



Guía Docente				
Datos Identificativos			2018/19	
Asignatura (*)	Internet e sistemas distribuídos	Código	614G01023	
Titulación				
Descritores				
Ciclo	Período	Curso	Tipo	Créditos
Grao	1º cuatrimestre	Terceiro	Obrigatoria	6
Idioma	Castelán			
Modalidade docente	Presencial			
Prerrequisitos				
Departamento	Computación			
Coordinación	Raposo Santiago, Juan	Correo electrónico	juan.raposo@udc.es	
Profesorado	Álvarez Díaz, Manuel	Correo electrónico	manuel.alvarez@udc.es	
	Bellas Permuy, Fernando		fernando.bellas@udc.es	
	Losada Perez, Jose		jose.losada@udc.es	
	Montoto Castela, Paula		paula.montoto@udc.es	
	Raposo Santiago, Juan		juan.raposo@udc.es	
Web				
Descrición xeral	Esta asignatura céntrase no desenvolvemento de aplicacións distribuídas. Utilízase un enfoque de deseño multicapa, no que en primeiro lugar se diseña unha capa de acceso de datos, posteriormente unha capa de lóxica de negocio (capa modelo) e, finalmente, unha capa que expón a funcionalidade da aplicación como un servizo que pode ser empregado por outras aplicacións. Explícanse tamén os principais estilos arquitectónicos para deseñar aplicacións distribuídas: RPC e REST. Para ilustrar todos estes conceptos xerais, empregaránse tecnoloxías utilizadas habitualmente na industria para estes propósitos, como JDBC, XML, servlets, HTTP e SOAP. O enfoque da asignatura é fundamentalmente práctico, polo que se proporcionan múltiples exemplos de código durante as clases teóricas, e a práctica ten un forte impacto na nota final da asignatura.			

Competencias / Resultados do título	
Código	Competencias / Resultados do título

Resultados da aprendizaxe			
Resultados de aprendizaxe	Competencias / Resultados do título		
Coñecemento e aplicación das características, funcionalidades e estrutura dos sistemas distribuídos, as redes de computadores e Internet, e deseñar e implementar aplicacións baseadas nelas.	A17	B1 B3	C2 C3 C4 C6

Contidos	
Temas	Subtemas
Tema 1. Introducción ás Aplicacións Distribuídas	O problema da integración de aplicacións Arquitecturas multicapa
Tema 2. Acceso a Datos	Acceso a base de datos: JDBC Acceso a datos doutras aplicacións
Tema 3. Capa Modelo	Conceptos xerais de deseño baseado en capas DAOs e Persistencia Deseño da API de Casos de Uso
Tema 4. Probas de Integración da Capa Modelo	Concepto de Probas de Integración Recordatorio da Implementación de Probas de Integración da Capa Modelo con Junit



Tema 5. Linguaxes de intercambio de datos: XML	Intercambio de datos entre aplicacións A linguaxe XML Introducción ós esquemas XML Parsing de documentos XML
Tema 6: Introducción o Deseño e Implementación de Servizos Web REST	Deseño por capas aplicado á integración de aplicacións Deseño de servizos REST Implementación de clientes REST: HTTP Implementación de servizos REST: Servlets
Tema 7: Deseño e Implementación de Servizos Web SOAP	O modelo RPC Deseño de servizos web SOAP Implementación de servizos web SOAP: JAX-WS
Tema 8: Frameworks para facer interfaces web	Concepto de Sistema de Plantillas Concepto de framework dirixido a componentes Concepto de aplicación AJAX

Planificación				
Metodoloxías / probas	Competencias / Resultados	Horas lectivas (presenciais e virtuais)	Horas traballo autónomo	Horas totais
Sesión maxistral	A17 B3 C6	30	22	52
Prácticas de laboratorio	A17 B1 B3 C2 C3	28	36	64
Proba de resposta múltiple	A17 B1 B3 C3 C6	2	5	7
Traballos tutelados	A17 B1 B3 C3 C4	2	20	22
Atención personalizada		5	0	5

*Os datos que aparecen na táboa de planificación son de carácter orientativo, considerando a heteroxeneidade do alumnado

Metodoloxías	
Metodoloxías	Descrición
Sesión maxistral	Clases impartidas polo profesor mediante a proxección de diapositivas. As clases teñen un enfoque práctico, explicando os conceptos teóricos mediante o uso de exemplos sinxelos e casos de estudo. As diapositivas e o código fonte dos exemplos e os casos de estudo están disponibles na páxina Web da asignatura en Campus Virtual
Prácticas de laboratorio	Ó longo do cuatrimestre realízase unha práctica (en Java), en grupo, na que o alumno utiliza a maior parte dos coñecementos teóricos de xeito integrado. A práctica realízase en dúas iteracións, sendo obrigatoria a entrega das dúas. Na corrección da primeira iteración, o profesor intenta garantir que o alumno enfoca ben a resolución da práctica. Para elo, o profesor tenta detectar erros importantes, e nese caso, orienta ó alumno cara a súa resolución. Na segunda iteración o alumno corrixe os erros detectados na primeira e engade as funcionalidades restantes.
Proba de resposta múltiple	Farase un exame tipo test, co obxectivo de comprobar que o alumno asimilou os conceptos correctamente. O exame tipo test componse dun conxunto de preguntas con varias respostas posibles, das que só unha é correcta. As preguntas non contestadas non puntúan, e as contestadas erróneamente puntúan negativamente.
Traballos tutelados	Propóñense traballos tutelados consistentes en desenvolver algunha extensión da práctica. Requerirán que o alumno, de maneira autónoma, deba estudar tecnoloxías ou solucións de deseño adicionais e aplicarlas á práctica.

