



SÍLABO PRUEBAS DE SOFTWARE

ÁREA CURRICULAR: INGENIERÍA DE SOFTWARE

I. DATOS GENERALES

1.1	Departamento Académico	:	Ingeniería y Arquitectura
1.2	Semestre Académico	:	2019-I
1.3	Código de la asignatura	:	091124E3040
1.4	Ciclo	:	Electivo de especialidad
1.5	Créditos	:	4
1.6	Horas semanales totales	:	8
	Horas lectivas (Total, Teoría, Práctica)	:	4 (T=4, P=0, L=0))
	Horas de trabajo independiente	:	4
1.7	Condición del curso	:	Electivo de Especialidad
1.8	Requisito(s)	:	09013707050 Ingeniería de Software II
1.9	Docente	:	Mg. Yamela Amparo Valenzuela Tasayco

II. SUMILLA

El curso es de naturaleza formación especializada dirigido a brindar a los estudiantes los distintos roles involucrados en el desarrollo de software y pruebas, una visión general de los conceptos, del proceso de pruebas y del modelo de madurez de pruebas, que faciliten la implementación de prácticas dentro de la organización. El sílabo posee 4 factores claves de aprendizaje: Recordar, Entender, Aplicar, Analizar.

Unidades: Fundamentos de pruebas a través del ciclo de desarrollo del Software – Técnicas estáticas – Técnicas de diseño de pruebas – Gestión de pruebas y Herramientas de pruebas.

III. COMPETENCIAS Y SUS COMPONENTES COMPRENDIDOS EN LA ASIGNATURA

3.1 Competencias

- Analiza un problema e identifica y define los requerimientos apropiados para su solución.
- Trabaja con efectividad en equipos para lograr una meta común.
- Se comunica con efectividad con un rango de audiencias.
- Usa técnicas, destrezas, y herramientas modernas necesarias para la práctica de la computación.
- Comprende de los procesos que soportan la entrega y la administración de los sistemas de información dentro de un entorno específico de aplicación.

3.2 Componentes

• Capacidades

- Explica las etapas del proceso del ciclo de vida de pruebas de software.
- Realiza el diagnóstico del modelo de negocio de una empresa, en la cual se implementará el Sistema de Información.

Contenidos actitudinales

- Participa en el desarrollo de los casos implementando las etapas del ciclo del proceso de pruebas de software.
- Emite informes en relación a la implementación de las mejores prácticas de las pruebas de software.

IV. PROGRAMACIÓN DE CONTENIDOS

UNIDAD I: FUNDAMENTOS DE PRUEBA A TRAVÉS DEL CICLO DE DESARROLLO DE SOFTWARE

• **CAPACIDAD:**

Describir con ejemplos la manera en que un defecto en el software puede causar daño a una persona, al ambiente o a una empresa.
Distinguir entre la raíz la causa de un defecto y sus efectos. Dar razones porqué probar es necesario y dar ejemplos
Explicar y comparar los términos de error, defecto, avería, fallo y los correspondientes términos de error y bugs utilizando ejemplos
Diferenciar testing y debugging
Contrastar la mentalidad de un probador y un desarrollador
Explicar la relación entre desarrollo, pruebas de actividades y productos de trabajo en el ciclo de vida de desarrollo brindando ejemplos utilizando proyectos y tipos de producto.
Reconocer el hecho de que los modelos de desarrollo de software se deben adaptar al contexto de proyecto y a las características del producto.
Comparar los 4 tipos de prueba de software (funcional, no funcional, estructural y cambio relacionado mediante ejemplos.
Identificar y describir tipos de pruebas no funcionales basadas en requerimientos no funcionales.
Identificar y describir tipos de pruebas basados en el análisis de una estructura de sistema de software o arquitectura.
Describir el propósito de una prueba de confirmación y regresión,
Describir el rol de una prueba de regresión y análisis de impacto en mantenimiento

