

Universidad Autónoma del Estado de México
Unidad Académica Profesional Tianguistenco
Licenciatura en Ingeniería en software

Guía pedagógica:

**HERRAMIENTAS PARA LA ADMINISTRACION
Y PROGRAMACION DE SISTEMAS**

Elaboró: L.I. LEONOR GONZALEZ MUÑOZ Fecha: 31/agosto/2015
L.S.C.A. CARLOS ALBERTO GARCÍA ACEVEDO

Fecha de
aprobación

H. Consejo académico

H. Consejo de Gobierno



Índice

	Pág.
I. Datos de identificación	3
II. Presentación de la guía pedagógica	4
III. Ubicación de la unidad de aprendizaje en el mapa curricular	5
IV. Objetivos de la formación profesional	5
V. Objetivos de la unidad de aprendizaje	6
VI. Contenidos de la unidad de aprendizaje, y su organización	6
VII. Acervo bibliográfico	13
VIII. Mapa curricular	14



I. Datos de identificación

Espacio educativo donde se imparte	Unidad Académica Profesional Tianguistenco								
Licenciatura	Licenciatura en Ingeniería en Software								
Unidad de aprendizaje	Herramientas para la administración y programación de sistemas			Clave	L40815				
Carga académica	1	2	3	4					
	Horas teóricas	Horas prácticas	Total de horas	Créditos					
Período escolar en que se ubica	1	2	3	4	5	6	7	8	9
Seriación	Sistemas Operativos			Ninguna					
	UA Antecedente			UA Consecuente					

Tipo de Unidad de Aprendizaje

Curso	☐	Curso taller	☐
Seminario	☐	Taller	☒
Laboratorio	☐	Práctica profesional	☐
Otro tipo (especificar)			

Modalidad educativa

Escolarizada. Sistema rígido	☐	No escolarizada. Sistema virtual	☐
Escolarizada. Sistema flexible	☒	No escolarizada. Sistema a distancia	☐
No escolarizada. Sistema abierto	☐	Mixta (especificar)	

Formación común

☐	☐
☐	☐
☐	☐

Formación equivalente

Unidad de Aprendizaje



II. Presentación de la guía pedagógica

La guía pedagógica tiene dentro de sus funciones plasmar como se realiza de manera práctica los principales objetivos de esta unidad de aprendizaje que a su vez pretende conocer e identificar las diferentes herramientas que apoyan la administración y la programación de software así como establecer criterios para aplicar las técnicas más adecuadas a cada caso de construcción, mejora o actualización, fundamentándose en las tecnologías existentes.

Analizando la necesidad que tiene la industria de aprovechar tanto las aportaciones de la Ingeniería de software, como el conocer las principales disciplinas de administración, programación y diseño, en virtud de los tan diversos campos de aplicación y de los cambiantes requerimientos que definen el software. Aportando así profesionistas capaces en el estudio e identificación de las diferentes técnicas de evaluación de sistemas críticos, que permiten hacer software bajo las condiciones que prevalecen en las organizaciones hoy.

Para ello se trabaja en conjunto evaluando que el alumno sea capaz de aplicar, distinguir y proponer adecuadamente las técnicas para definir los requisitos enfocados a solucionar problemas prácticos así como tener la habilidad de emplearlos para crear, mejorar o actualizar software con una actitud de valoración de la contribución de las diversas disciplinas de desarrollo de programas que buscan simplificar el trabajo humano.

La unidad de aprendizaje define fundamentalmente la formación práctica o aplicativa de la profesión y el desarrollo de competencias específicas. También provee al alumno de escenarios educativos para la integración, aplicación y desarrollo de los conocimientos, habilidades y actitudes que le permitan el desempeño de las funciones, tareas y resultados requeridos en los diferentes ámbitos de intervención profesional.

El estudiante amplía su visión al percibir cómo la industria está actualmente empleando diversas técnicas para desarrollar software en un entorno tan cambiante y exigente. Al mismo tiempo valora y aplica los conocimientos y habilidades adquiridos en su carrera haciéndose más consciente del relevante papel que desempeñan hoy todas las disciplinas de desarrollo así como la misma Ingeniería de software.



