

Curso: Fundamentos de Ingeniería de Software		Horas aula: 2
Clave: 061CP021		Horas virtuales: 2
Antecedentes:		Horas laboratorio: 0 Horas independientes: 1
Competencia del área: Desarrollar software y servicios de soporte técnico y redes, con la finalidad de solucionar problemas y agilizar procesos en la toma de decisiones en empresas públicas y privadas, bajo estándares de calidad nacional e internacional, a través del análisis de problemas, comunicación, liderazgo e innovación.	Competencia del curso: Analizar las metodologías ágiles de software para documentar de forma responsable el desarrollo de sistemas de información con un enfoque de calidad dentro de las organizaciones mediante la implementación de las técnicas y herramientas de la práctica de la ingeniería de software.	
Elementos de competencia:		
<ol style="list-style-type: none"> 1. Conocer los conceptos fundamentales de la terminología de la ingeniería de software para aplicarlos de forma responsable en la documentación del desarrollo de sistemas de información utilizando fuentes de información confiables. 2. Aplicar con un enfoque de calidad las metodologías de desarrollo ágil para documentar casos de estudio de una organización con base en los conceptos y lineamientos del desarrollo ágil. 3. Analizar las técnicas y herramientas de la práctica de la ingeniería de software con la finalidad de aplicarlas responsablemente en la documentación de casos de estudio de una organización con base en los lineamientos del desarrollo ágil. 		
Perfil del docente:		
Licenciatura en las áreas de Ingeniería de Software o Sistemas Computacionales preferentemente con posgrado. El docente debe tener amplios conocimientos sobre gestión de proyectos informáticos, normas, estándares, métricas, herramientas y documentación de la ingeniería de software. Contar experiencia docente de dos años, con habilidades para desarrollar la función de docencia y mostrar interés por los temas relacionados con su especialidad y tener espíritu investigador para garantizar su competencia y la de sus estudiantes. Evalúa los procesos de enseñanza y de aprendizaje con un enfoque por competencias, con una actitud de cambio a las innovaciones pedagógicas. Construye ambientes para el aprendizaje autónomo y colaborativo con apoyo de la tecnología.		
Elaboró: ALMA ISABEL ARIAS HURTADO; ISIDRO GARCÍA RANGEL		Octubre 2021
Revisó: DRA. CECILIA LÓPEZ CAMACHO		Octubre 2021
Última actualización:		
Autorizó: Coordinación de Procesos Educativos		

Elemento de competencia 1: Conocer los conceptos fundamentales de la terminología de la ingeniería de software para aplicarlos de forma responsable en la documentación del desarrollo de sistemas de información utilizando fuentes de información confiables.

Competencias blandas a promover: Responsabilidad

EC1 Fase I: Conceptos básicos y evolución del software

Contenido: Conceptos y mitos de software, ingeniería de software, aplicaciones del software, desarrollo de software.

EC1 F1 Actividad de aprendizaje 1: Investigación de conceptos sobre software

Realizar responsablemente de forma individual, una investigación de conceptos sobre la definición, características y mitos del software (mitos de la administración, mitos del cliente y mitos del profesional), hacer uso de los documentos establecidos en los recursos o fuentes confiables de internet.

Participar en clase presencial realizando aportaciones sobre cada uno de los aspectos investigados en la dinámica establecida por el docente para su retroalimentación.

2 hrs. Aula
1 hr. Virtual
1 hr. Independiente

Tipo de actividad:

Aula (X) Virtuales (X) Laboratorio ()
Grupal () Individual (X) Equipo ()
Independientes (X)

Recursos:

- Alonso Amo, F. Martínez Normand L. y Segovia Pérez, J. (2005). [Introducción a la Ingeniería del Software: modelos de desarrollo de programas](#).
- Aparicio, A. (2012). [Ingeniería de software](#).
- Dutoit, A. H. (2002). [Ingeniería de software orientado a objetos](#).
- Pressman, R. (2010). [Ingeniería de software](#).
- Sommerville, I. (2011). [Ingeniería de software](#).