SEMANA	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	ACTIVIDAD DE APRENDIZAJE	HORAS	
				L	T.I.
1	Por qué probar es necesario. Qué es probar Principios de pruebas. Proceso fundamental de pruebas. Psicología de pruebas. Examen de entrada Plantillas de pruebas , Virtual Ver las relaciones de ISOS en relación al desarrollo o mantenimiento de las pruebas de software	Estructura las pruebas de software	Lectivas (L): Desarrollo del tema – 3 h · Ejercicios en aula -1 horas De trabajo Independiente (T.I): · Resolución tareas - 1 hora · Trabajo de investigación – 3 hora	4	4
2	Modelos de desarrollo de software. Niveles de pruebas. Ejercicios aplicación de pruebas Virtual Ver las relaciones de ISOS en relación al desarrollo o mantenimiento de las pruebas de software	Analiza los diversos modelos de desarrollo de software	Lectivas (L): · Introducción al tema - 1 hora · Desarrollo del tema - 3 De trabajo Independiente (T.I): · Resolución tareas - 1 hora · Trabajo de investigación – 3 hora	4	4
3	Tipos de pruebas: Los objetivos de las pruebas. Pruebas de mantenimiento. Primera Práctica Calificada Virtual Relaciones con los tipos de pruebas de software	Analiza los tipos de pruebas de software.	Lectivas (L): · Desarrollo del tema – 3 h · Ejercicios en aula -1 horas De trabajo Independiente (T.I): · Resolución tareas - 1 hora · Trabajo de investigación grupal – 3 hora	4	4

UNIDAD II: TÉCNICAS ESTÁTICAS

- **CAPACIDAD:**
- Reconocer los productos de software que puedan ser examinados mediante las distintas técnicas estáticas
- Describir la importancia y el valor de considerar técnicas estáticas para la evaluación de productos de software.
- Explicar la diferencia entre técnicas estáticas y dinámicas, considerando objetivos, tipos de defectos a ser identificados y el rol de esas técnicas en el entorno del ciclo de vida del software.
- Explicar las diferencias entre distintos tipos de revisión: informal, técnico, periódico e inspección.
- Explicar los factores de revisiones satisfactorias.
- Describir utilizando ejemplos, los distintos beneficios de análisis estático.
- Diferenciar entre una especificación de diseño de prueba, especificación de caso de prueba y especificación de procedimiento de prueba.

SEMANA	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	ACTIVIDAD DE APRENDIZAJE	HORAS	
				L	T.I.
4	Revisión y pruebas de proceso. Revisión de proceso 1. Revisión de proceso 2. Ejercicios Revisión y pruebas de proceso Virtual Información complementaria de pruebas de proceso	Revisión y pruebas del proceso.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema - 2 · Ejercicios en aula -1 horas De trabajo Independiente (T.I.): <ul style="list-style-type: none"> · Resolución tareas - 1 hora · Trabajo de investigación – 1 hora · Trabajo grupal: 2 horas 	4	4
5	Análisis estático mediante herramientas. Práctica dirigida Virtual Funcionamientos de herramientas análisis estático.	Aplicación de herramientas	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema - 2 · Ejercicios en aula -1 horas De trabajo Independiente (T.I.): <ul style="list-style-type: none"> · Resolución tareas - 1 hora · Trabajo de investigación – 1 hora · Trabajo grupal: 2 horas 	4	4
6	Identificar condiciones de prueba y casos de diseño de pruebas. Categorías de técnicas de diseño de pruebas. Aplicación casos de diseño de pruebas Virtual Plantillas	Analizar la condiciones de las pruebas de software	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema - 2 · Ejercicios en aula -1 horas De trabajo Independiente (T.I.): <ul style="list-style-type: none"> · Resolución tareas - 1 hora · Trabajo de investigación – 1 hora · Trabajo grupal: 2 horas 	4	4

UNIDAD III: TÉCNICA DE DISEÑO DE PRUEBAS

- **CAPACIDAD:**
- Comparar los términos: condición de pruebas, caso de prueba y procedimiento de pruebas.
- Evaluar la calidad de los casos de prueba en términos de trazabilidad a los requerimientos y resultados esperados.
- Traducir casos de prueba en una especificación de procedimiento de prueba en un nivel de detalle relevante para el conocimiento de los testers.
- Explicar las características, singularidades y diferencias entre pruebas basadas en especificación, pruebas basadas en estructura y pruebas basadas en experiencia.
- Escribir casos de prueba desde modelos de software dados utilizando particiones equivalentes, análisis de valor límite, tablas de decisión y transición de estados en tablas y/o diagramas
- Explicar el propósito principal de las 4 técnicas de pruebas, que nivel y tipo de testing podrían usar y cómo la cobertura se puede medir.
- Explicar el concepto de casos de prueba y sus beneficios.
- Evaluar la declaración y cobertura de decisión para la integridad de las pruebas con respecto a los criterios de salida definidos.

