

Programa descriptivo por unidad de competencia

Programa educativo	Licenciatura en Ingeniería en Desarrollo y Tecnologías de Software	Modalidad		Presencial	
Clave	IS07	H S M		Horas semestrales	Créditos
Unidad de competencia	Calidad en los procesos de desarrollo de software	Teoría	Práctica	64	6
		2	2		
Ubicación	Quinto semestre.	Unidades CONAIC		42.67	
Prerrequisito	Ninguno.	H S M de cómputo		2	
Perfil docente	Contar con título profesional, grado de maestría y preferentemente con grado de doctorado en áreas afines a informática y computación. Demostrar experiencia en docencia en el nivel medio superior o superior mínima de dos años. Dominar los lenguajes de programación actuales, deseable con una certificación en lenguajes de programación.				
Presentación	En esta Unidad de Competencia (UC) el estudiante tiene la capacidad para entender y crear pruebas de software para asegurar la calidad en el desarrollo de productos de software. Se ubica en el área de conocimiento de programación e ingeniería de software, y se relaciona con la UC Taller 3. Atiende al perfil de egreso en el sentido que domine los conocimientos teóricos y prácticos para la implementación de procesos de calidad en el desarrollo de software.				
Propósito	Desarrolla soluciones de software implementando calidad en el proceso de desarrollo de un producto de software como resultado de la unidad de competencia.				
Competencias genéricas					
Maneja tecnologías de la información y comunicación para la gestión y construcción de conocimientos. Se adapta a nuevos contextos de su práctica profesional y de su vida personal en el ámbito local, regional, nacional e internacional. Se mantiene actualizado en los conocimientos y habilidades de manera permanente y los utiliza en su práctica profesional y vida personal.					
Competencias disciplinares					
Posee los conocimientos teóricos y prácticos para la construcción conceptual de soluciones de software.					
Competencias profesionales					
Aplica metodologías y técnicas de análisis y diseño para el desarrollo de software. Desarrolla productos de software aplicando modelos y metodologías para el aseguramiento de la calidad en los procesos de desarrollo.					

Mapa de la unidad de competencia

Unidad de competencia	Subcompetencia	Resultado de aprendizaje
<p>Calidad en los procesos de desarrollo de software</p>	1. Realiza estimaciones de proyectos de software.	1.1. Aprende a estimar tiempos y costos de los proyectos de software.
	2. Solicita requerimientos de software.	2.1. Realizar el levantamiento de requerimientos para su análisis y diseño de los proyectos de software.
	3. Analiza y Diseña sistemas de información con la notación UML 2.0.	3.1. Realiza el análisis y diseño de proyectos de software en base al modelado UML 2.0.
	4. Implementa control de cambios, métricas y control de riesgos.	4.1. Realiza documentación para el control de cambios en los proyectos de software, genera métricas y minimiza riesgos.
	5. Realiza pruebas e implementación de soluciones de software.	5.1. Genera pruebas de aceptación mediante generación de pruebas unitarias e integrales para la implementación de proyectos de software.

Cuadro descriptivo por subcompetencia

Subcompetencia	Realiza estimaciones de proyectos de software.	Número	1
Propósito de la subcompetencia	Aprende a hacer estimaciones de proyectos de software empleando diferentes técnicas para su realización.	Total de horas	12
Resultado de aprendizaje	1.1. Aprende a estimar tiempos y costos de los proyectos de software.	Horas asignadas	12
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> 1. Generación de una idea para un proyecto de software. 2. Investigación de las diferentes Técnicas usadas en la estimación de proyectos de Software. 3. Estimación y generación de calendario del proyecto de software usando las distintas técnicas revisadas. 4. Evaluación del resultado de aprendizaje. 	<ol style="list-style-type: none"> 1. Documento digital con la idea donde se describa en extenso. 2. Documento digital con la investigación de las diferentes técnicas a utilizar. 3. Documento con la estimación generación de calendario del proyecto de software usando las distintas técnicas revisadas. 4. Examen escrito. 	20%	<ol style="list-style-type: none"> 1. Técnicas de estimación de costos. 2. Técnicas de descomposición. 3. Estimación basado en el proceso. 4. COCOMO.