Criterios de evaluación de la actividad:

Rúbrica de [Investigación de conceptos](#).

EC1 F1 Actividad de aprendizaje 2: Mapa conceptual Ingeniería de software

Elaborar de forma individual, un mapa conceptual en el cual se incluya el concepto, antecedentes e importancia de la ingeniería de software, hacer uso de los documentos establecidos en los recursos o fuentes confiables de internet.

Participar grupalmente en una lluvia de ideas sobre el concepto de ingeniería de software y su importancia, complementar la información del mapa conceptual con la información que proporcionen en clase.

2 hrs. Aula
2 hrs. Virtuales
1 hr. Independiente

Tipo de actividad:

Aula (X) Virtuales (X) Laboratorio ()
Grupal (X) Individual (X) Equipo ()
Independientes (X)

Recursos:

- Alonso Amo, F. Martínez Normand, L. y Segovia Pérez, J. (2005). [Introducción a la Ingeniería del Software: modelos de desarrollo de programas](#).
- Aparicio, A. (2012). [Ingeniería de software](#).
- Dutoit, A. H. (2002). [Ingeniería de software orientado a objetos](#).
- Pressman, R. (2010). [Ingeniería de software](#).
- Sommerville, I. (2011). [Ingeniería de software](#).

Criterios de evaluación de la actividad:

Rúbrica de [Mapa conceptual](#).

EC1 F1 Actividad de aprendizaje 3: Resumen de desarrollo de software profesional

Elaborar con un enfoque de calidad de manera individual, un resumen sobre el tema de desarrollo de software profesional del capítulo 1 del libro de Sommerville, ubicado en el apartado de recursos.

Tipo de actividad:

Aula () Virtuales (X) Laboratorio ()
Grupal () Individual (X) Equipo ()
Independientes (X)

Recursos:

<p>Integrar antecedentes, tipos de productos de software, preguntas planteadas con frecuencia sobre el software y atributos esenciales del buen software.</p> <p>2 hrs. Virtuales 1 hr. Independiente</p>	<p>Sommerville, I. (2011). Ingeniería de software.</p> <p>Criterios de evaluación de la actividad: Rúbrica de Resumen.</p>
<p>EC1 F1 Actividad de aprendizaje 4: Esquema gráfico de las aplicaciones de software.</p> <p>Realizar en equipo un esquema gráfico sobre los siguientes temas:</p> <ul style="list-style-type: none"> • Diversidad o dominios de aplicación de software • La ingeniería de software y la Web (webapps) <p>Hacer uso de los documentos establecidos en los recursos o fuentes confiables de internet y presentar esquema gráfico en clase y aclarar dudas con ayuda del facilitador.</p> <p>2 hrs. Aula 1 hr. Virtual 1 hr. Independiente</p>	<p>Tipo de actividad: Aula (X) Virtuales (X) Laboratorio () Grupal (X) Individual () Equipo (X) Independientes (X)</p> <p>Recursos:</p> <ul style="list-style-type: none"> • Pressman, R. (2010). Ingeniería de software. • Sommerville, I. (2011). Ingeniería de software. <p>Criterios de evaluación de la actividad: Rúbrica de Esquema Gráfico.</p>
<p>EC1 F1 Actividad de aprendizaje 5: Wiki Modelos de procesos de software</p> <p>Participar de forma individual en la creación de un Wiki sobre los modelos de proceso de software, describir las características de un modelo de proceso en particular y la fuente de donde se obtuvo la información en formato APA.</p> <p>2 hrs. Virtuales 1 hr. Independiente</p>	<p>Tipo de actividad: Aula () Virtuales (X) Laboratorio () Grupal () Individual (X) Equipo () Independientes (X)</p> <p>Recursos:</p> <ul style="list-style-type: none"> • Aparicio, A. (2012). Ingeniería de software. • Pressman, R. (2010). Ingeniería de software. • Sommerville, I. (2011). Ingeniería de software. <p>Criterios de evaluación de la actividad: Rúbrica de Participación en un Wiki.</p>
<p>EC1 Fase II: Herramientas CASE</p> <p>Contenido: Concepto, importancia y clasificación.</p>	
<p>EC1 F2 Actividad de aprendizaje 6: Cuadro sinóptico Herramientas CASE</p> <p>Elaborar de manera individual, un cuadro sinóptico que contenga definición, importancia y estructura de una herramienta CASE, hacer uso de los documentos establecidos en los recursos o fuentes confiables de internet.</p>	<p>Tipo de actividad: Aula (X) Virtuales (X) Laboratorio () Grupal () Individual (X) Equipo () Independientes (X)</p> <p>Recursos:</p> <ul style="list-style-type: none"> • Chavarría-Báez, L., & Rojas, N. O. (2016). Sobre el uso de herramientas CASE para la enseñanza de