SEMANA	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	ACTIVIDAD DE APRENDIZAJE	HORAS	
				L	T.I.
7	Técnicas de especificación basada en la caja negra. Segunda Práctica Calificada Virtual Aplicación de pruebas basada en caja negra.	Analizar las condiciones de las pruebas de software	Lectivas (L): Introducción al tema - 1 hora Desarrollo del tema - 2 Ejercicios en aula -1 horas De trabajo Independiente (T.I): Resolución tareas - 1 hora Trabajo de investigación – 3 hora	4	4
8	Examen Parcial				
9	Estructura base o técnicas de caja blanca. Técnicas de experiencia base. Ejercicios. Virtual Aplicación de pruebas basada en caja blanca.	Analizar las condiciones de las pruebas de software	Lectivas (L): · Introducción al tema - 1 hora · Desarrollo del tema - 2 · Ejercicios en aula -1 horas De trabajo Independiente (T.I): · Resolución tareas - 1 hora · Trabajo de investigación – 3 hora	4	4
10	Eligiendo una técnica de prueba. Tercera Práctica Calificada Virtual Aplicación de pruebas basada en caja blanca	Analizar las condiciones de las pruebas de software	Lectivas (L): · Introducción al tema - 1 hora · Desarrollo del tema - 2 · Ejercicios en aula -1 horas De trabajo Independiente (T.I): · Resolución tareas - 1 hora · Trabajo de investigación – 1 hora · Trabajo grupal: 2 horas	4	4

UNIDAD IV: GESTIÓN DE PRUEBAS Y HERRAMIENTAS DE PRUEBAS

- **CAPACIDAD:**
- Reconocer la importancia de una prueba independiente.
- Reconocer los diferentes miembros de equipo a ser considerados para la creación de un equipo de pruebas.
- Reconocer los distintos niveles y objetivos de un planificador de pruebas.
- Resumir el propósito y contenido de un plan de pruebas, especificación de diseño de pruebas y documentos de procedimiento de pruebas acorde con el “Standard for Software Test Documentation” (IEEE Std 829-1998)
- Clasificar distintos tipos de herramientas de prueba según su propósito y para las actividades de los procesos de pruebas y el ciclo de vida del software.
- Resumir los beneficios potenciales y riesgos de una prueba de automatización y soporte para pruebas.
- Recordar consideraciones especiales para herramientas de ejecución, análisis estático y herramientas de gestión de pruebas (K1).
- Establecer los principales principios de introducir una herramienta en la organización

SEMANA	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	ACTIVIDAD DE APRENDIZAJE	HORAS	
				L	T.I.
11	Organización de pruebas. Planes de prueba, estimaciones y estrategias. Virtual Plantillas de Plan de pruebas.	Analizar las condiciones de las pruebas de software	Lectivas (L): Introducción al tema - 1 hora Desarrollo del tema - 2 Ejercicios en aula -1 horas De trabajo Independiente (T.I): Resolución tareas - 1 hora Trabajo de investigación – 1 hora Trabajo grupal: 2 horas	4	4
12	Monitoreo del progreso de las pruebas y control. Gestión de la configuración. Riesgos y pruebas. Virtual Herramienta. Gestión de Configuración	Analizar las condiciones de las pruebas de software	Lectivas (L): · Introducción al tema - 1 hora · Desarrollo del tema - 2 · Ejercicios en aula -1 horas De trabajo Independiente (T.I): Resolución tareas - 1 hora Trabajo de investigación – 1 hora Trabajo grupal: 2 horas	4	4
13	Gestión de incidentes. Ejercicios, práctica dirigida 2. Virtual Herramienta Gestión de incidentes.	Analiza la Gestión de incidentes.	Lectivas (L): · Introducción al tema - 1 hora · Desarrollo del tema - 2 · Ejercicios en aula -1 horas	4	4