III. Ubicación de la unidad de aprendizaje en el mapa curricular

Núcleo de formación:	Integral
Área Curricular:	Software de Base
Carácter de la UA:	Obligatoria

IV. Objetivos de la formación profesional.

Objetivos del programa educativo:

Formar profesionistas con los conocimientos, habilidades y actitudes necesarios para contribuir en cualquiera de los procesos de la Ingeniería de Software para proponer soluciones de calidad al manejo automatizado de información dentro de las organizaciones, aplicando un enfoque sistemático, disciplinado y cuantificado en la formulación, planeación, análisis, diseño, implantación y mantenimiento de software, así como la generación de conocimiento, metodologías y métricas en torno a la Ingeniería de Software

Objetivos del núcleo de formación:

Esta unidad de aprendizaje en conjunto con las otras unidades que conforman el núcleo integral de formación de la Licenciatura en Ingeniería de software que tiene como propósito:

Proveer al alumno/a de escenarios educativos para la integración, aplicación y desarrollo de los conocimientos, habilidades y actitudes que le permitan el desempeño de las funciones, tareas y resultados ligados directamente a las dimensiones y ámbitos de intervención profesional o campos emergentes de la misma.



Objetivos del área curricular o disciplinaria:

El área curricular establece 3 objetivos los cuales están relacionados con esta unidad de aprendizaje y a continuación se indican.

Aplicar la teoría, técnicas y metodologías para el diseño y construcción de intérpretes y compiladores, presentando las principales herramientas para su generación automática.

Aplicar la teoría, técnicas y metodologías para el diseño y construcción de sistemas operativos, aplicaciones para microcontroladores de propósito específico, con énfasis en cada uno de sus componentes: manejo del procesador, manejo de memoria, administración de dispositivos y manejo de información.

Emplear las principales categorías y herramientas de base necesarias para la configuración, el arranque, el uso eficiente y la operación de los sistemas de cómputo.

V. Objetivos de la unidad de aprendizaje.

Aplicar las diferentes herramientas para la administración y programación de sistemas comparándolas para la construcción de software especializado.

VI. Contenidos de la unidad de aprendizaje, y su organización.

Unidad 1. Introducción de herramientas para administrar y programar sistemas.
Objetivo: Conocer y distinguir herramientas de sistemas comprendiendo e identificando los requerimientos específicos y los rasgos que los categorizan para crear, mejorar o actualizar software en base a los requerimientos propios de cada caso, evaluando la contribución de las herramientas de sistemas para el desarrollo de programas.
Contenidos: 1.1 Razones para emplear herramientas para desarrollar sistemas 1.2 Evolución del surgimiento de las herramientas de administración y



programación 1.3 Uso de herramientas para desarrollar software especializado		
Métodos, estrategias y recursos educativos		
• Clase Magistral • Método Cognitivo • Técnicas interactivas		
Actividades de enseñanza y de aprendizaje		
Inicio	Desarrollo	Cierre
1.1. Clase Magistral que proporcione ideas relevantes acerca de las diferentes razones por las que se inicia el desarrollo de sistema y porque se hace cada vez más necesaria en la vida cotidiana 1.2. Método cognitivo mediante la lectura de un texto que contenga la evolución de las herramientas de administración y programación 1.3 Técnicas interactivas que permitan recordar por lo menos cinco herramientas de una clasificación dada.	1.1. Crear un cuadro sinóptico con las características razones principales que definen el surgimiento de los sistemas. 1.2 Debate sobre las diferencias entendidas entre las herramientas de administración y las de programación creando un cuadro comparativo 1.3 Elaborar una sopa de letras con las principales herramientas para desarrollar un sistema	1.1. Verificar la comprensión mediante la aplicación de un problema real donde cada uno propondrá algún sistema de innovación aclarando que necesidad cubriría. 1.2 Verificar resultados sobre las diferencias y como aplicarlas en algún caso de aplicación. 1.3 Evaluar aprendizaje mediante un cuestionario de preguntas sobre el tema.
(3 Hrs.)	(3 Hrs.)	(3 Hrs.)
Escenarios y recursos para el aprendizaje (uso del alumno)		
Escenarios	Recursos	
Salón de clase	Computadora Pizarrón Electrónico	



Unidad 2. Herramientas de administración de sistemas.

