

PROGRAMA DE LABORATORIO

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

**ANÁLISIS Y DISEÑO DE SISTEMAS 1**

CÓDIGO:	0283	PONDERACIÓN:	6
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:	SEGUNDO SEMESTRE 2025
HORAS POR SEMANA DEL CURSO:	3.4	HORAS POR SEMANA DEL LABORATORIO:	1.7
HORAS DE AUTOAPRENDIZAJE:	4	TOTAL DE HORAS DE APRENDIZAJE:	4
CATEDRÁTICO (A):	WILLIAM SAMUEL GUEVARA ORELLANA	AUXILIAR:	EDDY FERNANDO DÍAZ GALINDO
EDIFICIO:	MEET HÍBRIDO	SECCIÓN:	A
SALÓN DEL CURSO:	MEET HÍBRIDO	SALON DEL LABORATORIO:	MEET HÍBRIDO
DÍAS QUE SE IMPARTE EL CURSO:	MARTES Y JUEVES	DÍAS QUE SE IMPARTE EL LABORATORIO:	LUNES
HORARIO DEL CURSO:	07:10 - 08:50	HORARIO DEL LABORATORIO:	17:20 - 19:00

Breve 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.

Índice

Competencias Vinculadas al Perfil del Egresado.....	4
Competencias Específicas.....	4
Competencias Generales.....	4
Competencias del Laboratorio.....	5
Competencia(s) Específica(s).....	5
Competencia(s) General(es).....	6
Diseño Didáctico por Competencias.....	6
Sesión de Diagnóstico.....	7
Evaluación de conocimientos previos.....	7
Presentación del tutor.....	7
Presentación de los estudiantes.....	7
Presentación del programa del curso.....	7
Evaluación de conocimientos del laboratorio actual.....	7
Sesión No. 1, Unidad No. 1 - Control de Versiones.....	8
Conocimiento (Saber).....	8
Habilidades (Saber Hacer).....	8
Sesión No. 2, Unidad No. 1 - Control de Versiones.....	9
Valor de la semana (Saber ser).....	9
Conocimiento (Saber).....	9
Habilidades (Saber Hacer).....	9
Sesión No. 3, Unidad No. 1 - Control de Versiones.....	10
Valor de la semana (Saber ser).....	10
Conocimiento (Saber).....	10
Habilidades (Saber Hacer).....	10
Sesión No. 4, Unidad No. 2. - Metodologías de Desarrollo.....	11
Valor de la semana (Saber ser).....	11
Conocimiento (Saber).....	11
Habilidades (Saber Hacer).....	11
Sesión No. 5, Unidad No. 2. - Metodologías de Desarrollo.....	12
Valor de la semana (Saber ser).....	12
Conocimiento (Saber).....	12
Habilidades (Saber Hacer).....	12
Sesión No. 6, Unidad No. 2. - Metodologías de Desarrollo.....	13
Valor de la semana (Saber ser).....	13
Conocimiento (Saber).....	13
Habilidades (Saber Hacer).....	13

Sesión No. 7, Unidad No. 3 - Diseño y Desarrollo de Sistemas.....	14
Valor de la semana (Saber ser).....	14
Conocimiento (Saber).....	14
Habilidades (Saber Hacer).....	14
Sesión No. 8, Unidad No. 3 - Diseño y Desarrollo de Sistemas.....	15
Valor de la semana (Saber ser).....	15
Conocimiento (Saber).....	15
Habilidades (Saber Hacer).....	15
Sesión No. 9, Unidad No. 3 - Diseño y Desarrollo de Sistemas.....	16
Valor de la semana (Saber ser).....	16
Conocimiento (Saber).....	16
Habilidades (Saber Hacer).....	16
Sesión No. 10, Unidad No. 4 - Automatización de ciclos de vida del software.....	17
Valor de la semana (Saber ser).....	17
Conocimiento (Saber).....	17
Habilidades (Saber Hacer).....	17
Sesión No. 11, Unidad No. 4 - Automatización de ciclos de vida del software.....	18
Valor de la semana (Saber ser).....	18
Conocimiento (Saber).....	18
Habilidades (Saber Hacer).....	18
Tiempo de Auto-aprendizaje.....	19
Rúbrica de Evaluación.....	19
Resumen de Ponderaciones.....	19
Normativa Académica y Ética del Curso.....	20
Equipo Académico.....	21
Coordinador del Área.....	21
Sección A.....	21
Sección B.....	22
Sección C.....	23
Bibliografía.....	24

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
4	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 por Competencias

Esta sección organiza las sesiones del laboratorio en función de las competencias que el estudiante debe desarrollar. Cada clase incluye valores (saber ser), contenidos teóricos (saber) y habilidades prácticas (saber hacer), permitiendo un aprendizaje integral y aplicado. Las actividades están alineadas con los objetivos del curso y el perfil del egresado.