			De trabajo Independiente (T.I): · Resolución tareas - 1 hora · Trabajo de investigación – 1 hora · Trabajo grupal: 2 horas		
14	Tipos de herramientas de prueba. Herramientas de prueba. Virtual Herramienta.	Analiza los tipos de herramientas de pruebas-	Lectivas (L): · Introducción al tema - 1 hora · Desarrollo del tema - 2 · Ejercicios en aula -1 horas	4	4
			De trabajo Independiente (T.I): · Resolución tareas - 1 hora · Trabajo de investigación – 1 hora · Trabajo grupal: 2 horas		
15	Uso efectivo de las herramientas: Beneficios potenciales y riesgos. Introduciendo una herramienta en la organización. Cuarta Práctica Calificada Virtual Herramienta.	Aplicación de las herramientas de pruebas.	Lectivas (L): · Introducción al tema - 1 hora · Desarrollo del tema - 2 Ejercicios en aula -1 horas	4	4
			De trabajo Independiente (T.I): · Resolución tareas - 1 hora · Trabajo de investigación – 1 hora Trabajo grupal: 2 horas		
16	Examen final				
17	Entrega de promedios finales y acta del curso.				

V. ESTRATEGIAS METODOLÓGICAS

- Método Expositivo – Interactivo. Disertación docente, exposición del estudiante.
- Método de Discusión Guiada. Conducción del grupo para abordar situaciones y llegar a conclusiones y recomendaciones.
- Método de Demostración – Ejecución. El docente ejecuta para demostrar cómo y con que se hace y el estudiante ejecuta, para demostrar que aprendió.

VI. RECURSOS DIDÁCTICOS

Equipos: computadora, ecran, proyector de multimedia.

Materiales: Separatas, pizarra, plumones.

VII. EVALUACIÓN

El promedio final de la asignatura se obtiene mediante la fórmula siguiente:

$$PF = (2*PE+EP+EF)/4$$

$$PE = ((P1+P2+P3+P4-MN)/3 + W1) /2$$

Donde:

PF= Promedio final

PE= Promedio de evaluaciones

EP= Examen parcial

EF=Examen final

Donde:

W1 = Trabajo.

P1 - P4 = Evaluaciones

MN = Menor nota

VIII. FUENTES DE CONSULTA

Bibliográficas

- Foundations of Software Testing ISTQB Certification (Dorothy Graham, Erik Van Veenendaal, Isabel Evans, Rex Black) 2011. (Reino Unido)
- The Art of Software Testing. Glenflord Myers. 2004 (Canadá)

IX. APOORTE DEL CURSO AL LOGRO DE LOS RESULTADOS DEL ESTUDIANTE

El aporte del curso al logro de los Resultados del Estudiante (Student Outcomes) en la formación del graduado en Ingeniería de Computación y Sistemas, se establece en la tabla siguiente:

K = clave **R** = relacionado **Recuadro vacío** = no aplica

a.	Habilidad para aplicar conocimientos de computación y matemáticas apropiadas para los resultados del estudiante y las disciplinas enseñadas.	
b.	Habilidad para analizar un problema e identificar y definir los requerimientos apropiados para su solución.	K
c.	Habilidad para diseñar, implementar y evaluar un sistema basado en computadoras, procesos, componentes o programa que satisfagan las necesidades requeridas.	
d.	Habilidad para trabajar con efectividad en equipos para lograr una meta común.	R
e.	Comprensión de los aspectos y las responsabilidades profesional, ética, legal, de seguridad y social.	
f.	Habilidad para comunicarse con efectividad con un rango de audiencias.	R
g.	Habilidad para analizar el impacto local y global de la computación en los individuos, organizaciones y la sociedad.	
h.	Reconocer la necesidad y tener la habilidad para comprometerse a un continuo desarrollo profesional.	
i.	Habilidad para usar técnicas, destrezas, y herramientas modernas necesarias para la práctica de la computación.	K
j.	Comprensión de los procesos que soportan la entrega y la administración de los sistemas de información dentro de un entorno específico de aplicación.	K