<p>Reforzar la información presentada en la actividad, con la explicación que proporcione el facilitador en el aula del tema.</p> <p>1 hr. Aula 2 hrs. Virtuales 1 hr. Independiente</p>	<p>bases de datos .</p> <ul style="list-style-type: none"> • Chino Mamani, M. A. (2018). Herramientas CASE para el desarrollo de diagramas de flujo . • EcuRed. (s.f.). Herramienta CASE . <p>Criterios de evaluación de la actividad:</p> <p>Rúbrica de cuadro sinóptico .</p>
<p>EC1 F2 Actividad de aprendizaje 7: Exposición clasificación de Herramientas CASE</p> <p>Presentar en equipo, una exposición oral sobre la clasificación de las herramientas CASE, hacer uso como mínimo de la siguiente clasificación: Upper CASE (U-CASE), Middle CASE (M-CASE) y Lower CASE (L-CASE), incluir ejemplos de herramientas CASE especificando su nombre y para que se utiliza.</p> <p>Utilizar responsablemente los documentos establecidos en los recursos o fuentes confiables de internet, exponer frente al grupo de la información encontrada y subir a plataforma.</p> <p>2 hrs. Aula 2 hrs. Virtuales 1 hr. Independiente</p>	<p>Tipo de actividad: Aula (X) Virtuales (X) Laboratorio () Grupal (X) Individual () Equipo (X) Independientes (X)</p> <p>Recursos:</p> <ul style="list-style-type: none"> • Chavarría-Báez, L., & Rojas, N. O. (2016). Sobre el uso de herramientas CASE para la enseñanza de bases de datos . • Chino Mamani, M. A. (2018). Herramientas CASE para el desarrollo de diagramas de flujo . • EcuRed. (s.f.). Herramienta CASE . <p>Criterios de evaluación de la actividad:</p> <p>Rúbrica de Exposición oral .</p>
<p>EC1 F2 Actividad de aprendizaje 8: Evaluación primer elemento</p> <p>Realizar de manera individual en el aula, la evaluación de los temas vistos en el primer elemento de competencia, con base en las indicaciones proporcionadas por el docente.</p> <p>1 hr. Aula</p>	<p>Tipo de actividad: Aula (X) Virtuales () Laboratorio () Grupal () Individual (X) Equipo () Independientes ()</p> <p>Recursos:</p> <p>Evaluación.</p> <p>Criterios de evaluación de la actividad: Cantidad de respuestas correctas con respecto al total de preguntas.</p>
<p>Evaluación formativa:</p> <ul style="list-style-type: none"> • Investigación sobre software. • Mapa conceptual sobre ingeniería de software. • Resumen de desarrollo de software profesional. • Esquema gráfico de las aplicaciones de Software. • Wiki Modelos de procesos de software. • Cuadro sinóptico sobre herramientas CASE. 	

- Exposición, clasificación de Herramientas CASE.
- Evaluación primer elemento.

