

PROGRAMA DE LABORATORIO

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS

**Análisis y Diseño 1**

CÓDIGO:	0283	PUNTEO NETO LABORATORIO:	20
ESCUELA DE INGENIERÍA EN:	CIENCIAS Y SISTEMAS	ÁREA A LA QUE PERTENECE:	DESARROLLO DE SOFTWARE
PRE REQUISITO:	774 - SISTEMAS DE BASES DE DATOS 1	POST REQUISITO:	786 - SISTEMAS ORGANIZACIONALES Y GERENCIALES 1 785 - ANÁLISIS Y DISEÑO DE SISTEMAS 2 788 - SISTEMAS APLICADOS 1 2009 - PRÁCTICAS FINALES INGENIERÍA CIENCIAS Y SISTEMAS
CATEGORÍA:	OBLIGATORIO	VIGENCIA:	PRIMER SEMESTRE 2026

Descripción del Laboratorio

En el laboratorio de Análisis y diseño de sistemas 1 se complementa el contenido dado en la clase magistral, se imparten temas para que los estudiantes puedan implementar sus proyectos de forma mucho más eficiente, con la utilización de nuevas herramientas y técnicas que serán de mucha ayuda para los estudiantes.

Resumen de Ponderaciones y Tiempo de Auto-aprendizaje

TIPO	PONDERACIÓN	HORAS DE AUTO-APRENDIZAJE
Tareas (4)	12 pts	8
Exámenes Cortos (5)	10 pts	0
Prácticas	18 pts	30
Proyecto Fase 1	23 pts	100
Proyecto Fase 2	27 pts	100
Examen Final	10 pts	0
TOTAL	100 pts	238

Equipo Académico

Coordinador del Área

Nombre: Marlon Francisco Orellana Lopez	Correo electrónico: marlonorellana2005@gmail.com
---	--


Sección A

Docente

Nombre del Docente William Samuel Guevara Orellana	Correo electrónico del Docente willgo021983@gmail.com
--	---

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día		X		X		
Horario		07:10 - 08:50		07:10 - 08:50		
Lugar		MEET		MEET		

Tutor(es)

Nombre del Tutor	Byron Estuardo Solís González	
Correo electrónico institucional	3048067140116@ingenieria.usac.edu.gt	


Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día	X					
	Horario	17:20 - 19:00					
	Lugar	MEET					
Atención al Estudiante	Día						
	Horario						
	Lugar						

Sección B**Docente**

Nombre del Docente Edgar Francisco Rodas Robledo	Correo electrónico del Docente franciscorodas7@gmail.com
--	---

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día						X
Horario						07:10- 10:30
Lugar						MEET

Tutor(es)

Nombre del Tutor	Douglas Josué Martínez Huit	
Correo electrónico institucional	2504478190101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día	X					
	Horario	17:20 - 19:00					
	Lugar	MEET					
Atención al Estudiante	Día						
	Horario						
	Lugar						

Índice

Descripción del Laboratorio	1
Resumen de Ponderaciones y Tiempo de Auto-aprendizaje	2
Equipo Académico	2
Coordinador del Área	2
Sección A	2
Docente	2
Tutor(es)	3
Sección B	3
Docente	3
Tutor(es)	3
Competencias Vinculadas al Perfil del Egresado	8
Competencias Específicas	8
Competencias Generales	8
Competencias del Laboratorio	9
Competencia(s) Específica(s)	9
Competencia(s) General(es)	9
Diseño Didáctico	10
Sesión de Diagnóstico	10
Evaluación de conocimientos previos	10
Presentación del tutor	10
Presentación de los estudiantes	10
Presentación del programa del curso	10
Evaluación de conocimientos del laboratorio actual	10
Sesión No. 2, Unidad No. 1 – Control de Versiones	11
Área Actitudinal (Saber ser)	11
Área de Conocimiento (Saber)	11
Área de Habilidades (Saber Hacer)	11
Sesión No. 3, Unidad No. 2 – Diseño y Desarrollo de Sistemas	12
Área Actitudinal (Saber ser)	12
Área de Conocimiento (Saber)	12
Área de Habilidades (Saber Hacer)	12
Sesión No. 4, Unidad No. 2 – Metodologías de Desarrollo	13
Área Actitudinal (Saber ser)	13
Área de Conocimiento (Saber)	13
Área de Habilidades (Saber Hacer)	13
Sesión No. 5, Unidad No. 2 - Metodologías de Desarrollo	14
Área Actitudinal (Saber ser)	14
Área de Conocimiento (Saber)	14

