



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
 FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN
 PLAN DE ESTUDIOS DE LA LICENCIATURA EN INFORMÁTICA
 Sistema Escolarizado: Modalidad Presencial
 Programa de estudios de la asignatura



Implementación de sistemas

Clave 2533	Semestre 5°	Créditos 8	Campo de conocimiento: Informática	
			Eje de formación: Integración	
Modalidad	Curso (X) Taller () Lab () Seminario() Otros()		Tipo	T (X) P () T/P ()
Carácter	Obligatorio (X) Optativo () Obligatorio E () Optativo E ()		Horas	
Duración (Número de semanas)	16		Semana	Semestre
			Teóricas:	4 Teóricas: 64
			Prácticas:	0 Prácticas: 0
			Total	4 Total 64
Seriación				
Ninguna ()				
Obligatoria ()				
Asignatura antecedente				
Asignatura subsecuente				
Indicativa (X)				
Asignatura antecedente	Programación orientada a objetos			
Asignatura subsecuente	Desarrollo de aplicaciones móviles Desarrollo de aplicaciones web			
Objetivo general:				
Al finalizar el curso, el alumnado construirá un sistema informático con frameworks implementando patrones de arquitectura y de diseño asegurando su funcionamiento con la aplicación de pruebas.				

Objetivos particulares

Al finalizar el curso, el alumnado:

1. Configuraré las herramientas necesarias para construir un sistema.
2. Identificaré los patrones de arquitectura por medio de las características del framework de desarrollo o su implementación en el mismo.
3. Diseñaré escenarios de prueba a partir de una especificación funcional.
4. Construiré componentes del sistema por medio de un framework de desarrollo.
5. Ejecutaré escenarios de prueba para identificar y corregir errores.
6. Integraré un módulo o componente a un sistema.

Índice temático			
Unidad	Tema	Horas Semestre	
		Teóricas	Prácticas
1	Ambiente de trabajo	12	0
2	Diseño de la arquitectura	8	0
3	Diseño de pruebas	8	0
4	Implementación de sistema	20	0
5	Implementación de pruebas	8	0
6	Integración	8	0
Total		64	

Estrategias didácticas

- Exposición.
- Análisis de lecturas o videos.
- Aprendizaje basado en problemas.
- Aprendizaje basado en proyectos.

Evaluación del aprendizaje

- Listas de cotejo.
- Exámenes.
- Proyectos
- Participación en clase.
- Rúbricas

Perfil profesiográfico del docente	
Título o grado	Licenciatura en Informática o equivalente, preferentemente con estudios de posgrado con orientación a las tecnologías de la información y las organizaciones.
Experiencia docente	Mínima deseable de 2 años impartiendo clases en nivel media superior y/o superior.
Otras características	<p>Experiencia Profesional mínima de 3 años en área de conocimiento. Experiencia en el modelado de arquitecturas con Lenguaje Unificado de Modelado (UML) y en programación de sistemas con algún framework de desarrollo tanto front-end como back-end.</p> <p>Para profesoras/es de nuevo ingreso: Haber aprobado el “Curso Fundamental para profesores de Nuevo Ingreso (Didáctica Básica)” que imparte la Facultad de Contaduría y Administración, así como cubrir satisfactoriamente los requisitos impuestos por el departamento de selección y reclutamiento de la Facultad de Contaduría y Administración.</p> <p>Para profesoras/es que ya imparten clases en la Facultad: Haber participado recientemente en cursos de actualización docente y de actualización disciplinar con un mínimo de 20 horas.</p> <p>Compartir, respetar y fomentar los valores fundamentales que orientan a la Universidad Nacional Autónoma de México.</p>

Bibliografía básica
<ul style="list-style-type: none"> ◦ Cabrera, J. (2017). <i>Modular design frameworks: a projects-based guide for ui/ux designers</i>. New York: Apress. ◦ Chacon, S. y Straub, B. (2014). <i>Pro git</i> (2da Ed.). Berkeley: Apress. ◦ Dauton, S. (2018). <i>Git: controle la gestión de sus versiones (conceptos, utilización y casos prácticos)</i>. Barcelona: Ediciones ENI. ◦ Li, K. y Wu, M. (2004). <i>Effective software test automation: developing an automated software testing tool</i>. Alameda: Sybex. ◦ Madeyski, L. (2010). <i>Test-driven development: an empirical evaluation of agile practice</i>. Berlin: Springer. ◦ Sarcar, V. (2016). <i>Java design patterns: a tour of 23 gang of four design patterns in Java</i>. Berkeley: Apress.

Bibliografía complementaria
<ul style="list-style-type: none"> ◦ Laster, B. (2016). <i>Professional Git</i>. Indianapolis: Wrox. ◦ Pressler, R. (Ed.) (1985). <i>Framework: A developer's handbook</i>. Culver: Ashton-tate.

Mesografía (referencias electrónicas)

- OpenJS Foundation. (s. f.). *node js*. Recuperado el 6 de marzo de 2021 de: <https://nodejs.org/en/>
- Git. (s. f.). *Git –everything-is-local*. Recuperado el 6 de marzo de 2021 de: <https://git-scm.com>
- GitHub. (s. f.). *Where the world builds software*. Recuperado el 6 de marzo de 2021 de: <https://github.com>
- Google. (s. f.). *Flutter 2 is here!*. Recuperado el 6 de marzo de 2021 de: <https://flutter.dev>
- VMWare. (s. f.). *spring*. Recuperado el 6 de marzo de 2021 de: <https://spring.io>