Fuentes de información

1. Alonso Amo, F. Martínez Normand, L. y Segovia Pérez, J. (2005). Introducción a la Ingeniería del Software: modelos de desarrollo de programas. Delta Publicaciones. <https://elibro.net/es/lc/ues/titulos/170188>
2. Aparicio, A. (2012). Ingeniería de software. Universidad Nacional Abierta y a Distancia. http://www.usfx.bo/nueva/vicerrectorado/citas/TECNOLOGICAS_20/Ingenieria%20Sistemas/37.pdf
3. Chavarría-Báez, L., & Rojas, N. O. (2016). Sobre el uso de herramientas CASE para la enseñanza de bases de datos. *Sistemas, Cibernética e Informática*, 13(2), 51-56. <http://www.iiisci.org/journal/pdv/risci/pdfs/CB321MR16.pdf>
4. Chino Mamani, M. A. (2018). Herramientas CASE para el desarrollo de diagramas de flujo. Repositorio institucional de la Universidad Mayor de San Andrés. <https://repositorio.umsa.bo/handle/123456789/17448>
5. Dutoit, A. H. (2002). Ingeniería de software orientado a objetos. Pearson Educación. <https://elibro.net/es/lc/ues/titulos/74078>
6. EcuRed. Herramienta CASE. https://www.ecured.cu/Herramienta_CASE
7. Pressman, R. (2010). Ingeniería de software. Séptima edición. México. McGraw-Hill. <http://cotana.informatica.edu.bo/downloads/Id-Ingenieria.de.software.enfoque.practico.7ed.Pressman.PDF>
8. Sommerville, I. (2011). Ingeniería de software (9a. ed.). Pearson Educación. <https://elibro.net/es/ereader/ues/37857?page=11>

Elemento de competencia 2: Aplicar con un enfoque de calidad las metodologías de desarrollo ágil para documentar casos de estudio de una organización con base en los conceptos y lineamientos del desarrollo ágil.

Competencias blandas a promover: Enfoque de calidad

EC2 Fase I: Conceptos básicos del desarrollo ágil

Contenido: Definición, principios y ventajas del desarrollo ágil.

EC2 F1 Actividad de aprendizaje 9: Apuntes de clase conceptos básicos del desarrollo ágil

Realizar de manera individual, apuntes de clase sobre el desarrollo ágil, con base en la información proporcionada en el aula sobre los temas ¿qué es la agilidad?, definición del proceso ágil, los principios de agilidad y diferencias entre modelos tradicionales y el desarrollo ágil.

Hacer uso de los recursos establecidos en la actividad.

3 hrs. Aula
1 hr. Independiente

Tipo de actividad:

Aula (X) Virtuales () Laboratorio ()
Grupal () Individual (X) Equipo ()
Independientes (X)

Recursos:

- Agile Alliance. (s.f.). [Extreme Programming \(XP\)](#).
- Orjuela Duarte, A., & Rojas C., M. (2008). [Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo](#).
- Pressman, R. (2010). [Ingeniería de software](#).
- Sommerville, I. (2011). [Ingeniería de software](#).

Criterios de evaluación de la actividad:

Rúbrica de [Apuntes de clase](#).

EC2 F1 Actividad de aprendizaje 10: Foro desarrollo ágil

Participar de forma individual en el Foro sobre el desarrollo ágil, publicando una aportación con un enfoque de calidad que pueda resolver la problemática planteada por Sommerville (2011, pág. 79), misma que se describe a continuación:

Con la finalidad de reducir costos y el impacto ambiental, su compañía decide cerrar algunas oficinas y ofrecer apoyo al personal para trabajar desde casa. Sin embargo, el gerente que introdujo la política no está consciente de que el software se desarrolló usando métodos ágiles, que se apoya en el trabajo cercano del equipo y de la programación de pares. Analice las dificultades que causaría esta nueva política y cómo podría solventar estos problemas.