Cuadro descriptivo por subcompetencia

Subcompetencia	Solicita requerimientos de software.	Número	2
Propósito de la subcompetencia	Aprende a levantar requerimientos de proyectos de software identificando usuarios, roles, procesos realizados.	Total de horas	12
Resultado de aprendizaje	2.1. Realiza el levantamiento de requerimientos para su análisis y diseño de los proyectos de software.	Horas asignadas	12
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> Definición de necesidad de negocio, usuarios y roles del caso de estudio. Definición del alcance de la solución propuesta. Definición del alcance de la solución propuesta. Evaluación del resultado de aprendizaje. 	<ol style="list-style-type: none"> Documento digital donde se determine la necesidad de negocio, roles, nombres y responsabilidades de los involucrados. Documento digital con las plantillas que se usaran para documentar los requerimientos establecidos por proveedores validos de requerimientos. Minutas de reuniones de trabajo con el equipo de desarrollo y con el cliente. Documento digital donde se establezca los requerimientos necesarios en base al estándar de la IEEE 830. Examen escrito. 	20%	<ol style="list-style-type: none"> Proceso de preparación de levantamiento de requerimientos. Técnicas de levantamiento de requerimientos. Estándar IEEE 830.

Cuadro descriptivo por subcompetencia

Subcompetencia	Analiza y Diseña sistemas de información con la notación UML 2.0.	Número	3
Propósito de la subcompetencia	Analiza y diseña software usando la notación UML 2.0.	Total de horas	18
Resultado de aprendizaje	3.1. Realiza el análisis y diseño de proyectos de software en base al modelado UML 2.0.	Horas asignadas	18
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> Investigación de los 13 diagramas de UML 2.0. Generación de diagramas UML en base al caso de estudio. Evaluación del resultado de aprendizaje. 	<ol style="list-style-type: none"> Documento digital con las especificaciones, características y definiciones de cada diagrama de UML 2.0. Documento digital del análisis del caso de estudio usando software especializado para la creación de los diagramas. Examen escrito. 	20%	<ol style="list-style-type: none"> Diagramas de CU. Diagramas de Actividades. Diagramas de Secuencia. Diagramas de Estado. Diagramas de Clase. Diagramas de Objeto. Diagramas de Componentes. Diagramas de Paquetes. Diagramas de Estructuras compuestas. Diagramas de Comunicaciones. Diagramas de Despliegue. Diagramas de Tiempo. Diagramas Global de Interacciones.

Cuadro descriptivo por subcompetencia

Subcompetencia	Implementa control de cambios, métricas y control de riesgos.	Número	4
Propósito de la subcompetencia	Aprende a minimizar riesgos, generar control de cambios y administrarlos.	Total de horas	10
Resultado de aprendizaje	4.1 Realiza documentación para el control de cambios en los proyectos de software, genera métricas y minimiza riesgos.	Horas asignadas	10
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> Descripción de hojas de control de cambios. Definición de métricas del caso de estudio. Definición del control de riesgos de un proyecto de software. Evaluación del resultado de aprendizaje. 	<ol style="list-style-type: none"> Documento con la descripción de control de cambios para el caso de estudio. Documento para la validación de métricas del proyecto de software. Documento de control de riesgos del proyecto de software. Examen escrito. 	20%	<ol style="list-style-type: none"> Control de cambios. Métricas de software. Control de riesgos.

Cuadro descriptivo por subcompetencia

Subcompetencia	Realiza pruebas e implementación de soluciones de software.	Número	5
Propósito de la subcompetencia	Crea pruebas de software unitarias e integrales para implementar de manera exitosa el proyecto de software.	Total de horas	12
Resultado de aprendizaje	5.1 Genera pruebas de aceptación mediante generación de pruebas unitarias e integrales para la implementación de proyectos de software.	Horas asignadas	12
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> Definición de pruebas de aceptación. Validación de pruebas unitarias e integrales. Carta de aceptación de entrega de software. Evaluación del resultado de aprendizaje. 	<ol style="list-style-type: none"> Documento digital de la definición de pruebas de aceptación. Documento digital del estándar IEEE 829 del caso de estudio. Documento digital de la carta de entrega-aceptación de proyecto de software. Examen escrito. 	20%	<ol style="list-style-type: none"> Definición de pruebas de aceptación. Definición de pruebas unitarias y de integración. Estándar IEEE 829. Implementación de proyecto.

Actitudes y valores	Responsabilidad. Actitud emprendedora. Cultura de trabajo. Espíritu de superación personal.	
Recursos, materiales y equipo didáctico		
	Recursos didácticos	Equipo de apoyo didáctico
	Apuntes. Diapositivas. Antologías. Manuales.	Proyector. Laboratorio de cómputo. Computadoras. Software especializado.
Fuentes de información		
Bibliografía básica: Barranco, Jesús (2002). <i>Metodología del análisis estructurado de sistemas</i> (2a. ed.). España: Universidad Pontificia de Comillas de Madrid. Witten (2008). <i>Análisis de sistemas, diseño y métodos</i> (7a. ed.). México: Mc Graw Hill. Sommerville, Ian (2005). <i>Ingeniería de software</i> (7a. ed.). España: Pearson Educación.		
Bibliografía complementaria: Aldama De Vega (2010). <i>Administración por calidad</i> . México: Alfaomega.		
Recursos digitales: Ninguno.		