricas de calidad a código existente utilizando herramientas digitales apropiadas	Evaluar críticamente la calidad del software desde diversas perspectivas culturales y organizacionales	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>TA-3.3:</b> Plan de pruebas (pirámide), contratos/snapshot y métricas. (3) <b>PE-3.3:</b> Taller de pruebas de mutación y desempeño básico. (3)

	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 7 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	

Unidad 9: Mantenimiento de Software	Asistentes de código, generación de tests y refactor asistido Riesgos (licencias/datos) y verificación humana	14	∞	0	∞	Describir procesos de evolución y mantenimiento en diversos contextos organizacionales	Refactorizar sistemas heredados aplicando estrategias de resolución de problemas	Analizar cómo los sistemas de software impactan diferentes contextos culturales y regionales	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>PE-3.4:</b> Pair programming con asistente de código: generar tests/refactor + verificación humana (registro de prompts/cambios). (3)
Unidad 6: Mantenimiento y Gestión de Proyectos	 Mantenimiento, legacy y patrón estrangulador Migraciones, FinOps básico y monitoreo/alertas	15	∞	0	∞	Explicar gestión de riesgos y trabajo en equipo efectivo	Implementar un proyecto en herramientas de gestión demostrando colaboración efectiva	Desarrollar estrategias para el crecimiento profesional continuo en ingeniería de software	NotebookLM Recursos Bibliográficos Podcasts Casos de estudio Ejercicios prácticos Tarjetas didácticas Advanced Skill Certificate	<b>TA-3.2:</b> Plan de evolución: patrón estrangulador, migraciones, SLO/SLA, alertas y FinOps básico. (3)
Evaluación	Evaluación Sumativa	16	∞	0	∞	EVALUACIÓN SEMANA 1-15, PRESENTACIÓN PROYECTO - FASE III FINAL				GA-3.1: Evaluación Teórica (6) GA-3.2: Revisión Proyecto Final (10)



	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 8 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	

## COMPETENCIA TRANSVERSAL

Competencias que se desarrollarán en esta asignatura durante el presente periodo académico, con el resultado de aprendizaje y evidencia de evaluación.

Competencias a desarrollar	Resultados de aprendizaje	Evidencia de Evaluación
Comunicación efectiva	Elaborar documentación técnica clara y precisa (SRS, ADR, runbook, planes de prueba). Presentar hallazgos y decisiones técnicas con lenguaje profesional y fundamentado.	TA-1.1: Visión de producto y mapa de stakeholders. TA-1.3: Historias de usuario con criterios de aceptación. TA-2.1: ADR y bosquejo arquitectónico. TA-2.2: Diagramas de casos de uso y clases. TA-3.2: Runbook SRE con SLO y alertas. TA-3.3: Plan de pruebas y métricas. TA-3.4: Plan de evolución y FinOps.
Desarrollo personal y profesional	Participar activamente en dinámicas de equipo ágil (Scrum, Kanban, pair programming). Fomentar la ética, responsabilidad y aprendizaje continuo en el ciclo de vida del software.	PE-1.1: Dinámica de stakeholders. PE-1.4: Simulación Scrum/Kanban. PE-2.1: Modelado DDD y ADR colaborativo. PE-3.4: Pair programming asistido por IA. GA-1.6: Revisión Proyecto (SRS y backlog). GA-2.5: Revisión Proyecto (modelo y pipeline). GA-3.2: Revisión Proyecto Final.
Pensamiento crítico y sistémico	Analizar procesos de desarrollo y proponer mejoras basadas en métricas y retroalimentación. Evaluar decisiones de arquitectura, despliegue y calidad en entornos reales.	GA-1.1–GA-2.3: Módulos ASC-5 (aprendizaje aplicado). PE-2.3: Despliegue cloud y compatibilidad. PE-3.1: Seguridad en CI/CD y SBOM. PE-3.2: Contenedor + observabilidad. GA-3.1: Evaluación teórica final.
Cultura digital	Implementar herramientas de integración continua, control de versiones, pruebas y observabilidad.	PE-2.4: Pipeline mínimo CI/CD. PE-3.1: Integración SAST/DAST/SCA. PE-3.2: Despliegue K8s y observabilidad (OTel).





	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 9 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	

Competencias a desarrollar	Resultados de aprendizaje	Evidencia de Evaluación
	Aplicar DevOps y automatización en entornos empresariales.	TA-2.4: Estrategia Git y CODEOWNERS. TA-3.2: Runbook SRE.
Investigación y espíritu empresarial	Gestionar proyectos de software desde la ideación hasta el mantenimiento. Aplicar principios de calidad, métricas y patrones de diseño en la toma de decisiones técnicas.	GA-1.5: Evaluación teórica. GA-2.4: Evaluación teórica. GA-3.1: Evaluación teórica final. PE-3.3: Taller de pruebas de mutación y desempeño. TA-3.5: Whitepaper/tendencias con recomendaciones.