2 hrs. Virtuales
1 hr. Independiente

Tipo de actividad:

Aula () Virtuales (X) Laboratorio ()
Grupal () Individual (X) Equipo ()
Independientes (X)

Recursos:

Sommerville, I. (2011). [Ingeniería de software](#).

Criterios de evaluación de la actividad:

Rúbrica de [Participación en Foro](#).

EC2 Fase II: Metodologías de desarrollo ágil

Contenido: Definición y características de XP, Scrum y otras metodologías de desarrollo ágil.

<p>EC2 F2 Actividad de aprendizaje 11: Video Programación extrema (XP)</p> <p>Producir en equipo, un video en el cual describa la metodología de desarrollo ágil Programación Extrema (XP), explicar al menos los siguientes aspectos de la metodología: antecedentes y definición, valores, proceso, roles, ventajas y desventajas.</p> <p>Hacer uso de los recursos establecidos en la actividad, fuentes confiables de internet para el desarrollo del video y participar en clase en la retroalimentación del tema de forma grupal.</p> <p>2 hrs. Aula 2 hrs. Virtuales 1 hr. Independiente</p>	<p>Tipo de actividad: Aula (X) Virtuales (X) Laboratorio () Grupal (X) Individual () Equipo (X) Independientes (X)</p> <p>Recursos:</p> <ul style="list-style-type: none"> • Agile Alliance. (s.f.). Extreme Programming (XP). • Orjuela Duarte, A., & Rojas C., M. (2008). Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo. • Pressman, R. (2010). Ingeniería de software. • RedHat. (2021). ¿Qué es la metodología ágil?. • Sommerville, I. (2011). Ingeniería de software. <p>Criterios de evaluación de la actividad: Rúbrica de Elaboración de video.</p>
<p>EC2 F2 Actividad de aprendizaje 12: Presentación Oral SCRUM</p> <p>Realizar en equipo, una presentación oral sobre la metodología SCRUM, la cual debe de contener como mínimo los siguientes aspectos: antecedentes y definición, valores, equipos, eventos y artefactos.</p> <p>Hacer uso de los recursos establecidos en la actividad, exponer en clase y no exceder los 10 minutos de duración.</p> <p>3 hrs. Aula 2 hrs. Virtuales 1 hr. Independiente</p>	<p>Tipo de actividad: Aula (X) Virtuales (X) Laboratorio () Grupal () Individual () Equipo (X) Independientes (X)</p> <p>Recursos:</p> <ul style="list-style-type: none"> • Agile Alliance. (s.f.). Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo. • Las Heras del Dedo, R. D. y Álvarez García, A. (2017). Métodos ágiles: Scrum, Kanban, Lean. • Pressman, R. (2010). Ingeniería de software. • Schwaber, K. y Sutherland, J. (2017). La Guía Definitiva de Scrum: Las Reglas del Juego. • Sommerville, I. (2011). Ingeniería de software. <p>Criterios de evaluación de la actividad: Rúbrica de Presentación Oral.</p>
<p>EC2 F2 Actividad de aprendizaje 13: Blog metodologías de desarrollo ágil</p> <p>Crear en equipo, un blog de forma grupal sobre las metodologías de desarrollo ágil, partiendo de investigar sobre una metodología de desarrollo ágil incluyendo la XP y la SCRUM.</p> <p>Integrar todas las metodologías ágiles que encuentren y presentar el blog ejerciendo un enfoque de la calidad, en la fecha acordada.</p> <p>2 hrs. Aula 3 hrs. Virtuales</p>	<p>Tipo de actividad: Aula (X) Virtuales (X) Laboratorio () Grupal (X) Individual () Equipo (X) Independientes (X)</p> <p>Recursos:</p> <ul style="list-style-type: none"> • Agile Alliance. (s.f.). Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo. • Las Heras del Dedo, R. D. y Álvarez García, A. (2017). Métodos ágiles: Scrum, Kanban, Lean. • Pressman, R. (2010). Ingeniería de software. <p>Criterios de evaluación de la actividad:</p>