Objetivo:

Diferenciar las características propias de cada herramienta de administración reconociendo y evaluando las ventajas y desventajas de cada herramienta de administración para mejorar y aplicar la herramienta más conveniente en diferentes casos según las necesidades de desarrollo, y la efectividad de los procesos de programación de sistemas.

Contenidos:

- 2.1 Análisis de sistemas
- 2.2 Herramientas CASE
- 2.3 Administración de riesgos
- 2.4 Diseño de sistemas

Métodos, estrategias y recursos educativos

- Método de caso
- Técnicas de corrillos
- Aprendizaje orientado a proyectos

Actividades de enseñanza y de aprendizaje

Inicio	Desarrollo	Cierre
2.1 Método de caso se forman equipos donde se distribuye una lectura sobre Análisis de sistemas que se realiza durante los primeros 15 minutos.	2.1. Debate acerca de la problemática de la lectura donde cada integrante del equipo formara una estrategia de solución donde destacaran las herramientas que se utilizaran especificando su uso.	2.1. Verificar las estrategias de solución mediante un informe detallado de cada una de las soluciones planteadas
2.2 Se inicia mediante una investigación en internet sobre qué, cuáles y que clasificación hay de Herramientas CASE	2.2. Con la investigación se realiza una extracción de los nombres de las herramientas más	2.2 Evaluar el aprendizaje de por lo menos 10 herramientas plasmadas en el memorama.



2.3 Lluvia de ideas escrita sobre la administración de riesgos.	utilizadas con las que armara un memorama	
2.3 Debate donde se organizaran las ideas que se escribieron mediante una discusión y se elabora un informe.	2.3 Debate donde se organizaran las ideas que se escribieron mediante una discusión y se elabora un informe.	2.3 Se presentan los informes que se elaboraron para su discusión.
2.4 Técnica de trabajo cooperativo que permite identificar que es para que sirve y como se realiza el diseño de sistemas.	2.4 Después de conocer las principales características así como su forma de realizar se elabora un ejercicio práctico para complementar el reporte.	2.4. Se entrega reporte de un caso práctico donde destacara el uso de las herramientas empleadas y por lo menos dos sugerencias más que puedan sustituir a las empleadas
3 (Hrs.)	10 (Hrs.)	3(Hrs.)
Escenarios y recursos para el aprendizaje (uso del alumno)		
Escenarios		Recursos
Sala de Computo Salón de clases		Computadoras Pizarrón interactivo Marcadores Internet

Unidad 3. Herramientas de programación de sistemas

Objetivo:

Distinguir los rasgos particulares de cada herramienta de programación identificando las ventajas e inconvenientes de cada herramienta de programación para identificar y evaluar las condiciones y requisitos establecidos por la industria del software proponiendo y aplicando las herramientas de



programación más adecuadas.

Contenidos:

- 3.1 Teoría de Programación de sistemas
- 3.2 Administración de lenguajes de programación
- 3.3 Administración de sistemas operativos y redes
- 3.4 Administración de centros de cómputo

Métodos, estrategias y recursos educativos

- Técnicas de selección
- Técnicas de corrillos
- Aprendizaje orientado a proyectos

Actividades de enseñanza y de aprendizaje

Inicio	Desarrollo	Cierre
3.1.- Técnicas de selección sobre de Teoría de Programación de sistemas. En base a la lectura de tres artículos diferentes.	3.1 Elaborar una definición en base a la lectura determinando un mapa mental que plasme la definición y sus características sobre el tema.	3.1 Verificar conocimiento mediante preguntas claras presentando un informe sobre lo entendido.
3.2 Técnicas de exploración de hechos para identificar la Administración de lenguajes de programación mediante archivos, investigación y la aplicación de sus conocimientos.	3.2 Mediante la técnica de corrillos proporciona a cada equipo preguntas en tarjetas u hojas. Para la elaboración de un reporte.	3.3 Se evalúa las respuestas de cada equipo a fin de verificar las respuestas así como para unificar criterios.
3.3 Se imparte una lectura sobre la administración de sistemas operativos y redes donde se destacan las ideas principales durante un lapso de tiempo	3.3.Por la técnica de resumen sobre la lectura realizada se elabora un informe por equipo que se expondrá.	3.3. Se evalúa el informe más completo.