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. 1, Unidad No. 1 - Control de Versiones

Nombre: Respeto
Se colocará un slide dentro de la presentación con un ejemplo de respeto.

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

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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.	Corto / Quizz	1

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

Valor de la semana (Saber ser)

Nombre: Honestidad
Se mostrará una imagen donde se muestre la honestidad cuando se trabaja en grupos.

Conocimiento (Saber)

Competencia(s)	
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.	
Tema	Subtema
Git	Introducción
	Fases de Git
	Buenas Prácticas en Git
	Estructura de commits
	Comandos
	Ramas

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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	Ejercicio (Creación de cuenta en github/gitlab y comandos básicos)	1

Sesión No. 3, Unidad No. 1 - Control de Versiones

Valor de la semana (Saber ser)

Nombre: Tolerancia
Se hablará durante 1 minuto al iniciar la clase sobre la tolerancia con los compañeros de trabajo.

Conocimiento (Saber)

Competencia(s)	
Integra componentes, servicios y tecnologías de software mediante el uso de APIs, servicios web y herramientas de integración continua, asegurando la interoperabilidad, cohesión funcional y eficiencia en la entrega de soluciones informáticas.	
Tema	Subtema
GitFlow y Versionamiento Semántico	Introducción
	Ramas principales
	Ramas desarrollo
	Flujo de trabajo
	Versionamiento semántico
	Releases

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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	Práctica (Creación de ramas en repositorio)	1

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

Valor de la semana (Saber ser)

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

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

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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.	Corto 1	3

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

Valor de la semana (Saber ser)

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

Conocimiento (Saber)

Competencia	
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

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Utiliza metodologías ágiles como scrum y tableros Kanban para entregar software funcional en cortos periodos de tiempo	Quizz	1

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

Valor de la semana (Saber ser)

Nombre: Compromiso
Se hará una encuesta para verificar que tan comprometidos están los integrantes de los grupos para terminar el curso.

Conocimiento (Saber)

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	
Tema	Subtema
Herramientas de Gestión Ágil	Introducción
	Jira
	Trello
	Selección de herramientas adecuadas

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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	Ejercicio (crear tablero en jira/trello)	1

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

Valor de la semana (Saber ser)

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

Conocimiento (Saber)

Competencia	
Aprende a elegir metodologías de desarrollo comparando los requerimientos que exige el proyecto para entregar software que satisfagan las necesidades solicitadas. 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	
Tema	Subtema
Modelado y Construcción de Sistemas	Ciclo de vida del software
	Ingeniería de requerimientos
	Requerimientos funcionales y no funcionales
	Historias de usuarios
	Casos de uso
	Modelado de sistemas con UML

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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	Quizz	1

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

Valor de la semana (Saber ser)

Nombre: Gratitude
Gratitude, se hablará durante unos 30 segundos sobre cómo deben ser agradecidos con sus compañeros si estos les ayudaron en algo.

Conocimiento (Saber)

Competencia	
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

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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	Corto 3	3

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

Valor de la semana (Saber ser)

Nombre: Cooperación
Cooperación, en un slide se mostrará un ejemplo sobre como cooperando unos con otros se logran grandes cosas como terminar el proyecto a tiempo.

Conocimiento (Saber)

Competencia	
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	
Tema	Subtema
Aseguramiento Integral del Software	Pruebas No Funcionales
	Tipos de Pruebas
	Pruebas End to End
	Cypress

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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	Práctica (Genere una prueba no funcional en cypress)	1

Sesión No. 10, Unidad No. 4 - Automatización de ciclos de vida del software

Valor de la semana (Saber ser)

Nombre: 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.

Conocimiento (Saber)

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.	
Tema	Subtema
Docker	Introducción a docker
	Creación y gestión de imágenes
	Creación y gestión de contenedores
	Docker compose
	Container Registry

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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.	Ejercicio (crear imagen de backend)	1

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

Valor de la semana (Saber ser)

Nombre: Perseverancia
Perseverancia, se dará una plática de 1 minuto animando a los alumnos a no desistir y seguir perseverando en la carrera porque ya están cerca de la meta.

Conocimiento (Saber)

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	
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

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
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.	Corto 3	3

Tiempo de Auto-aprendizaje

Tipo	Horas de Auto-aprendizaje
Proyectos	200
Prácticas	30
Tareas	5.5
Total	235.5

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.

Resumen de Ponderaciones

Tipo	Valor
Actividades en Clase	17
Proyectos	50
Prácticas	15
Tareas	8
Examen Final	10
Total	100

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.

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: 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

Nombre del Tutor	Eddy Fernando Díaz Galindo	
Correo electrónico institucional	3044397700114@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 franciscorodas7@gmail.com
--	---

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

Tutor(es)

Nombre del Tutor	Ronald Geovany Ordoñez Xiloj	
Correo electrónico institucional	2326310690411@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						

Bibliografía

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