



Casa abierta al tiempo

UNIVERSIDAD AUTONOMA METROPOLITANA

PROGRAMA DE ESTUDIOS

UNIDAD	CUAJIMALPA	DIVISION	CIENCIAS NATURALES E INGENIERIA	1 / 4
NOMBRE DEL PLAN LICENCIATURA EN INGENIERIA EN COMPUTACION				
CLAVE	UNIDAD DE ENSEÑANZA-APRENDIZAJE		CRED.	8
4605004	PATRONES DE DISEÑO Y ARQUITECTURA DE SOFTWARE		TIPO	OPT.
H. TEOR. 3.0	SERIACION		TRIM.	
H. PRAC. 2.0	4604048		VII AL XII	

OBJETIVO(S) :

Objetivo General:

Al final de la UEA el alumno será capaz de aplicar patrones de diseño y arquitectura de software para la construcción de sistemas de software.

Objetivos Parciales:

1. Comprender la técnica de los patrones de arquitectura de software y aplicarla en el desarrollo de software.
2. Comprender las diferentes variantes de la arquitectura Modelo-Vista-Controlador y su papel en el desarrollo de software.
3. Comprender la técnica de los patrones de diseño de software y aplicarla en el desarrollo de software.
4. Aplicar algunas de las tecnologías que soportan la implementación de la arquitectura Modelo-Vista-Controlador y sus variantes.

CONTENIDO SINTETICO:

1. El papel de los patrones en el desarrollo de software.
 - 1.1. El uso de patrones en ingeniería de software.
 - 1.2. Componentes de un patrón.
 - 1.3. Categorías de patrones en la ingeniería de software.
 - 1.4. Ejemplos de patrones.
2. Patrones arquitectónicos.
 - 2.1. Patrones de arquitectura lógica.
 - 2.2. Patrones de arquitectura física.
 - 2.3. Caso de estudio sobre la aplicación de los patrones arquitectónicos.



Casa abierta al tiempo

UNIVERSIDAD AUTONOMA METROPOLITANA

ADECUACION
PRESENTADA AL COLEGIO ACADEMICO
EN SU SESION NUM. 477

EL SECRETARIO DEL COLEGIO

3. Patrones de diseño.
 - 3.1. Patrones de creación.
 - 3.2. Patrones estructurales.
 - 3.3. Patrones de comportamiento.
 - 3.4. Caso de estudio sobre la aplicación de los patrones de diseño.
4. La arquitectura lógica Modelo-Vista-Controlador (MVC) y sus variantes.
 - 4.1. Modelo-Vista-Presentador (MVP).
 - 4.2. Presentación-Abstracción-Control (PAC).
 - 4.3. Modelo-Vista-Controlador Jerárquico (MVC-Jerárquico).
 - 4.4. Arquitectura de más de tres capas.
 - 4.5. Caso de estudio sobre la adopción de una de estas variantes.
5. Tecnologías que soportan la implementación de la arquitectura MVC.
 - 5.1. Arquitectura MVC monolítica.
 - 5.2. Arquitectura MVC cliente-servidor
 - 5.3. Casos de estudio.

MODALIDADES DE CONDUCCION DEL PROCESO DE ENSEÑANZA-APRENDIZAJE:

Clase teórico-práctica a cargo del profesor con participación activa del alumno y clase práctica en laboratorio.

Como estrategia de enseñanza el profesor hará exposiciones de los temas en el aula. Promoverá la elaboración de programas claros y comentados que apliquen patrones de diseño y arquitectura de software.

El profesor diseñará experiencias de aprendizaje por problemas, con nivel de complejidad incremental, tanto en el aula como en el laboratorio. El alumno analizará los problemas planteados y aplicará los conceptos de patrones de diseño y de arquitectura de software aprendidos para darles solución.

Las habilidades transversales que deberá adquirir el alumno asociadas a esta UEA son:

(Ht0) Lenguaje disciplinar: reforzará y aprenderá nuevos conceptos relacionados con patrones de diseño y arquitectura de software.

(Ht2) Trabajar armónicamente en equipo: deberán poder transmitir sus ideas para la solución de problemas y recibir retroalimentación a las mismas.

