



**Centro de Investigación y de Estudios Avanzados  
Del Instituto Politécnico Nacional  
Secretaría Académica**

**Registro de Cursos o Asignaturas**

<b>Nombre Completo del Programa de Posgrado</b>		<b>Maestría en Ciencias en Ingeniería Eléctrica</b>		
<b>Nombre Completo del Curso</b>		<b>Ingeniería de Software I</b>		
<b>Tipo de Curso</b>		<b>Electivo</b>	<b>Créditos</b>	<b>8</b>
<b>Número de horas</b>		<b>Teóricas:</b>	<b>60</b>	<b>Prácticas:</b>
		Presenciales		No presenciales
<b>Profesores que impartirán el curso</b>				
<b>Mario Ángel Siller González Pico</b>				
<b>Objetivos del curso:</b>	<b>General</b>	1. Entender conceptos y principios básicos de la ingeniería de software. 2. Desarrollar habilidades de desarrollo de software bajo los principios que rigen la ingeniería de software. 3. Aplicar los conocimientos a proyectos prácticos de desarrollo.		
	<b>Específicos</b>			
<b>Contenidos temáticos</b>				
<b>1. Introducción y revisión general de la ingeniería de software</b>				
1.1. Introducción				
1.2. Procesos de software				
1.3. Sistemas socio-técnicos				
<b>2. Requerimientos</b>				
2.1. Requerimientos de software				
2.2. Ingeniería de procesos de requerimientos				
2.3. Modelos de sistemas				
2.4. Prototipos de software				
2.5. Especificación formal				
<b>3. Diseño</b>				
3.1. Diseño de arquitecturas				
3.2. Arquitecturas distribuidas de sistemas				
3.3. Diseño orientado a objetos				
3.4. Diseño de software de tiempo real				
3.5. Diseño y reúso				
3.6. Diseño e interfaces				
<b>4. Sistemas críticos</b>				
4.1. Confiabilidad				
4.2. Especificación de sistemas críticos				
4.3. Desarrollo de sistemas críticos				
<b>5. Verificación y validación</b>				
5.1. Verificación y validación				

5.2. Pruebas
5.3. Validación de sistemas críticos
<b>6. Administración</b>
6.1. Administración de gente
6.2. Estimación de costo
6.3. Administración de la calidad
<b>7. Evolución</b>
7.1. Sistemas heredados
7.2. Cambios de Software
7.3. Reingeniería de Software
7.4. Gestión de Configuración

**Bibliografía**

1. "Software Engineering" by Ian Sommerville, 9th and 10th Edition, Pearson.
2. Guide to the Software Engineering Body of Knowledge (SWEBOK), IEEE Computer Society.
3. "Software Engineering, A practitioner's Approach" by Roger S. Pressman, McGrawHill International Edition, 9th Edition.
4. "Software Engineering" by K.K. Aggarwal & Yogesh Singh, New Age International Publishers.
5. "Software Requirements (Developer Best Practices)", 3rd Edition by Karl Wiegers, Joy Beatty.
6. "Requirements Engineering Processes and Techniques", Gerald Kotonya and Ian Sommerville, John Wiley & Sons.
7. Formal Methods: State of the Art and Future Directions, Edmund M. Clarke and Jeannette M. Wing, report by the Working Group on Formal Methods for the ACM Workshop on Strategic Directions in Computing Research, ACM Computing Surveys, vol. 28, no. 4, December 1996, pp. 626-643. Also CMU-CS-96-178.
8. "Code Complete: A Practical Handbook of Software Construction", Steve McConnell Principles of Model Checking, Christel Baier and Joost-Pieter Katoen, MIT Press 2008.
9. "Software Testing and Analysis: Process, Principles and Techniques" by Mauro PezzandMichal Young, John Wiley & Sons 2008.
10. "The Art of Software Testing" by Glenford J. Myers Corey Sandler, Tom Badgett, 3rd Edition.

**Criterios de evaluación**

1 proyecto	0%
15 tareas	0%
2 exámenes	0%
<b>Total</b>	<b>00%</b>

**Contribución del curso al perfil de egreso del programa**

**Conocimientos:**

**Habilidades:**

**Actitudes y valores:**