
	FORMATO DE SYLLABUS	Código: AA-FR-003	
	Macroproceso: Direccionamiento Estratégico	Versión: 01	
	Proceso: Autoevaluación y Acreditación	Fecha de Aprobación: 27/07/2023	

FACULTAD:	FACULTAD DE INGENIERÍA		
PROYECTO CURRICULAR:	Modelos de Programación	CÓDIGO PLAN DE ESTUDIOS:	

I. IDENTIFICACIÓN DEL ESPACIO ACADÉMICO

NOMBRE DEL ESPACIO ACADÉMICO: Modelos de Programación

Código del espacio académico:		Número de créditos académicos:			3	
Distribución horas de trabajo:	HTD	4	HTC	2	HTA	3
Tipo de espacio académico:	Asignatura	X	Cátedra			

NATURALEZA DEL ESPACIO ACADÉMICO:

Obligatorio Básico	X	Obligatorio Complementario		Electivo Intrínseco		Electivo Extrínseco	
--------------------	---	----------------------------	--	---------------------	--	---------------------	--

CARÁCTER DEL ESPACIO ACADÉMICO:

Teórico		Práctico		Teórico-Práctico	X	Otros:		Cuál: _____
---------	--	----------	--	------------------	---	--------	--	-------------

MODALIDAD DE OFERTA DEL ESPACIO ACADÉMICO:

Presencial	X	Presencial con incorporación de TIC	X	Virtual		Otros:		Cuál: _____
------------	---	-------------------------------------	---	---------	--	--------	--	-------------

II. SUGERENCIAS DE SABERES Y CONOCIMIENTOS PREVIOS

1. Programación Orientada a Objetos
2. Principios de diseño orientado a objetos
3. Modelamiento con diagrama de clases

III. JUSTIFICACIÓN DEL ESPACIO ACADÉMICO

La ingeniería de software implica abordar problemas desde diferentes enfoques y perspectivas de análisis, diseño e implementación, los patrones de diseño permiten al estudiante aprender a partir de experiencias de soluciones generales a situaciones recurrentes y evitar prácticas incorrectas que pueden comprometer la calidad de las soluciones software.

La comprensión de los patrones de diseño y el desarrollo de software por componentes permite la aplicación de los principios de Ingeniería de software, facilitando la implementación y despliegue de soluciones de software, así como la reutilización en diferentes tipos de proyectos. de soluciones fácilmente integrables robustas, mantenibles y escalables.

El desarrollo de software aplicando patrones de diseño y basado en componentes probados, reducen tanto el tiempo de desarrollo, como el mantenimiento, a la vez que mejoran la escalabilidad y flexibilidad de las soluciones obtenidas

Los patrones de diseño y componentes facilitan el manejo a diferente nivel de abstracción y proporcionan un lenguaje común entre los integrantes del equipo, facilitando la documentación y comunicación para mejorar la colaboración.

IV. OBJETIVOS DEL ESPACIO ACADÉMICO (GENERAL Y ESPECÍFICOS)

Objetivo General:

Presentar al estudiante la conceptualización y aplicación de modelos de programación basado en patrones de diseño y desarrollo de componentes para la obtención de soluciones que cumplan los principios de la Ingeniería de Software.

Objetivos específicos:

1. Proporcionar al estudiante la base conceptual de la programación basada en patrones de diseño y desarrollo de componentes, explicando sus características, conformación así como la ejemplificación mediante diagramas e implementación en lenguajes de programación para facilitar la comprensión de soluciones computacionales.
2. Establecer escenarios con problemáticas en diferentes grados de dificultad que permitan al estudiante la identificación del patrón acorde al problema así como el diseño y desarrollo de la solución, mediante un lenguaje de programación.
3. Plantear situaciones que permitan a los estudiantes mediante diferentes tipos de actividades, diseñar e implementar soluciones software aplicando patrones de diseño y la adecuada implementación mediante lenguajes de programación.
4. Efectuar seguimiento a la comprensión y adecuada aplicación de los patrones de diseño y desarrollo de componentes a través de diferentes dinámicas y mecanismos de evaluación, que permitan determinar el grado de apropiación de las temáticas abordadas.

