



DATOS IDENTIFICATIVOS

Diseño de arquitecturas de grandes sistemas software

| | | | | |
|---------------------|---|------------|-------|--------------|
| Asignatura | Diseño de arquitecturas de grandes sistemas software | | | |
| Código | O06G150V01947 | | | |
| Titulación | Grado en Ingeniería Informática | | | |
| Descriptores | Creditos ECTS | Seleccione | Curso | Cuatrimestre |
| | 6 | OP | 4 | 1c |
| Lengua | Castellano | | | |
| Impartición | Gallego | | | |
| Departamento | Informática | | | |
| Coordinador/a | González Peña, Daniel | | | |
| Profesorado | González Peña, Daniel Ribadas Pena, Francisco José | | | |
| Correo-e | dgpena@uvigo.es | | | |
| Web | http://moovi.uvigo.gal | | | |
| Descripción general | Esta asignatura abarca todo el proceso de ingeniería de software pero centrándose en sistemas software de grandes dimensiones. En este tipo de sistemas las técnicas y herramientas habituales en ingeniería del software requieren un mayor grado de complejidad en la distribución de tareas y objetivos generales del sistema. Se comentan también las diversas aptitudes necesarias para enfocar el desarrollo de grandes sistemas de software desde un punto de vista orientado a componentes y con una perspectiva de producción industrial: las denominadas factorías de software. | | | |

Competencias

| | |
|--------|--|
| Código | |
| A2 | Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio. |
| A3 | Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética. |
| A4 | Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado. |
| A5 | Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía. |
| B1 | Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos en el ámbito de la ingeniería en informática que tengan por objeto, de acuerdo con los conocimientos adquiridos, la concepción, el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas. |
| B5 | Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos. |
| B9 | Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática. |
| C13 | Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema |
| C19 | Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web |
| C22 | Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software |
| C25 | Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software |

| | |
|-----|---|
| C27 | Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles |
| C28 | Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales |
| C30 | Capacidad para diseñar soluciones apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos |
| C32 | Capacidad para seleccionar, diseñar, desplegar, integrar, evaluar, construir, gestionar, explotar y mantener las tecnologías de hardware, software y redes, dentro de los parámetros de coste y calidad adecuados |
| D4 | Capacidad de análisis, síntesis y evaluación |
| D5 | Capacidad de organización y planificación |
| D6 | Capacidad de abstracción: capacidad de crear y utilizar modelos que reflejen situaciones reales |
| D7 | Capacidad de buscar, relacionar y estructurar información proveniente de diversas fuentes y de integrar ideas y conocimientos. |
| D8 | Capacidad de trabajar en situaciones de falta de información y/o bajo presión |
| D9 | Capacidad de integrarse rápidamente y trabajar eficientemente en equipos unidisciplinarios y de colaborar en un entorno multidisciplinar |
| D10 | Capacidad de relación interpersonal. |
| D11 | Razonamiento crítico |
| D14 | Tener motivación por la calidad y la mejora continua |

Resultados de aprendizaje

| Resultados previstos en la materia | Resultados de Formación y Aprendizaje | | | |
|---|---------------------------------------|----------------|--|---|
| RA1: Conocer y analizar la complejidad de los grandes sistemas software y abordar de forma efectiva cada una de las fases de su desarrollo | A2 A3 A4 A5 | B1 B5 B9 | C13 C19 C22 C25 C27 C28 C30 C32 | D4 D5 D6 D7 D8 D9 D10 D11 D14 |
| RA2: Distribuir el trabajo de cada uno de los equipos humanos encargados del desarrollo entre las diferentes partes del sistema | A2 A4 | B1 B5 B9 | C22 C30 | D9 |
| RA3: Ser capaz de dividir y estructurar todo gran sistema software en pequeñas piezas de software susceptibles de ser tratadas de forma independiente | A2 | B1 B5 | C13 C22 C25 C27 C32 | D4 D5 D6 D14 |
| RA4: Validar y verificar la integración de diversos componentes y arquitecturas software con el fin de crear grandes sistemas software | A2 A4 | B1 B5 | C22 C25 C27 C28 C32 | D4 D11 D14 |
| RA5: Orientar el proceso de desarrollo desde un punto de vista industrial | A2 | B1 B5 B9 | C13 C19 C22 C25 C27 C28 C30 C32 | D4 D5 D6 D10 D14 |
| RA6: Conocer las técnicas de ingeniería del software específicas para grandes sistemas software y grandes equipos de trabajo | A2 A3 A4 A5 | B1 B5 | C22 C25 C28 C30 | D4 D5 D7 D8 D11 D14 |