Área de Habilidades (Saber Hacer)	14
Sesión No. 6, Unidad No. 2- Metodologías de Desarrollo	15
Área Actitudinal (Saber ser)	15
Área de Conocimiento (Saber)	15
Área de Habilidades (Saber Hacer)	15
Sesión No. 7, Unidad No. 3 - Diseño y Desarrollo de Sistemas	16
Área Actitudinal (Saber ser)	16
Área de Conocimiento (Saber)	16
Área de Habilidades (Saber Hacer)	16
Sesión No. 8, Unidad No. 3 - Diseño y Desarrollo de Sistemas	17
Área Actitudinal (Saber ser)	17
Área de Conocimiento (Saber)	17
Área de Habilidades (Saber Hacer)	17
Sesión No. 9, Unidad No. 3 - Diseño y Desarrollo de Sistemas	18
Área Actitudinal (Saber ser)	18
Área de Conocimiento (Saber)	18
Área de Habilidades (Saber Hacer)	18
Sesión No. 10, Unidad No. 3 - Diseño y Desarrollo de Sistemas	19
Área Actitudinal (Saber ser)	19
Área de Conocimiento (Saber)	19
Área de Habilidades (Saber Hacer)	19
Sesión No. 11, Unidad No. 4 - Automatización de ciclos de vida del software	20
Área Actitudinal (Saber ser)	20
Área de Conocimiento (Saber)	20
Área de Habilidades (Saber Hacer)	20
Sesión No. 12, Unidad No. 4 - Automatización de ciclos de vida del software	21
Área Actitudinal (Saber ser)	21
Área de Conocimiento (Saber)	21
Área de Habilidades (Saber Hacer)	21
Rúbrica de Evaluación	22
Normativa Académica y Ética del Curso	22
Bibliografía	23

Competencias Vinculadas al Perfil del Egresado

Competencias Específicas

No.	Competencia
1	Demuestra pensamiento crítico, actitud investigativa y rigor analítico en el planteamiento y la resolución de problemas complejos.
2	Toma decisiones profesionales con base en fundamentos teóricos, datos e información pertinente, válida y confiable.
3	Demuestra destreza y habilidad en la selección, uso y adaptación de herramientas metodológicas, tecnológicas, equipos especializados y en la lectura e interpretación de datos, pertinentes al contexto de su ejercicio profesional.
4	Lidera y colabora proactivamente en equipos de trabajo y en comunidades profesionales para el logro de objetivos y mejoramiento de la calidad de vida.

Competencias Generales

No.	Competencia
1	Aplica estándares de calidad, eficiencia y seguridad en la implementación adecuada de soluciones de software, hardware y TIC en general.
2	Construye soluciones integrales trabajando en forma colaborativa y propositiva en equipos interdisciplinarios, en forma presencial o utilizando plataformas virtuales.
3	Aplica principios básicos de ingeniería, ciencias de computación y sistemas de información y comunicación, en la formulación y resolución adecuada de problemas complejos.
4	Actualiza permanentemente sus conocimientos relacionados con TIC en general, apoyándose en las estrategias de aprendizaje apropiadas.

Competencias del Laboratorio

Competencia(s) Específica(s)

No.	Competencia	Nivel de Aprendizaje
1	Identifica requerimientos funcionales y no funcionales mediante análisis de procesos actuales, revisión de documentación y elaboración de casos de uso para asegurar que el sistema cumpla con las necesidades del usuario	Recordar
2	Valida la calidad del software mediante pruebas unitarias, pruebas E2E y herramientas de cobertura de código para detectar errores tempranos y mejorar la confiabilidad del sistema	Evaluar
3	Utiliza metodologías ágiles como scrum y tableros Kanban para entregar software funcional en cortos periodos de tiempo.	Evaluar
4	Aprende a elegir metodologías de desarrollo comparando los requerimientos que exige el proyecto para entregar software que satisfagan las necesidades solicitadas.	Comprender

Competencia(s) General(es)

No.	Competencia	Nivel de Aprendizaje
1	Genera soluciones de software aplicando buenas prácticas de análisis, control de versiones y metodologías ágiles para garantizar proyectos eficientes y colaborativos en entornos virtuales de trabajo	Crear
2	Integra herramientas de gestión de requerimientos, pruebas y despliegue continuo utilizando estándares de calidad de software y herramientas como Jira, Git y CI/CD con el fin de asegurar la trazabilidad, confiabilidad y funcionalidad del sistema	Aplicar
3	Coordina el desarrollo de soluciones grupales empleando metodologías ágiles y plataformas colaborativas para organizar y ejecutar reuniones que favorezcan la toma de decisiones y el cumplimiento de objetivos en entornos de trabajo en equipo.	Crear