V. PROPÓSITOS DE FORMACIÓN Y DE APRENDIZAJE (PFA) DEL ESPACIO ACADÉMICO

Competencias	Dominio-Nivel	Dominio-Nivel	RA	Resultados de Aprendizaje
Selecciona patrones de diseño probados en la solución de problemas recurrentes para el diseño de sistemas modulares flexibles y reutilizables		Cognitivo - Analizar	1	Determinar situaciones en las que el uso de patrones de diseño dan respuesta a requisitos del sistema
		Cognitivo - Analizar	2	Identificar patrones de diseño para resolver problemas específicos aplicando principios que proporcionen principios como modularidad, flexibilidad, extensibilidad a la solución propuesta

Establece diseños de software que dan solución a problemas específicos aplicando patrones de diseño y componentes	Cognitivo - Aplicar	3	Evaluar la pertinencia de los patrones seleccionados para un problema específico justificando las ventajas que presenta su incorporación a la solución.
	Cognitivo - Crear	4	Diseñar componentes basados en patrones de diseño, que puedan ser implementados mediante métodos de desarrollo de componentes
	Cognitivo - Conocer	5	Establecer los lenguajes de programación que correspondan a las características del diseño definido para la solución del problema
Implementa diseños de software, que puedan ser integrados en diferentes contextos	Cognitivo - Aplicar	6	Implementar la solución del problema de manera modular aplicando desarrollo por componentes, mediante lenguajes de programación que mantengan el diseño
	Cognitivo - Aplicar	7	Integrar componentes que permitan el despliegue de funcionalidades en diferentes contextos
Participa en la definición de las dinámicas de comunicación y documentación de los proyectos de software, utilizando diseños basados en patrones y componentes	Afectivo	8	Documentar soluciones de software utilizando especificaciones que permiten la aplicación de representaciones basadas en patrones de diseño y desarrollo de componentes

VI. CONTENIDOS TEMÁTICOS

- Patrones de diseño
 - Creacionales
 - Estructurales
 - De comportamiento
- Anti-patrones
 - Antipatrones de Programación
 - Antipatrones de Diseño de Software
- Componentes
 - Concepto
 - Tipos
 - Evolución
 - Diseño de componentes
 - Construcción

VII. ESTRATEGIAS DE ENSEÑANZA QUE FAVORECEN EL APRENDIZAJE

Tradicional		Basado en Proyectos	X	Basado en Tecnología	
Basado en Problemas	X	Colaborativo	X	Experimental	
Aprendizaje Activo	X	Autodirigido		Centrado en el estudiante	

VIII. EVALUACIÓN

Resultados de aprendizaje (RA) a ser evaluados:	Resultados de aprendizaje asociados a las evaluaciones					
	Actividades Entregables	Talleres	Parciales	Informes de proyecto final	Proyecto final	Exposiciones
RA01	X	X	X	X	X	X
RA02	X	X	X	X	X	X
RA03	X	X	X	X	X	X
RA04	X	X	X	X	X	X
RA05	X	X	X	X	X	
RA06	X	X	X	X	x	
RA07	X	X	X	X		
RA08	X	X	X	X		
Tipo de evaluación**	EF,	EF,EHP	EE,EHP		EHP,EBP,EOP, ED	EF, EOP
Porcentaje de evaluación (%)	11	14	35	15	15	10
Trabajo Individual (I) o Grupal (G)	I,G	I,G	I	I,G	I,G	I,G
Tipo de nota	0-5	0-5	0-5	0-5	0-5	0-5

IX. MEDIOS Y RECURSOS EDUCATIVOS

Salón con tablero y Videobeam.
 Plataforma para gestión de contenidos de aprendizaje
 Entorno integrado de desarrollo de software
 Espacio de almacenamiento con acceso mediante internet
 Acceso a fuentes de información en línea

X. PRÁCTICAS ACADÉMICAS - SALIDAS DE CAMPO			
No aplica			
XI. BIBLIOGRAFÍA			
<p>Básicas:</p> <p>Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Addison Wesley</p>			
<p>Complementarias:</p> <p>Elisabeth Freeman, Eric Freeman, Bert Bates, Kathy Sierra. Head First Desing Patterns. O'Reilly Media</p> <p>James W. Cooper. Desing Patterns. Addison Wesley</p> <p>Bertrand Meyer. Construcción de Software Orientado a Objetos. Prentice Hall.</p> <p>Horstmann Cornell, Core Java 2 vol 1 y vol 2. Pretince Hall.</p> <p>Bruce Eckel. Thinking Java. Pretince Hall</p> <p>Deitel & Deitel. Java2 How To Program. Prentice Hall.</p> <p>Agustín Froufe Quintas. Java 2 Manual de usuario y tutorial. Alfaomega.</p> <p>Guía de certificación de java. Sun Microsystem.</p>			
<p>Páginas web</p> <p>https://stackoverflow.com/</p> <p>https://www.reddit.com/</p> <p>https://martinfowler.com/</p> <p>https://refactoring.guru/</p> <p>https://www.dzone.com/</p> <p>https://github.com/</p>			
XII. SEGUIMIENTO Y ACTUALIZACIÓN DEL SYLLABUS			
Fecha revisión por Consejo Curricular:			
Fecha aprobación por Consejo Curricular:		Número de acta:	

**Tipo de Evaluación	Abreviatura
1. Evaluación de habilidad	EHP
2. Evaluación basada en p	EBP
3. Evaluación oral o prese	EOP
4. Evaluación escrita	EE
5. Evaluación formativa	EF
6. Evaluación de desempe	ED