



| Guía docente          |   |                    |                                    |          |
|-----------------------|---|--------------------|------------------------------------|----------|
| Datos Identificativos |   |                    |                                    | 2015/16  |
| Asignatura (*)        | Validación y Verificación del Software  | Código             | 614G01053                          |          |
| Titulación            | Grao en Enxeñaría Informática   |                    |                                    |          |
| Descritores           |   |                    |                                    |          |
| Ciclo                 | Periodo   | Curso              | Tipo                               | Créditos |
| Grado                 | 1º cuatrimestre   | Cuarto             | Obligatoria                        | 6        |
| Idioma                | Castellano  |                    |                                    |          |
| Modalidad docente     | Presencial  |                    |                                    |          |
| Prerrequisitos        |   |                    |                                    |          |
| Departamento          | Computación   |                    |                                    |          |
| Coordinador/a         | Castro Souto, Laura Milagros  | Correo electrónico | laura.milagros.castro.souto@udc.es |          |
| Profesorado           | Cabalar Fernandez, Jose Pedro   | Correo electrónico | pedro.cabalar@udc.es               |          |
|                       | Castro Souto, Laura Milagros  |                    | laura.milagros.castro.souto@udc.es |          |
|                       | Perez Vega, Gilberto  |                    | gilberto.pvega@udc.es              |          |
| Web                   | moodle.udc.es   |                    |                                    |          |
| Descripción general   | <p>Esta asignatura busca dominar las alternativas actuales de la ingeniería de software para la validación y verificación del software, mediante:</p> <ul style="list-style-type: none"> <li>- el conocimiento de técnicas y herramientas de validación de software funcionales y no funcionales a todos los niveles (unidad, integración, sistema);</li> <li>- el conocimiento de técnicas y herramientas de razonamiento automático; y</li> <li>- el conocimiento de técnicas y herramientas de verificación formal.</li> </ul> |                    |                                    |          |

| Competencias del título |   |
|-------------------------|---|
| Código                  | Competencias del título   |
| A28                     | 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. |
| B1                      | Capacidad de resolución de problemas  |
| B3                      | Capacidad de análisis y síntesis  |
| C2                      | Dominar la expresión y la comprensión de forma oral y escrita de un idioma extranjero.  |
| C3                      | Utilizar las herramientas básicas de las tecnologías de la información y las comunicaciones (TIC) necesarias para el ejercicio de su profesión y para el aprendizaje a lo largo de su vida.                           |
| C6                      | Valorar críticamente el conocimiento, la tecnología y la información disponible para resolver los problemas con los que deben enfrentarse.  |
| C7                      | Asumir como profesional y ciudadano la importancia del aprendizaje a lo largo de la vida.   |
| C8                      | Valorar la importancia que tiene la investigación, la innovación y el desarrollo tecnológico en el avance socioeconómico y cultural de la sociedad.   |

| Resultados de aprendizaje   |                         |          |                            |
|---|-------------------------|----------|----------------------------|
| Resultados de aprendizaje   | Competencias del título |          |                            |
| Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento apropiado de las teorías, modelos y técnicas actuales. | A28                     | B1<br>B3 | C2<br>C3<br>C6<br>C7<br>C8 |

| Contenidos |         |
|------------|---------|
| Tema       | Subtema |
|            |         |



|   |   |
|---|---|
| Bloque I: Validación de software                      | <p>I.1 Especificación, diseño y ejecución de pruebas</p> <p>I.1.1. Niveles y tipos de pruebas</p> <p>I.1.2. Propiedades y trazabilidad de requisitos</p> <p>I.2 Gestión de pruebas: planificación, evaluación, métricas y revisiones</p>  |
| Bloque II: Métodos formales y razonamiento automático | <p>II.1 Introducción: Deducción natural y cálculo de secuentes</p> <p>II.2 Prueba automática utilizando PVS</p> <p>II.3 ¿Qué es y para que se utiliza un probador de teoremas?</p> <p>II.4 Lenguaje de especificación de PVS: Tipos, expresiones, teorías, subtipado</p> <p>II.5 Probador PVS: tácticas, recursión, razonamiento ecuacional</p> |
| Bloque III: Comprobación por modelos (model checking) | <p>III.1 Introducción a las lógicas modales temporales</p> <p>III.2 Especificación de propiedades: deadlocks, safety, liveness, fairness</p> <p>III.3 Funcionamiento de un comprobador por modelos</p> <p>III.4 Introducción al manejo de una herramienta de model checking</p>   |