Diseño Didáctico

Sesión de Diagnóstico

Evaluación de conocimientos previos

Se aplicará una actividad diagnóstica con el objetivo de identificar el nivel de conocimientos y habilidades que los estudiantes poseen al inicio del curso. No influye en la nota final, pero es obligatoria para todos los estudiantes.

Tipo de Actividad	Descripción
(puede ser un cuestionario, una dinámica participativa o un ejercicio práctico breve)	

Presentación del tutor

El tutor se presenta formalmente al grupo, compartiendo su formación académica, experiencia profesional y educativa, así como sus expectativas sobre el curso. También se abordan aspectos como normas de convivencia, canales de comunicación, disponibilidad para consultas y métodos de acompañamiento.

Presentación de los estudiantes

Se escogen un grupo de estudiantes al azar. En su presentación, se les pedirá que compartan información básica como su nombre, intereses personales o profesionales, experiencias previas relacionadas con el curso y sus expectativas. Esta actividad busca promover la interacción, el reconocimiento entre pares y la construcción de un entorno participativo y respetuoso.

Presentación del programa del curso

Se presenta el contenido del programa del curso, se aclaran dudas y se fomenta el compromiso del estudiante con su aprendizaje.

Evaluación de conocimientos del laboratorio actual

Se realiza una evaluación o práctica que permite conocer el grado de familiaridad de los estudiantes con las herramientas, entornos o competencias técnicas necesarias para el laboratorio actual.

Tipo de Actividad	Descripción
(Por ejemplo, uso de simuladores, entornos de desarrollo, hardware específico, etc. Puede incluir ejercicios prácticos, pruebas técnicas o autoevaluaciones guiadas.)	

Sesión No. 2, Unidad No. 1 – Control de Versiones

Área Actitudinal (Saber ser)

Nombre del valor: Honestidad
Se colocará un slide dentro de la presentación con un ejemplo de honestidad.

Área de Conocimiento (Saber)

Competencia(s)	
Genera soluciones de software aplicando buenas prácticas de análisis, control de versiones y metodologías ágiles para garantizar proyectos eficientes y colaborativos en entornos virtuales de trabajo	
Tema	Subtema
Introducción a los Sistemas de Control de Versiones	Introducción
	Sistema de control de versiones local
	Sistema de control de versiones centralizado
	Sistema de control de versiones distribuido
	Repositorios públicos
	Repositorios privados
	Herramientas
Git	Fases de Git
	Buenas Prácticas en Git
	Estructura de commits
	Comandos
	Ramas
GitFlow y Versionamiento Semántico	Ramas principales
	Ramas desarrollo
	Flujo de trabajo
	Versionamiento semántico
	Releases

Área de Habilidades (Saber Hacer)

Competencia
Genera soluciones de software aplicando buenas prácticas de análisis, control de versiones y metodologías ágiles para garantizar proyectos eficientes y colaborativos en entornos virtuales de trabajo

ágiles para garantizar proyectos eficientes y colaborativos en entornos virtuales de trabajo.	
Tipo de Actividad	Ponderación
Quizz	0

Sesión No. 3, Unidad No. 2 – Diseño y Desarrollo de Sistemas

Área Actitudinal (Saber ser)

Nombre del valor: Empatía
Se enseñará una imagen sobre cómo debemos ser empáticos ante las situaciones de los demás compañeros.

Área de Conocimiento (Saber)

Competencia(s)	
Aprende a elegir metodologías de desarrollo comparando los requerimientos que exige el proyecto para entregar software que satisfagan las necesidades solicitadas.	
Tema	Subtema
UX/UI	Introducción
	10 Principios de Jakob Nielsen
	Evolución de las interfaces de usuario
	Herramientas para UX/UI
	UX/UI con Frameworks

Área de Habilidades (Saber Hacer)

Competencia	
Genera soluciones de software aplicando buenas prácticas de análisis, control de versiones y metodologías ágiles para garantizar proyectos eficientes y colaborativos en entornos virtuales de trabajo.	
Tipo de Actividad	Ponderación
Tarea 1: Desarrollar API REST con versionamiento (tags y hotfix)	3

Sesión No. 4, Unidad No. 2 – Metodologías de Desarrollo

Área Actitudinal (Saber ser)

Nombre del valor: Responsabilidad
Se hablará durante 1 minuto sobre la responsabilidad en el cumplimiento de tareas y compromisos con el equipo.