Contenidos

| Tema | |
|--|--|
| Análisis y diseño de grandes sistemas software | Recopilación de requisitos en grandes sistemas software. Diseños de arquitecturas de alto nivel de detalle. Análisis y diseño de software orientado a componentes (COTS). Análisis y diseño de piezas de software distribuido. |

| | |
|---|---|
| Tecnologías para la implementación de grandes sistemas software | Uso de middlewares de integración entre componentes y subsistemas. Aplicación de frameworks y metodologías específicas de software factories. |
| Pruebas de grandes sistemas software | Validación, pruebas y puesta en producción de grandes sistemas software. |

Planificación

| | Horas en clase | Horas fuera de clase | Horas totales |
|-------------------------------|----------------|----------------------|---------------|
| Lección magistral | 13 | 34 | 47 |
| Prácticas de laboratorio | 24 | 36 | 60 |
| Seminario | 10 | 0 | 10 |
| Presentación | 0.5 | 9.5 | 10 |
| Examen de preguntas objetivas | 3 | 0 | 3 |
| Proyecto | 2 | 18 | 20 |

*Los datos que aparecen en la tabla de planificación son de carácter orientativo, considerando la heterogeneidad de alumnado

Metodologías

| | Descripción |
|--------------------------|--|
| Lección magistral | Exposición de los contenidos teóricos de la materia. Con el fin de facilitar la comprensión de la misma y aumentar el interés del alumno, se incluirán diversos ejemplos y ejercicios en los que se puede requerir la participación activa del alumno. |
| Prácticas de laboratorio | Realización de problemas de carácter práctico que incluyen la programación de software relacionado con los contenidos de la materia. |
| Seminario | Resolución de dudas xerais e posta en común de problemas específicos de carácter teórico/práctico relacionados coa materia. |
| Presentación | Preparación y presentación en pequeños grupos de un tema, su exposición oral y planteamiento de aplicaciones. |

Atención personalizada

Pruebas Descripción

| | |
|----------|--|
| Proyecto | El profesor tutorizará al alumno en el laboratorio para la realización de los proyectos que se evaluarán al final de la materia, respondiendo dudas individualmente. |
|----------|--|

Evaluación

| | Descripción | Calificación | Resultados de Formación y Aprendizaje | | | |
|-------------------------------|--|--------------|---------------------------------------|----------------|--|---|
| Prácticas de laboratorio | Asistencia regular al laboratorio de prácticas y participación (planteamiento de dudas sobre el trabajo, etc.). RESULTADOS DE APRENDIZAJE EVALUADOS: RA1, RA2, RA3, RA4, RA5, RA6 | 5 | A4 A5 | B9 C30 | C13 D9 | D8 D14 |
| Presentación | Preparación y presentación en pequeños grupos de un tema, su exposición oral y planteamiento de aplicaciones. Se tendrá en cuenta la claridad de la exposición, la calidad de la presentación y el ajuste al tiempo máximo preestablecido. RESULTADOS DE APRENDIZAJE EVALUADOS: RA1, RA5, RA6 | 15 | A3 A4 | B1 B5 B9 | C22 C25 C27 C28 C30 | D4 D5 D6 D7 D8 D9 D10 D11 D14 |
| Examen de preguntas objetivas | Realización de diferentes pruebas tipo test a lo largo del curso que incluirán contenidos teóricos y prácticos de la materia. RESULTADOS DE APRENDIZAJE EVALUADOS: RA1, RA3, RA5, RA6 | 35 | A3 A5 | B5 B9 | C13 C19 C22 C25 C28 C32 | D6 |
| Proyecto | Realización de proyectos que integren los contenidos vistos en la materia. RESULTADOS DE APRENDIZAJE EVALUADOS: RA1, RA2, RA3, RA4, RA5, RA6 | 45 | A2 A3 A5 | B5 B9 | C19 C22 C25 C27 C28 C32 | D5 D6 D7 D11 |

Otros comentarios sobre la Evaluación

CRITERIOS DE EVALUACIÓN PARA ASISTENTES 1ª EDICIÓN DE ACTAS

No existe una asistencia mínima a clases para considerar un alumno como "asistente". Si un alumno desea ser evaluado expresamente como "no asistente", deberá indicarlo al profesor antes de la fecha del examen de la primera edición de actas.