(Ht3) Comunicarse eficazmente en forma oral y escrita en español: presentará



UNIVERSIDAD AUTONOMA METROPOLITANA

Casa abierta al tiempo

ADECUACION
PRESENTADA AL COLEGIO ACADEMICO
EN SU SESION NUM. 477

EL SECRETARIO DEL COLEGIO

NOMBRE DEL PLAN	LICENCIATURA EN INGENIERIA EN COMPUTACION	3/ 4
CLAVE	4605004	PATRONES DE DISEÑO Y ARQUITECTURA DE SOFTWARE

al profesor, en forma oral y escrita, informes de los trabajos realizados.

(Ht4) Comprender perfectamente textos técnicos en español: el profesor deberá proporcionar lecturas sobre temas relacionados con el contenido sintético. Es recomendable que las lecturas ayuden al alumno a encontrar soluciones a los temas tratados en esta UEA.

(Ht5) Comprender textos técnicos en inglés: el profesor deberá proporcionar lecturas asociadas al contenido sintético, para que posteriormente el alumno explique en español lo que entendió de dichas lecturas.

Las habilidades disciplinares que deberá adquirir el alumno asociadas a esta UEA son:

(H1) Abstracción, como la habilidad para conceptualizar un problema que permita plantear una solución al mismo: utilizar patrones de diseño y arquitectura en el desarrollo de software.

(H2) Aplicar modelos y técnicas para diseñar, implementar y probar sistemas eficientes: aplicar métodos y técnicas para diseñar e implementar sistemas de software a gran escala mediante patrones de diseño y arquitectura de software.

(H5) Desarrollar la capacidad para tomar decisiones: Desarrollar la capacidad para tomar decisiones durante el desarrollo de software basado en el uso de patrones de diseño y arquitectura.

Las actitudes que se fomentarán en el alumno asociadas a esta UEA son:

- (A1) Liderazgo en equipos de trabajo multidisciplinario.
- (A3) Disciplina para aplicar los conocimientos adquiridos.
- (A5) Voluntad de mantenerse actualizado en su área de trabajo.
- (A6) Responsabilidad y ética en su desempeño profesional.
- (A7) Conciencia de la realidad social y responsabilidad ecológica.
- (A8) Adaptación a nuevos o diferentes entornos tecnológicos.

MODALIDADES DE EVALUACION:

Evaluación Global:

Se ponderarán las siguientes actividades a criterio del profesor:



UNIVERSIDAD AUTONOMA METROPOLITANA

ADECUACION
PRESENTADA AL COLEGIO ACADEMICO
EN SU SESION NUM. 477

EL SECRETARIO DEL COLEGIO

- Tareas individuales y en equipo.
- Exposición y entrega de reportes de prácticas de laboratorio.
- Evaluaciones periódicas.
- Evaluación terminal.
- Participación en los procesos de argumentación, tanto en las sesiones de teoría como en las de práctica.
- Evaluación de la comprensión de lecturas en inglés y en español, mediante reportes escritos o de forma oral en español.

Evaluación de Recuperación:

El alumno deberá presentar una evaluación teórico-práctica que contemple los contenidos de la unidad de enseñanza-aprendizaje de tal forma que se cumpla el objetivo general de la UEA.

No requiere inscripción previa a la UEA.

BIBLIOGRAFIA NECESARIA O RECOMENDABLE:

1. Cusumano, M. A The factory approach to large-scale software development: Implications far strategy, technology, and structure. Palala Press, 2015.
2. Gamma, E., Helm, R., Johnson, R., Vlissides, J. Design patterns: Elements of reusable object-oriented software. Addison-Wesley, 2007.
3. Larman, C. UML y patrones. Pearson-Prentice Hall, 2003.
4. Patzer, A Foundations of JSP design patterns, APress Media, 2004.
5. Pitt, C. Pro PHP MVC. APress,2012.
6. Tsui, F., Karam, O., Bernal, B. Essentials of software engineering. Jonas & Bartlett Learning books, 2015.
7. Van Vliet, H. Software engineering: Principies and practice. John Wiley & Sons, 2000.



Casa abierta al tiempo

UNIVERSIDAD AUTONOMA METROPOLITANA

ADECUACION
PRESENTADA AL COLEGIO ACADEMICO
EN SU SESION NUM. 477

EL SECRETARIO DEL COLEGIO