Área de Conocimiento (Saber)

Competencia(s)	
Aprende a elegir metodologías de desarrollo comparando los requerimientos que exige el proyecto para entregar software que satisfagan las necesidades solicitadas.	
Tema	Subtema
Introducción a las Metodologías de Desarrollo	Introducción
	Historia
	Metodologías robustas
	Metodologías Ágiles
	Factores para la Selección de una Metodología
	Tendencias Actuales y Futuras

Área de Habilidades (Saber Hacer)

Competencia	
Genera soluciones de software aplicando buenas prácticas de análisis, control de versiones y metodologías ágiles para garantizar proyectos eficientes y colaborativos en entornos virtuales de trabajo.	
Tipo de Actividad	Ponderación
Corto 1 : Evaluación sobre metodologías de desarrollo	2

Sesión No. 5, Unidad No. 2 - Metodologías de Desarrollo

Área Actitudinal (Saber ser)

Nombre del valor: Solidaridad
Se dará un breve ejemplo hablado sobre cómo podemos apoyar a los compañeros si están atrasados y preguntar si necesitan ayuda.

Área de Conocimiento (Saber)

Competencia(s)	
Utiliza metodologías ágiles como scrum y tableros Kanban para entregar software funcional en cortos periodos de tiempo	
Tema	Subtema
Scrum	Introducción
	Roles
	Artefactos
	Eventos
	Pilares
	Técnicas de Estimación

Área de Habilidades (Saber Hacer)

Competencia	
Utiliza metodologías ágiles como scrum y tableros Kanban para entregar software funcional en cortos periodos de tiempo	
Tipo de Actividad	Ponderación
Quizz: Evaluación sobre Scrum	0

Sesión No. 6, Unidad No. 2- Metodologías de Desarrollo

Área Actitudinal (Saber ser)

Nombre del valor: Disciplina
Se dará un breve ejemplo sobre la importancia de la disciplina en el seguimiento de metodologías ágiles y la organización de tareas.

Área de Conocimiento (Saber)

Competencia(s)	
Integra herramientas de gestión de requerimientos, pruebas y despliegue continuo utilizando estándares de calidad de software y herramientas como Jira, Git y CI/CD con el fin de asegurar la trazabilidad, confiabilidad y funcionalidad del sistema	
Tema	Subtema
Herramientas de Gestión Ágil	Introducción
	Jira
	Trello
	Fundamentos de la Gestión Visual

Área de Habilidades (Saber Hacer)

Competencia	
Integra herramientas de gestión de requerimientos, pruebas y despliegue continuo utilizando estándares de calidad de software y herramientas como Jira, Git y CI/CD con el fin de asegurar la trazabilidad, confiabilidad y funcionalidad del sistema	
Tipo de Actividad	Ponderación
Corto 2: Evaluación sobre herramientas ágiles	2

Sesión No. 7, Unidad No. 3 - Diseño y Desarrollo de Sistemas

Área Actitudinal (Saber ser)

Nombre del valor: Tolerancia
Se hablará durante 1 minuto sobre la tolerancia con los compañeros de trabajo y sus diferentes perspectivas.

Área de Conocimiento (Saber)

Competencia(s)	
Identifica requerimientos funcionales y no funcionales mediante análisis de procesos actuales, revisión de documentación y elaboración de casos de uso para asegurar que el sistema cumpla con las necesidades del usuario	
Tema	Subtema
Requerimientos	Ciclo de vida del software
	Ingeniería de requerimientos
	Requerimientos funcionales
	Requerimientos no funcionales
Historias de usuarios	Formato de historias de usuario
	Criterios de aceptación
	Priorización
	Relación con el Product Backlog

Área de Habilidades (Saber Hacer)

Competencia	
Identifica requerimientos funcionales y no funcionales mediante análisis de procesos actuales, revisión de documentación y elaboración de casos de uso para asegurar que el sistema cumpla con las necesidades del usuario	
Tipo de Actividad	Ponderación
Tarea 2: Identificar requerimientos funcionales y no funcionales de un sitio e-commerce	3

Sesión No. 8, Unidad No. 3 - Diseño y Desarrollo de Sistemas

Área Actitudinal (Saber ser)

Nombre del valor: Paciencia
Se mostrará una caricatura animada sobre la paciencia.