Para superar la materia, la calificación final debe ser igual o superior a 5, debiéndose obtener como mínimo un 3,5 (sobre 10) en cada una de las pruebas (pruebas tipo test, proyectos y presentaciones/exposiciones).

La nota por asistencia a clase se calculará de forma proporcional a la asistencia real del alumno (no existiendo ningún porcentaje de faltas permitidas).

Ponderación

Calificación final = $0,05 * \text{nota por asistencia y participación} + 0,15 * \text{presentaciones/exposiciones} + 0,35 * \text{nota media de las pruebas tipo test} + 0,45 * \text{proyectos}$.

CRITERIOS DE EVALUACIÓN PARA NO ASISTENTES

Para superar la materia, la calificación final debe ser igual o superior a 5, debiéndose obtener como mínimo un 3,5 (sobre 10) en cada una de las pruebas (pruebas tipo test, proyectos y presentaciones/exposiciones).

Metodología/Prueba 1: Presentaciones/exposiciones.

- **Descripción:** Preparación y presentación en pequeños grupos de un tema, su exposición oral y planteamiento de aplicaciones. Se tendrá en cuenta la claridad de la exposición, la calidad de la presentación y el ajuste al tiempo máximo preestablecido.
- **% Calificación:** 15% (Para liberar esta parte, el estudiante deberá obtener una calificación igual o superior a 3,5 puntos sobre 10).
- **Competencias evaluadas:** CB3, CB4, CG1, CG5, CG9, CE22, CE25, CE27, CE28, CE30, CT4, CT5, CT6, CT7, CT8, CT9, CT10, CT11, CT14
- **Resultados de aprendizaje evaluados:** RA1, RA5, RA6.

Metodología/Prueba 2: Pruebas de tipo test.

- **Descripción:** Realización de diferentes pruebas tipo test a lo largo del curso que incluirán contenidos teóricos y prácticos de la materia.
- **% Calificación:** 40% (Para liberar esta parte, el estudiante deberá obtener una calificación igual o superior a 3,5 puntos sobre 10).
- **Competencias evaluadas:** CB3, CB5, CG5, CG9, CE13, CE19, CE22, CE25, CE28, CE32, CT6.
- **Resultados de aprendizaje evaluados:** RA1, RA3, RA5, RA6.

Metodología/Prueba 3: Proyecto.

- **Descripción:** Realización de proyectos que integren los contenidos vistos en la materia.
- **% Calificación:** 45% (Para liberar esta parte, el estudiante deberá obtener una calificación igual o superior a 3,5 puntos sobre 10).
- **Competencias evaluadas:** CB2, CB3, CB5, CG5, CG9, CE19, CE22, CE25, CE27, CE28, CE32, CT5, CT6, CT7, CT11.
- **Resultados de aprendizaje evaluados:** RA1, RA2, RA3, RA4, RA5, RA6.

Ponderación

Calificación final = $0,15 * \text{presentaciones/exposiciones} + 0,4 * \text{nota media de las pruebas tipo test} + 0,45 * \text{trabajos y proyectos}$.

CRITERIOS DE EVALUACIÓN PARA 2ª EDICIÓN DE ACTAS Y FIN DE CARRERA

Se empleará el mismo esquema de evaluación detallado en la sección "CRITERIOS DE EVALUACIÓN PARA NO ASISTENTES".

PROCESO DE CALIFICACIÓN DE ACTAS

Independientemente de la convocatoria y del tipo de asistencia, en el caso de no superar alguna parte de la evaluación, cuando la puntuación total ponderada fuese superior a 5 sobre 10, la calificación en actas será de 4,9.

FECHAS DE EVALUACIÓN

El calendario de pruebas de evaluación aprobado oficialmente por la Junta de Centro de la ESEI se encuentra publicado en la página web <http://www.esei.uvigo.es>.