formando equipos		
3.4 Clase magistral sobre administración de centros de cómputo y características principales elementos a tomar en cuenta.	3.4 Elaborar diagrama que represente la organización que debe llevarse en la administración del centro de cómputo	3.4 Verificar la administración de centros de cómputo mediante preguntas que determinen contenido del aprendizaje.
(3 Hrs.)	(10 Hrs.)	(3 Hrs.)
Escenarios y recursos para el aprendizaje (uso del alumno)		
Escenarios		Recursos
Salón de clases Sala de Computo		Pizarron Proyector Sala de evaluación digital Computadora

Unidad 4. Selección e implementación de las herramientas adecuadas para un caso específico de desarrollo.

Objetivo:

Determinar los requerimientos de administración, programación, soporte y costo del desarrollo a partir de la estructura y el análisis de una organización aplicando las técnicas, estándares y criterios más convenientes para diseñar un programa de administración y programación en un caso particular.

Contenidos

- 4.1 Selección de las herramientas de administración más adecuadas
- 4.2 Técnicas y criterios para implementar herramientas de programación
- 4.3 Diseño e implementación herramientas de sistemas para desarrollar software específico.



Métodos, estrategias y recursos educativos		
Actividades de enseñanza y de aprendizaje		
Inicio	Desarrollo	Cierre
4.1,4.2.- Técnica de preguntas recíprocas sobre selección de las herramientas de administración más adecuadas en base a conocimientos aprendidos en la unidad anterior.	4.1,4.2 En base al planteamiento de un problema real se elabora un diagrama que sea representativo de solución sobre el caso y un reporte complementario.	4.1,4.2 Verificar mediante la exposición del diagrama la solución más adecuada sobre el problema.
4.3 Clase magistral sobre técnicas de exploración de hechos para identificar requerimientos para el diseño e implementación de herramientas de sistemas para desarrollar software.	4.3. Método del caso (preparación individual y posteriormente discusión en pequeños grupos) sobre las técnicas de exploración de hechos para identificar requerimientos	4.3. Aprendizaje orientado a proyectos colaborativos sobre las técnicas de exploración de hechos adecuados al proyecto a desarrollar.
(3 Hrs.)	(13Hrs.)	(3 Hrs.)
Escenarios y recursos para el aprendizaje (uso del alumno)		
Escenarios	Recursos	
Salón de clase Sala de computo	Pizarrón Computadora Libros Aula de evaluación digital	



VII. Acervo bibliográfico

Básica:

1. Pressman R. S. (2006), *Ingeniería de Software: Un enfoque Práctico*, México: McGraw-Hill, ISBN: 8448100263
2. Greenfield, J., Short, K., Cook, S., & Kent, S. (2003) *Software Factories: Assembling Applications with Patterns, Models, Frameworks, and Tools*. New York, NY, USA: ACM, ISBN: 1581137516
3. Jacky, J. (2001) *The Way of Z: Practical Programming with Formal Methods*. USA: Cambridge University Press.

Complementaria:

1. Jacobson, I., Booch, G., & Rumbaugh, J. (2000) *El Proceso Unificado de Desarrollo de Software*. Madrid: Addison-Wesley
2. Kelly, S. & Tolvanen, J. (2008) *Domain-Specific Modeling: Enabling Full Code Generation*. New Jersey, USA: John Wiley & Sons, Inc.
3. Stahl, T., Völter, M. (2006) *Model-Driven Software Development*. England: John Wiley & Sons, Inc.
4. Pastor, O., Molina, J. (2004) *Model driven architecture in practice*. Valencia, España: Springer.



VIII. Mapa curricular

