

Programación semanal

En la programación semanal te presentamos un **reparto del trabajo de la asignatura** a lo largo de las semanas del cuatrimestre.

	Contenido teórico	Trabajos (6 puntos)	Eventos (3 puntos)	Laboratorios (6 puntos)
Semana 1	Tema 1. Marco de Ingeniería <i>software</i> 1.1. ¿Cómo estudiar este tema? 1.2. Introducción a la Ingeniería <i>software</i> 1.3. Características del <i>software</i> 1.4. Modelos de proceso de desarrollo <i>software</i> 1.5. Marco de referencia estándar para el proceso de desarrollo		Asistencia a 2 sesiones presenciales virtuales a lo largo del cuatrimestre (0,2 puntos cada una) Test Tema 1 (0,1 puntos)	
Semana 2	Tema 2. Proceso Unificado de desarrollo <i>software</i> 2.1. ¿Cómo estudiar este tema? 2.2. Proceso Unificado 2.3. Dimensiones del Proceso Unificado			
Semana 3	Tema 2. Proceso Unificado de desarrollo <i>software</i> (continuación) 2.4. Proceso de desarrollo dirigido por casos de uso 2.5. Flujos de trabajo fundamentales de Procesos Unificados		Test Tema 2 (0,1 puntos)	Laboratorio #1: Modelado con ArgoUML (3 puntos)
Semana 4	Tema 3. Desarrollo <i>software</i> ágil 3.1. ¿Cómo estudiar este tema? 3.2. Introducción al desarrollo <i>software</i> ágil 3.3. Características del desarrollo <i>software</i> ágil 3.4. Principios del desarrollo <i>software</i> ágil 3.5. El desarrollo ágil y el factor humano 3.6. Marco de desarrollo ágil <i>scrum</i>		Test Tema 3 (0,1 puntos)	
Semana 5	Tema 4. Estilos de diseño de <i>software</i> distribuido 4.1. ¿Cómo estudiar este tema? 4.2. Introducción a los sistemas <i>software</i> distribuidos 4.3. Modelos de comunicación en sistemas <i>software</i> distribuidos 4.4. Capa intermedia o <i>middleware</i> 4.5. Patrones de arquitectura para sistemas distribuidos		Test Tema 4 (0,1 puntos)	
Semana 6	Tema 5. Arquitecturas <i>software</i> orientadas a servicios 5.1. ¿Cómo estudiar este tema? 5.2. Introducción al diseño de arquitecturas orientadas a servicios 5.3. Proceso general de diseño de servicios <i>software</i> 5.4. Aspectos de diseño de servicios <i>software</i> 5.5. Composición de servicios 5.6. Arquitectura de servicios web 5.7. Componentes de infraestructura y SOA	Trabajo: Seguridad y servicios web (2 puntos)	Foro: ¿Qué aportan de novedad las arquitecturas orientadas a servicios? (1,4 puntos) Test Tema 5 (0,1 puntos)	