Atención personalizada	
Metodoloxías	Descrición
Prácticas de laboratorio Traballos tutelados	Tutorías e consultas vía correo electrónico para dúbidas específicas. Presenza do profesor no laboratorio para contestar dúbidas da práctica.

Avaliación



Metodoloxías	Competencias / Resultados	Descrición	Cualificación
Prácticas de laboratorio	A17 B1 B3 C2 C3	A práctica estrutúrase en 2 partes (chamadas iteracións). É obrigatorio presentar ambas iteracións. A corrección de cada iteración realízase mediante unha defensa dos membros do grupo có profesor. A corrección da primeira iteración non leva asociada unha nota, pero a súa entrega é obrigatoria, e debe obrigatoriamente cumprir cos requisitos establecidos no enunciado da práctica para a devandita iteración. O profesor pon a nota da práctica despois da corrección da segunda iteración. É preciso obter polo menos un 5 sobre 10 na práctica para aprobar a materia.	32
Proba de resposta múltiple	A17 B1 B3 C3 C6	O examen será tipo test e pode constar tanto de cuestións directas como de pequenos problemas de deseño que o alumno debe resolver cos conceptos explicados en teoría e que foron usados na parte práctica. É necesario obter polo menos un 4 no exame para aprobar a materia.	60
Traballos tutelados	A17 B1 B3 C3 C4	A realización dos traballos tutelados é opcional.	8
Outros			

Observacións avaliación

Para superar a materia é necesario obter como mínimo un 4 sobre 10 no exame teórico e un 5 sobre 10 na práctica de laboratorio, e ademais a media ponderada (60% teoría, 32% práctica e 8% traballos tutelados) ten que acadar o 5.

No caso de alumnos matriculados a tempo parcial flexibilizaranse as horas de defensa de prácticas para adaptarse ós horarios laborais dos alumnos, sempre dentro do prazo máximo de entrega fixado para todos os alumnos.

Fontes de información

Bibliografía básica	<ul style="list-style-type: none"> - Leonard Richardson, Sam Ruby (2007). RESTful Web Services. O'Reilly - Martin Kalin (2009). Java Web Services: Up and Running. O'Reilly - J. Crupi, D. Alur, D. Malks (2003). Core J2EE Patterns. Prentice-Hall - E. R. Harold, W. S. Means (2004). XML in a Nutshell: A Desktop Quick Reference, Third edition. O'Reilly - B. McLaughlin (2006). Java and XML, Third Edition. O'Reilly <p>Recursos adicionais: Tutorial JDBC: http://docs.oracle.com/javase/tutorial/jdbc/index.html. Tutorial Servlets: http://docs.oracle.com/javaee/6/tutorial/doc/bnafd.html. Recursos adicionais: Tutorial JDBC: http://docs.oracle.com/javase/tutorial/jdbc/index.html. Tutorial Servlets: http://docs.oracle.com/javaee/6/tutorial/doc/bnafd.html.</p>
Bibliografía complementaria	<p>E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994. K. Arnold, J. Gosling, D. Holmes, The Java Programming Language, 4th edition, Addison-Wesley, 2005. G. Booch, I. Jacobson, J. Rumbaugh, Unified Modeling Language User Guide, 2nd edition, Addison-Wesley, 2005. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994. K. Arnold, J. Gosling, D. Holmes, The Java Programming Language, 4th edition, Addison-Wesley, 2005. G. Booch, I. Jacobson, J. Rumbaugh, Unified Modeling Language User Guide, 2nd edition, Addison-Wesley, 2005.</p>

Recomendacións

Materias que se recomenda ter cursado previamente

Bases de Datos/614G01013
 Deseño Software/614G01015
 Redes/614G01017

Materias que se recomenda cursar simultaneamente

Materias que continúan o temario



Programación Avanzada/614G01030

Integración de Aplicacións/614G01080

Observacións

Para facilitar que os alumnos con matrícula a tempo parcial poidan cursar a materia, tódolos materiais teóricos e prácticos da asignatura estarán disponibles en liña na aplicación <https://campusvirtual.udc.es>

(*A Guía docente é o documento onde se visualiza a proposta académica da UDC. Este documento é público e non se pode modificar, salvo casos excepcionais baixo a revisión do órgano competente dacordo coa normativa vixente que establece o proceso de elaboración de guías