Área de Conocimiento (Saber)

Competencia(s)	
Identifica requerimientos funcionales y no funcionales mediante análisis de procesos actuales, revisión de documentación y elaboración de casos de uso para asegurar que el sistema cumpla con las necesidades del usuario	
Tema	Subtema
Casos de uso	Introducción a Casos de Uso del Negocio (CUN)
	Actores
	Tipos de relaciones («extend» e «include»)
	CUN: Core
	CUN: Primera Descomposición
	CUN: Expandidos
	Plantilla de documentación
Modelado y Construcción de Sistemas	Introducción a UML
	Diagrama de clases
	Diagrama de secuencia
	Herramientas de modelado

Área de Habilidades (Saber Hacer)

Competencia	
Identifica requerimientos funcionales y no funcionales mediante análisis de procesos actuales, revisión de documentación y elaboración de casos de uso para asegurar que el sistema cumpla con las necesidades del usuario	
Tipo de Actividad	Ponderación
Tarea 3: Conferencia del curso o investigación sobre Diagramas de Componentes y Secuencia	3

Sesión No. 9, Unidad No. 3 - Diseño y Desarrollo de Sistemas

Área Actitudinal (Saber ser)

Nombre del valor: Responsabilidad
Se hablará sobre la responsabilidad de entregar software de calidad y con las pruebas adecuadas.

Área de Conocimiento (Saber)

Competencia(s)	
Valida la calidad del software mediante pruebas unitarias, pruebas E2E y herramientas de cobertura de código para detectar errores tempranos y mejorar la confiabilidad del sistema.	
Identifica requerimientos funcionales y no funcionales mediante análisis de procesos actuales, revisión de documentación y elaboración de casos de uso para asegurar que el sistema cumpla con las necesidades del usuario	
Tema	Subtema
Garantía de Calidad del Software	Pruebas funcionales y no funcionales
	Pirámide de pruebas
	Principios SOLID
	Pruebas unitarias
	Mocks, stubs y fakes
	Cobertura de código y métricas
	Refactorización de código

Área de Habilidades (Saber Hacer)

Competencia	
Identifica requerimientos funcionales y no funcionales mediante análisis de procesos actuales, revisión de documentación y elaboración de casos de uso para asegurar que el sistema cumpla con las necesidades del usuario	
Tipo de Actividad	Ponderación
Corto 3: Evaluación sobre pruebas y calidad de software	2

Sesión No. 10, Unidad No. 3 - Diseño y Desarrollo de Sistemas

Área Actitudinal (Saber ser)

Nombre del valor: Integridad
Se mostrará un ejemplo sobre cómo la integridad en las pruebas garantiza software confiable.

Área de Conocimiento (Saber)

Competencia(s)	
Cooperación, en un slide se mostrará un ejemplo sobre como cooperando unos con otros se logran grandes cosas como terminar el proyecto a tiempo.	
Tema	Subtema
Aseguramiento Integral del Software	Pruebas No Funcionales
	Tipos de Pruebas
	Pruebas End to End
	Cypress

Área de Habilidades (Saber Hacer)

Competencia	
Identifica requerimientos funcionales y no funcionales mediante análisis de procesos actuales, revisión de documentación y elaboración de casos de uso para asegurar que el sistema cumpla con las necesidades del usuario	
Tipo de Actividad	Ponderación
Tarea 4: Desarrollar prueba E2E con Cypress en saucedemo.com)	3

Sesión No. 11, Unidad No. 4 - Automatización de ciclos de vida del software

Área Actitudinal (Saber ser)

Nombre del valor: Amabilidad
Amabilidad, se dará una pequeña charla de 1 minuto sobre cómo pedir las cosas con amabilidad hacen sentir bien a la otra persona y no solo exigir cosas.