1 hr. Independiente	Rúbrica de Blog .
<p>EC2 F2 Actividad de aprendizaje 14: Caso de estudio metodología de desarrollo ágil</p> <p>Resolver en equipo, el caso de estudio presentado en el aula en el cual se ponga en práctica la documentación del desarrollo de un software por medio de una metodología de desarrollo ágil, con base en la solución de la metodología XP o Scrum según lo establezca el facilitador.</p> <p>Utilizar los recursos establecidos en la actividad y fuentes confiables de internet.</p> <p>3 hrs. Aula 2 hrs. Virtuales 1 hr. Independiente</p>	<p>Tipo de actividad: Aula (X) Virtuales (X) Laboratorio () Grupal () Individual () Equipo (X) Independientes (X)</p> <p>Recursos:</p> <ul style="list-style-type: none"> • Agile Alliance. (s.f.). Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo. • Las Heras del Dedo, R. D. y Álvarez García, A. (2017). Métodos ágiles: Scrum, Kanban, Lean. • Pressman, R. (2010). Ingeniería de software. • Schwaber, K. y Sutherland, J. (2017). La Guía Definitiva de Scrum: Las Reglas del Juego. • Sommerville, I. (2011). Ingeniería de software. <p>Criterios de evaluación de la actividad:</p> <p>Rúbrica de Análisis de Casos.</p>

<p>Evaluación formativa:</p> <ul style="list-style-type: none"> • Apuntes de clase conceptos básicos del desarrollo ágil. • Foro desarrollo ágil. • Video programación extrema (XP). • Presentación oral SCRUM. • Blog metodologías de desarrollo ágil. • Caso de estudio metodología de desarrollo ágil.
--

Fuentes de información

<ol style="list-style-type: none"> 1. Agile Alliance. (s.f.). Extreme Programming (XP). https://www.agilealliance.org/glossary/xp 2. Las Heras del Dedo, R. D. y Álvarez García, A. (2017). Métodos ágiles: Scrum, Kanban, Lean. Difusora Larousse - Anaya Multimedia. https://elibro.net/es/lc/ues/titulos/122933 3. Monte Galiano, J. (2016). Implantar scrum con éxito. Editorial UOC. https://elibro.net/es/lc/ues/titulos/58575 4. Orjuela Duarte, A., & Rojas C., M. (2008). Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo. Revista Avances en Sistemas e Informática. 5(2), 159-171. https://www.redalyc.org/articulo.oa?id=133115027022 5. Pressman, R. (2010). Ingeniería de software. Séptima edición. México. McGraw-Hill. http://cotana.informatica.edu.bo/downloads/Id-Ingenieria.de.software.enfoque.practico.7ed.Pressman.PDF 6. Redhat. (2021). ¿Qué es la metodología ágil?. https://www.redhat.com/es/devops/what-is-agile-methodology 7. Schwaber, K. y Sutherland, J. (2017). La Guía Definitiva de Scrum: Las Reglas del
--

Juego. <https://scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Spanish-SouthAmerican.pdf>

8. Sommerville, I. (2011). Ingeniería de software (9a. ed.). Pearson

Educación. <https://elibro.net/es/ereader/ues/37857?page=11>

Elemento de competencia 3: Analizar las técnicas y herramientas de la práctica de la ingeniería de software con la finalidad de aplicarlas responsablemente en la documentación de casos de estudio de una organización con base en los lineamientos del desarrollo ágil.

Competencias blandas a promover: Responsabilidad

EC3 Fase I: Prácticas de la ingeniería de software

Contenido: La esencia de la práctica y principios que guían la práctica.

EC3 F1 Actividad de aprendizaje 15: Reporte escrito de la esencia de la práctica

Elaborar de manera individual, un reporte escrito del tema la esencia de la práctica el cual se encuentra en el capítulo I del libro de Pressman, incluir responsablemente cada una de las preguntas de las etapas de la solución de problemas y sus 7 principios generales.