#### EVALUACIÓN:

Parámetros – Saber, Saber Hacer	Porcentaje
<b>Semana 5</b> <i>Gestión de trabajo autónomo:</i> <ul style="list-style-type: none"> <li>TA-1.1: Visión de producto (1 pág.) + mapa de stakeholders — 2</li> <li>TA-1.2: Borrador SRS (alcance, stakeholders, RF/RNF) — 2</li> <li>TA-1.3: Historias de usuario con criterios de aceptación (INVEST) — 2</li> <li>TA-1.4: Roadmap + tablero (épicas, DoR/DoD) — 3</li> </ul> <i>Gestión de practica y experimentación:</i> <ul style="list-style-type: none"> <li>PE-1.1: Dinámica de stakeholders y contexto del proyecto — 2</li> <li>PE-1.2: Taller de entrevistas y prototipo — 2</li> <li>PE-1.3: Validación con prototipo clicable + checklist — 2</li> <li>PE-1.4: Simulación Scrum/Kanban y Git básico (issue + PR) — 3</li> </ul> <i>Gestión de aprendizaje:</i> <ul style="list-style-type: none"> <li>GA-1.1: Completar módulo 1 de ASC-5 — 1</li> <li>GA-1.2: Completar módulo 2 de ASC-5 — 1</li> <li>GA-1.3: Completar módulo 3 de ASC-5 — 1</li> <li>GA-1.4: Completar módulo 4 de ASC-5 — 1</li> <li>GA-1.5: Evaluación Teórica — 2</li> <li>GA-1.6: Revisión Proyecto: SRS/backlog + planning y criterios — 6</li> </ul>	30%
<b>Semana 10</b> <i>Gestión de trabajo autónomo:</i> <ul style="list-style-type: none"> <li>TA-2.1: 1 ADR + bosquejo de arquitectura (monolito modular vs microservicios) — 2</li> <li>TA-2.2: Diagramas: casos de uso + clases del dominio (v1) — 2</li> <li>TA-2.3: Diagrama de secuencia + despliegue cloud (borrador) — 2</li> <li>TA-2.4: Estrategia Git (ramas cortas), plantilla de PR y CODEOWNERS — 3</li> </ul>	30%



	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 10 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	

<i>Gestión de practica y experimentación:</i> <ul style="list-style-type: none"> <li>PE-2.1: Modelado DDD ligero y ADR colaborativo — 2</li> <li>PE-2.2: Revisión guiada de clases (consistencia con historias) — 2</li> <li>PE-2.3: Componentes y despliegue orientado a cloud — 2</li> <li>PE-2.4: Pipeline mínimo (build + artefacto + revisión por PR) — 3</li> </ul> <i>Gestión de aprendizaje:</i> <ul style="list-style-type: none"> <li>GA-2.1: Completar módulo 5 de ASC-5 — 1</li> <li>GA-2.2: Completar módulo 6 de ASC-5 — 1</li> <li>GA-2.3: Completar módulo 7 de ASC-5 — 1</li> <li>GA-2.4: Evaluación Teórica — 3</li> <li>GA-2.5: Revisión Proyecto: modelo-s/ADR/pipeline — 6</li> </ul>	
<b>Semana 16</b> <i>Gestión de trabajo autónomo:</i> <ul style="list-style-type: none"> <li>TA-3.1: SBOM + verificación de dependencias — 3</li> <li>TA-3.2: Runbook SRE con 2 SLO y 1 alerta crítica — 3</li> <li>TA-3.3: Plan de pruebas (pirámide), contratos/snapshot y métricas — 3</li> <li>TA-3.4: Plan de evolución: patrón estrangulador, migraciones, SLO/SLA y FinOps — 3</li> <li>TA-3.5: Resumen crítico de un whitepaper/tendencia con 2 recomendaciones — 3</li> </ul> <i>Gestión de practica y experimentación:</i> <ul style="list-style-type: none"> <li>PE-3.1: SAST/DAST/SCA en CI, SBOM y firma de artefacto (demo) — 3</li> <li>PE-3.2: Contenedor + despliegue (local/K8s) y observabilidad (OTel) — 3</li> <li>PE-3.3: Taller de pruebas de mutación y desempeño básico — 3</li> <li>PE-3.4: Microservicios + tracing OpenTelemetry – 2.4</li> <li>PE-3.4: Pair programming con asistente de código: generación de tests/refactor — 3</li> </ul> <i>Gestión de aprendizaje:</i> <ul style="list-style-type: none"> <li>GA-3.1: Evaluación Teórica Final — 6</li> <li>GA-3.2: Revisión Proyecto Final — 10</li> </ul>	40%