	Contenido teórico	Trabajos (6 puntos)	Eventos (3 puntos)	Laboratorios (6 puntos)
Semana 7	<p>Tema 6. Introducción al desarrollo software dirigido por modelos</p> <p>6.1. ¿Cómo estudiar este tema?</p> <p>6.2. Introducción al desarrollo <i>software</i> dirigido por modelos</p> <p>6.3. El concepto de modelo</p> <p>6.4. Desarrollo <i>software</i> dirigido por modelos</p> <p>6.5. Marco de referencia de desarrollo dirigido por modelos MDA</p> <p>6.6. Elementos de un modelo de transformación</p> <p>6.7. Referencias</p>		<p>Test Tema 6 (0,1 puntos)</p>	
Semana 8	<p>Tema 7. Diseño de interfaces gráficas de usuario</p> <p>7.1. ¿Cómo estudiar este tema?</p> <p>7.2. Introducción a las interfaces de usuario</p> <p>7.3. Principios de diseño de interfaces de usuario</p> <p>7.4. Diseño centrado en el usuario</p> <p>7.5. Proceso de análisis y diseño de la interfaz gráfica de usuario</p> <p>7.6. Usabilidad de interfaces de usuario</p> <p>7.7. Accesibilidad en interfaces de usuario</p> <p>7.8. Referencias</p>	<p>Trabajo: Evaluación de accesibilidad de forma automatizada (2 puntos)</p>	<p>Test Tema 7 (0,1 puntos)</p>	
Semana 9	<p>Tema 8. Diseño de aplicaciones web</p> <p>8.1. ¿Cómo estudiar este tema?</p> <p>8.2. Introducción a las aplicaciones web</p> <p>8.3. Características de las aplicaciones web</p> <p>8.4. Interfaz de usuario de una aplicación web</p> <p>8.5. Diseño de navegación</p> <p>8.6. Protocolo de interacción base para aplicaciones web</p> <p>8.7. Estilos de arquitectura para aplicaciones web</p> <p>8.8. Referencias</p>		<p>Test Tema 8 (0,1 puntos)</p>	
Semana 10	<p>Tema 9. Estrategias y técnicas de pruebas <i>software</i></p> <p>9.1. ¿Cómo estudiar este tema?</p> <p>9.2. Fundamentos de las pruebas <i>software</i></p> <p>9.3. Estrategias de prueba</p>			
Semana 11	<p>Tema 9. Estrategias y técnicas de pruebas <i>software</i> (continuación)</p> <p>9.4. Diseños de casos de prueba</p>		<p>Test Tema 9 (0,1 puntos)</p>	<p>Laboratorio #2: Pruebas funcionales con Selenium (3 puntos)</p>
Semana 12	<p>Tema 10. Factores de calidad del <i>software</i></p> <p>10.1. ¿Cómo estudiar este tema?</p> <p>10.2. Introducción a la calidad en Ingeniería <i>software</i></p> <p>10.3. Relación coste calidad</p> <p>10.4. Modelos de calidad de producto</p> <p>10.5. Modelo de calidad McCall</p> <p>10.6. Modelos de calidad estándar</p> <p>10.7. Gestión de la calidad del <i>software</i></p> <p>10.8. Referencias</p>			

	Contenido teórico	Trabajos (6 puntos)	Eventos (3 puntos)	Laboratorios (6 puntos)
Semana 13	<p>Tema 11. Introducción a las métricas en Ingeniería <i>software</i></p> <p>11.1. ¿Cómo estudiar este tema?</p> <p>11.2. Introducción a las métricas en Ingeniería <i>software</i></p> <p>11.3. Conceptos básicos: medidas, métricas e indicadores</p> <p>11.4. Tipos de métricas en Ingeniería <i>software</i></p> <p>11.5. El proceso de medición</p> <p>11.6. Métrica orientada a objetos</p> <p>11.7. Referencias</p>	<p>Trabajo: Cálculo de métricas (orientación a objetos) (2 puntos)</p>	<p>Test Tema 10 (0,1 puntos)</p>	
Semana 14	<p>Tema 12. Mantenimiento y reingeniería <i>software</i></p> <p>12.1. ¿Cómo estudiar este tema?</p> <p>12.2. Introducción al mantenimiento <i>software</i></p> <p>12.3. Proceso de mantenimiento</p> <p>12.4. Modelo de proceso de reingeniería de <i>software</i></p>		<p>Test Tema 11 (0,1 puntos)</p>	
Semana 17	<p>Tema 12. Mantenimiento y reingeniería <i>software</i> (continuación)</p> <p>12.5. Ingeniería inversa</p> <p>12.6. Reestructuración <i>software</i></p> <p>12.7. Referencias</p>		<p>Test Tema 12 (0,1 puntos)</p>	
Semana 16	Semana de exámenes			