2 hrs. Virtuales

Tipo de actividad:

Aula () Virtuales (X) Laboratorio ()
Grupal () Individual (X) Equipo ()
Independientes ()

Recursos:

- Cartuche Jara, A. (2017). [Ingeniería de Software I. Principios, conceptos, métodos y herramientas](#).
- Pressman, R. (2010). [Ingeniería de software](#).

Criterios de evaluación de la actividad:

Rúbrica de [Reporte escrito](#).

EC3 F1 Actividad de aprendizaje 16: Exposición principios que guían la práctica

Realizar una exposición en equipo sobre el tema: los principios que guían la práctica de la Ingeniería en software. Hacer uso de los siguientes temas:

- Práctica de la Comunicación: introducción, principios y documentos a desarrollar.
- Práctica de la Planeación: introducción, principios y documentos a desarrollar.
- Práctica del Modelado: introducción, principios del modelado del análisis, principios del modelado del diseño y documentos a desarrollar.
- Práctica de la Construcción: introducción, principios de la construcción del software, principios de pruebas de software y documentos a desarrollar.
- Práctica del Despliegue: introducción, principios, actividades de la práctica de despliegue y documentos a desarrollar.

Hacer uso del capítulo 4 del libro de Pressman y fuentes confiables de internet.

3 hrs. Aula
1 hr. Virtual
1 hr. Independiente

Tipo de actividad:

Aula (X) Virtuales (X) Laboratorio ()
Grupal () Individual () Equipo (X)
Independientes (X)

Recursos:

- Cartuche Jara, A. (2017). [Ingeniería de Software I. Principios, conceptos, métodos y herramientas](#).
- Pressman, R. (2010). [Ingeniería de software](#).

Criterios de evaluación de la actividad:

Rúbrica de [Exposición](#).

EC3 Fase II: Técnicas de estimación

Contenido: COCOMO y métricas del punto de función

EC3 F2 Actividad de aprendizaje 17: Cuadro comparativo de las técnicas de estimación

Elaborar de manera individual, un cuadro comparativo en la cual se detalle nombre, descripción y ejemplos de las técnicas de estimación de proyectos de software y presentar responsablemente en clase el resultado para su retroalimentación.

2 hrs. Aula
2 hrs. Virtuales
1 hr. Independiente

Tipo de actividad:

Aula (X) Virtuales (X) Laboratorio ()
Grupal () Individual (X) Equipo ()
Independientes (X)

Recursos:

- Cartuche Jara, A. (2017). [Ingeniería de Software I. Principios, conceptos, métodos y herramientas](#) .
- Pressman, R. (2010). [Ingeniería de software](#) .

Criterios de evaluación de la actividad:

Rúbrica de [Cuadro Comparativo](#) .

EC3 F2 Actividad de aprendizaje 18: Caso de estudio prácticas y técnicas de estimación de la Ingeniería de Software

Resolver de manera individual, el caso de estudio desarrollado en la actividad 14, lo que corresponde a las prácticas de la ingeniería de software y las técnicas de estimación, con base en las indicaciones que proporcione el docente.

Presentar en clase una vez concluido la propuesta de solución de caso de estudio.

2 hrs. Aula
2 hrs. Virtuales

Tipo de actividad:

Aula (X) Virtuales (X) Laboratorio ()
Grupal () Individual () Equipo (X)
Independientes ()

Recursos:

- Pressman, R. (2010). [Ingeniería de software](#) .
- Sommerville, I. (2011). [Ingeniería de software](#) .
- Cartuche Jara, A. (2017). [Ingeniería de Software I. Principios, conceptos, métodos y herramientas](#) .

Criterios de evaluación de la actividad:

Rúbrica de [Análisis de Casos](#) .

Evaluación formativa:

- Reporte escrito de la esencia de la práctica.
- Exposición principios que guían la práctica.
- Cuadro comparativo de las técnicas de estimación.
- Caso de estudio prácticas y técnicas de estimación de la ingeniería de software.