Área de Conocimiento (Saber)

Competencia(s)	
Integra herramientas de gestión de requerimientos, pruebas y despliegue continuo utilizando estándares de calidad de software y herramientas como Jira, Git y CI/CD con el fin de asegurar la trazabilidad, confiabilidad y funcionalidad del sistema.	
Tema	Subtema
Docker	Introducción a docker
	Creación y gestión de imágenes
	Creación y gestión de contenedores
	Docker compose
	Container Registry

Área de Habilidades (Saber Hacer)

Competencia	
Integra herramientas de gestión de requerimientos, pruebas y despliegue continuo utilizando estándares de calidad de software y herramientas como Jira, Git y CI/CD con el fin de asegurar la trazabilidad, confiabilidad y funcionalidad del sistema.	
Tipo de Actividad	Ponderación
Corto 4 : Evaluación sobre Docker y contenedores	2

Sesión No. 12, Unidad No. 4 - Automatización de ciclos de vida del software

Área Actitudinal (Saber ser)

Nombre del valor: Colaboración
Se motivará a los estudiantes sobre la importancia de la colaboración en equipos DevOps para lograr entregas exitosas.

Área de Conocimiento (Saber)

Competencia(s)	
Integra herramientas de gestión de requerimientos, pruebas y despliegue continuo utilizando estándares de calidad de software y herramientas como Jira, Git y CI/CD con el fin de asegurar la trazabilidad, confiabilidad y funcionalidad del sistema	
Tema	Subtema
DevOps	Introducción
	Componentes de CI/CD
	Principios de integración continua
	Beneficios de entrega continua
	Herramientas de integración continua
	Configuración de pipelines CI
	Ejecución de pruebas automatizadas

Área de Habilidades (Saber Hacer)

Competencia	
Integra herramientas de gestión de requerimientos, pruebas y despliegue continuo utilizando estándares de calidad de software y herramientas como Jira, Git y CI/CD con el fin de asegurar la trazabilidad, confiabilidad y funcionalidad del sistema.	
Tipo de Actividad	Ponderación
Corto 5: Evaluación sobre CI/CD	2

Rúbrica de Evaluación

Cada una de las actividades del laboratorio (proyectos, prácticas, tareas y otras) cuenta con una rúbrica de evaluación específica, la cual está detallada en el documento que se entrega al estudiante al momento de asignar la actividad. Estas rúbricas describen los criterios de evaluación, niveles de desempeño esperados y la ponderación correspondiente de cada aspecto evaluado.

Es **responsabilidad del estudiante** leer detenidamente la rúbrica asignada antes de iniciar el desarrollo de la actividad. Comprender los criterios de evaluación no solo permite orientar adecuadamente el trabajo, sino también mejorar el desempeño académico y fomentar la autorregulación del aprendizaje.

En caso de no recibir la rúbrica al momento de la asignación, el estudiante **debe solicitarla directamente al tutor académico**, ya que constituye una herramienta esencial para el cumplimiento de los objetivos de aprendizaje y la evaluación transparente.

Normativa Académica y Ética del Curso

En concordancia con el perfil del estudiante de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, se espera un alto nivel de compromiso con la excelencia académica y la ética profesional. Por ello, que se establece los siguientes lineamientos de carácter obligatorio que regulan el comportamiento académico del estudiante:

Plagio y copias

- Todo proyecto será sometido a verificación para confirmar su autoría y originalidad, con la finalidad de evitar cualquier plagio, copia o que la actividad no haya sido realizada por el estudiante.
- Cualquier evidencia de lo antes descrito en las distintas actividades será sancionada con una calificación de 0 (cero) y el caso será reportado al Docente quien a su vez informará a la Escuela de Ciencias y Sistemas para su seguimiento institucional.

Prórrogas y reposiciones

- No se otorgarán prórrogas para entregas de actividades.
- No se permitirá la reposición de proyectos bajo ninguna circunstancia.

Requisitos para evaluación final del curso

- Es obligatorio aprobar el laboratorio para tener derecho a la evaluación final del curso.
- La calificación de prácticas, proyectos y otras actividades que se indique será asignada de forma presencial, en la fecha y hora establecidas por el tutor académico.

Asistencia

- Para obtener la nota del laboratorio, se requiere un mínimo del 80% de asistencia a las sesiones de laboratorio.
- En caso de inasistencia, sólo se aceptarán justificaciones válidas respaldadas por constancia oficial.

Entregas

- No se aceptarán entregas tardías de tareas, prácticas, exámenes cortos, exámenes finales o proyectos sin justificación.

Medio oficial de entrega

- La plataforma UEDI de la Facultad será el único medio oficial para la entrega de actividades del curso.

Bibliografía

- “Ingeniería de software”, Novena Edición, PEARSON EDUCACIÓN, México, 2011
- “Version Control with Subversion”, Ben Collins-Sussman, Brian W. Fitzpatrick, C. Michael Pilato.
- “Continuous Delivery”, Jez Humble, David Farley