Fuentes de información

Bibliografía Básica

Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, **Design Patterns: Elements of Reusable Object-Oriented Software**, 978-0201633610, 1, Addison-Wesley, 1995

Elisabeth Freeman (Author), Eric Freeman, Bert Bates, Kathy Sierra, Elisabeth Robson, **Head First Design Patterns**, 978-0596007126, 1, O'Reilly, 2004

OODesign.com. Object Oriented Design,

Antonio Goncalves, **Beginning Java EE 7**, 978-1430246268, 1, Apress, 2013

Eric Jendrock, Ricardo Cervera-Navarro, Ian Evans Devika, Gollapudi Kim Haase, William, Markito Chin, **The Java EE 6 Tutorial**, 978-0137081868, 2013

Bibliografía Complementaria

Jack Greenfield, Keith Short, Steve Cook, Stuart Kent, John Crupi, **Software Factories: Assembling Applications with Patterns, Models, Frameworks, and Tools**, 978-0471202844, 1, Wiley Publishing, 2004

Clemens Szyperski, **Component Software: Beyond Object-Oriented Programming**, 978-0201745726, 2, Addison-Wesley, 2002

Andy Ju An Wang, **Component-Oriented Programming**, 978-0-471-71369-2, 1, Wiley, 2005

Craig Walls, **Spring in Action**, 978-1617294945, 5, Manning, 2018

GoPivotal, Inc., **Spring Framework**,

Recomendaciones

Asignaturas que continúan el temario

Desarrollo ágil de aplicaciones/O06G150V01944

Asignaturas que se recomienda cursar simultáneamente

Desarrollo de aplicaciones para internet/O06G150V01962

Tecnologías y servicios web/O06G150V01970

Asignaturas que se recomienda haber cursado previamente

Ingeniería del software II/O06G150V01403

Plan de Contingencias

Descripción

=== MEDIDAS EXCEPCIONALES PLANIFICADAS ===

Ante la incierta e imprevisible evolución de la alerta sanitaria provocada por la COVID-19, la Universidad de Vigo establece una planificación extraordinaria que se activará en el momento en el que las administraciones y la propia institución lo determinen atendiendo a criterios de seguridad, salud y responsabilidad, y garantizando la docencia en un escenario no presencial o parcialmente presencial. Estas medidas ya planificadas garantizan, en el momento que sea preceptivo, el desarrollo de la docencia de un modo más ágil y eficaz al ser conocido de antemano (o con una amplia antelación) por el alumnado y el profesorado a través de la herramienta normalizada e institucionalizada de las guías docentes.

=== ESCENARIO 1: DOCENCIA MIXTA ===

En el caso de una situación excepcional en la cual no se pueda emplear el aforo completo de las aulas en las que se imparta la docencia se realizará una docencia mixta, en la que parte del alumnado podrá asistir presencialmente a las clases, mientras que otra parte del alumnado podrá seguir las clases de forma online a través del Campus Remoto.

En tal situación, se mantendrán las metodologías y sistemas de evaluación. Las evaluaciones se tratarán de hacer de forma presencial (p.ex. solicitando aulas con maior aforo). En el caso de no ser posible, se realizarán a través del Campus Remoto, Faticit y/o otros servicios de la Universidad de Vigo. En tal caso, se comunicará al alumnado con suficiente antelación.

Respecto a las tutorías, éstas se harán, preferentemente, de forma online. Con el fin de poder hacer una mejor organización, los alumnos deberán comunicar al profesorado su deseo de realizar una tutoría de forma previa a través de un correo electrónico.

=== ESCENARIO 2: DOCENCIA NO PRESENCIAL ===

En el caso de una situación excepcional en la cual no se pueda impartir docencia presencial, se impartirán las clases de forma online a través del Campus Remoto.

En tal situación, se mantendrán las metodologías y sistemas de evaluación. Las evaluaciones se realizarán a través de Campus Remoto, Fatic y/o otros servicios de la Universidad de Vigo. Estos cambios se comunicarán al alumnado con suficiente antelación.

Respecto das titorías, éstas se harán de forma online y, con el fin de poder hacer una mejor organización, los alumnos deberán comunicar al profesorado su deseo de realizar una tutoría de forma previa a través de un correo electrónico.

En casos excepcionales en los que un alumno justifique la existencia de una situación que le impida seguir la materia de forma normal (p.ej. problemas de conectividad, problemas de conciliación, etc.), podrá acordar con el profesorado la adaptación de las fechas de las pruebas de evaluación, así como los medios para realizarlas. En cualquier caso, se mantienen los sistemas de evaluación.