Fuentes de información

1. Alonso Amo, F. Martínez Normand, L. y Segovia Pérez, J. (2005). Introducción a la Ingeniería del Software: modelos de desarrollo de programas. Delta Publicaciones. <https://elibro.net/es/lc/ues/titulos/170188>
2. Aparicio, A. (2012). Ingeniería de software. Universidad Nacional Abierta y a Distancia. http://www.usfx.bo/nueva/vicerrectorado/citas/TECNOLOGICAS_20/Ingenieria%20Sistemas/37.pdf
3. Cartuche Jara, A. (2017). Ingeniería de Software I. Principios, conceptos, métodos y herramientas. Instituto Superior Tecnológico Particular. <https://instbolivarmadero.org/onewebmedia/INGENIERIA%20DE%20SOFTWARE%20I%20RESUMEN.p>

[df](#)

4. Dutoit, A. H. (2002). Ingeniería de software orientado a objetos. Pearson Educación. <https://elibro.net/es/lc/ues/titulos/74078>
5. Pressman, R. (2010). Ingeniería de software. Séptima edición. México. McGraw-Hill. <http://cotana.informatica.edu/downloads/Id-Ingenieria.de.software.enfoque.practico.7ed.Pressman.PDF>
6. Sommerville, I. (2011). Ingeniería de software (9a. ed.). Pearson Educación. <https://elibro.net/es/ereader/ues/37857?page=11>

Políticas

- Obligatorio contar con el 85% de asistencia para acreditar el curso.
- La entrega de actividades será por medio de plataforma y los medios adicionales establecidos por el docente.
- Cumplir con las fechas establecidas para la entrega de actividades.
- Mostrar una actitud de respeto para sus compañeros y docente.
- Mantener los teléfonos celulares en vibrador o silencio.

Metodología

- Es responsabilidad del estudiante gestionar los procedimientos necesarios para alcanzar el desarrollo de las competencias del curso.
- El curso se desarrollará combinando sesiones presenciales y virtuales, así como prácticas presenciales en laboratorios, campos o a distancia en congruencia con la naturaleza de la asignatura.
- Los productos académicos escritos deberán ser entregados en formato PDF en la plataforma institucional, de acuerdo con los criterios establecidos por el facilitador.

Evaluación

La evaluación del curso se realizará de acuerdo al Reglamento Escolar vigente que considera los siguientes artículos:

ARTÍCULO 27. La evaluación es el proceso que permite valorar el desarrollo de las competencias establecidas en las secuencias didácticas del plan de estudio del programa educativo correspondiente. Su metodología es integral y considera diversos tipos de evidencias de conocimiento, desempeño y producto por parte del alumno.

ARTÍCULO 28. Las modalidades de evaluación en la Universidad son: Diagnóstica permanente, entendiéndola esta como la evaluación continua del estudiante durante la realización de una o varias actividades;

1. Formativa, siendo esta, la evaluación al alumno durante el desarrollo de cada elemento de competencia; y

2. Sumativa, es la evaluación general de todas y cada una de las actividades y evidencias de las secuencias didácticas.

3. Sólo los resultados de la evaluación sumativa tienen efectos de acreditación y serán reportados al departamento de registro y control escolar.

ARTÍCULO 29. La evaluación sumativa será realizada

tomando en consideración de manera conjunta y razonada, las evidencias del desarrollo de las competencias y los aspectos relacionados con las actitudes y valores logrados por el alumno.

ARTÍCULO 30. Los resultados de la evaluación expresarán el grado de dominio de las competencias, por lo que la escala de evaluación contemplará los niveles de:

1. Competente sobresaliente;
2. Competente avanzado;
3. Competente intermedio;
4. Competente básico; y
5. No aprobado.

El nivel mínimo para acreditar una asignatura será el de competente básico. Para fines de acreditación los niveles tendrán un equivalente numérico conforme a lo siguiente:

Competente sobresaliente=10

Competente avanzado=9

Competente intermedio=8

Competente básico=7

No aprobado=6