## BIBLIOGRAFÍA:

### Bibliografía básica:

- Pressman, R. S. (2021). Ingeniería de software: un enfoque práctico. McGraw-Hill Interamericana
- Sommerville, I. (2021). Engineering software products: an introduction to modern software engineering. Pearson Education.
- Jiménez de Parga, C. (2021). *UML: arquitectura de aplicaciones en Java, C++ y Python*: (2 ed.). RA-MA Editorial. <https://0310s0jwj-y-https-elibro-net.proxy.uide.edu.ec/es/lc/uide/titulos/222720>

	Universidad Internacional del Ecuador	
	SISTEMA DE ASEGURAMIENTO INTERNO DE CALIDAD	
	MACROPROCESO GESTIÓN ACADÉMICA	
Versión: 1.2	PROCESO GESTIÓN DE ENSEÑANZA- APRENDIZAJE	Los cursos indicados con el logotipo de ASU indican cursos que se complementan con el contenido y los materiales del curso de ASU. El curso se impartirá en todas las sedes de la UIDE.
Página 11 de 11	Escuela de Ciencias de la Computación Ingeniería en Tecnologías de la Información Sílabo de LTI_05A_INS Ingeniería del Software	

- Llerena, L., Perez, H., Plazarte, J., Rodríguez, N., & Castro, J. W. (2024). Ethical Framework for the Software Development Process: A Systematic Mapping Study: Vol. 799 LNNS. Springer Science and Business Media Deutschland GmbH. [https://031060jw5-y-https-doi-org.proxy.uide.edu.ec/10.1007/978-3-031-45642-8\\_14](https://031060jw5-y-https-doi-org.proxy.uide.edu.ec/10.1007/978-3-031-45642-8_14)
- Copei, S., & Kosiol, J. (2024). DevOps Patterns: A Rapid Review: Vol. 14590 LNCS. Springer Science and Business Media Deutschland GmbH. [https://031060jw5-y-https-doi-org.proxy.uide.edu.ec/10.1007/978-3-031-66326-0\\_3](https://031060jw5-y-https-doi-org.proxy.uide.edu.ec/10.1007/978-3-031-66326-0_3)
- Canosa Ferreiro, A. J. (2024). SCRUM: teoría e Implementación práctica: (1 ed.). RA-MA Editorial. <https://0310s0jwj-y-https-elibro-net.proxy.uide.edu.ec/es/lc/uide/titulos/267660>

#### Bibliografía complementaria:

- Ljung, K., & Gonzalez-Huerta, J. (2022). “To Clean Code or Not to Clean Code” A Survey Among Practitioners: Vol. 13709 LNCS. Springer Science and Business Media Deutschland GmbH. [https://031060jw5-y-https-doi-org.proxy.uide.edu.ec/10.1007/978-3-031-21388-5\\_21](https://031060jw5-y-https-doi-org.proxy.uide.edu.ec/10.1007/978-3-031-21388-5_21)
- Carullo, G. (2020). Implementing Effective Code Reviews: How to Build and Maintain Clean Code. Apress Media LLC. <https://031060jw5-y-https-doi-org.proxy.uide.edu.ec/10.1007/978-1-4842-6162-0>
- Poulton, N. (2023). Docker Deep Dive. Nigel Poulton Ltd.

#### Fuentes ASU:

- Arizona State University. (2025). ASU: SER-216 Software Enterprise Processes and Quality. Information Technology Program. Online.
- Arizona State University. (2025). ASU: Inclusive Mindset: Tools for Building Positive Team Culture. Information Technology Program. Online.

#### RESPONSABILIDAD:

Elaborado Por:	Revisado Por:	Aprobado Por:
<b>Cargo:</b> Docente de la carrera de Ingeniería en Tecnologías de la información <b>Nombre:</b> Mgs. Charlie Cárdenas Toledo	<b>Cargo:</b> Coordinador académico de la carrera de Ingeniería en Tecnologías de la información <b>Nombre:</b> Mgs. Darío Javier Valarezo León	<b>Cargo:</b> Directora de la carrera de Ingeniería en Tecnologías de la información <b>Nombre:</b> Mgs. Lorena Elizabeth Conde Zhingre