| Planificación            |                    |                    |  |               |
|--------------------------|--------------------|--------------------|--|---------------|
| Metodologías / pruebas   | Competencias       | Horas presenciales | Horas no presenciales / trabajo autónomo | Horas totales |
| Sesión magistral         | B3 C2 C7 C8        | 21                 | 26.25                                    | 47.25         |
| Prácticas de laboratorio | A28 B1 B3 C2 C3 C6 | 14                 | 35                                       | 49            |
| Trabajos tutelados       | A28 B1 B3 C2 C3 C6 | 7                  | 7  | 14            |
| Prueba objetiva          | B1 B3 C6           | 3                  | 31.5                                     | 34.5          |
| Atención personalizada   |                    | 5.25               | 0  | 5.25          |

(\*) Los datos que aparecen en la tabla de planificación són de carácter orientativo, considerando la heterogeneidad de los alumnos

| Metodologías             |   |
|--------------------------|---|
| Metodologías             | Descripción   |
| Sesión magistral         | Clases de teoría donde se imparten los contenidos del temario.                                      |
| Prácticas de laboratorio | Elaboración de trabajos prácticos en el laboratorio.  |
| Trabajos tutelados       | Resolución de trabajos tutelados planteados y resueltos en horario de tutorías de grupos reducidos. |
| Prueba objetiva          | Examen escrito.   |

| Atención personalizada  |  |
|---|--|
| Metodologías  | Descripción  |
| Prueba objetiva<br>Trabajos tutelados<br>Sesión magistral<br>Prácticas de laboratorio | Resolución de dudas de teoría y/o prácticas, trabajos tutelados, etc. en horario de tutorías de cada profesor/a. |

| Evaluación               |                    |   |              |
|--------------------------|--------------------|---|--------------|
| Metodologías             | Competencias       | Descripción   | Calificación |
| Prueba objetiva          | B1 B3 C6           | Examen escrito que computa un máximo de 4 puntos sobre la nota final. Se requiere un mínimo de 2 puntos para superar la asignatura.   | 40           |
| Trabajos tutelados       | A28 B1 B3 C2 C3 C6 | Resolución y participación en trabajos tutelados en horario de tutorías de grupos reducidos. Computa un máximo de 2 puntos de la nota final. Su realización no es obligatoria para superar la asignatura. | 20           |
| Prácticas de laboratorio | A28 B1 B3 C2 C3 C6 | Entrega e defensa de trabajos prácticos de laboratorio. Computa hasta un máximo de 4 puntos de la nota final. Su realización no es obligatoria para superar la asignatura.                                | 40           |



## Observaciones evaluación

En caso de no alcanzar el mínimo en la prueba objetiva, la nota final será la obtenida en la prueba objetiva.

En la convocatoria de segunda oportunidad, la prueba objetiva irá acompañada de una actividad de evaluación de las prácticas de laboratorio.

De acuerdo con la normativa de la UDC en relación al estudiantado matriculado a tempo parcial, el régimen de asistencia a clase no afectará negativamente al proceso de evaluación, admitiéndose en esta asignatura la dispensa académica para la asistencia, solicitada por las vías institucionales habilitadas al efecto. Sin embargo, esta flexibilidad asistencial no eximirá de la entrega de trabajos tutelados y prácticas de laboratorio en los mismos plazos fijados para el estudiantado a tiempo completo.

## Fuentes de información

|               |  |
|---------------|--|
| <b>Básica</b> | <ul style="list-style-type: none"><li>- Mordechai Ben-Ari (2012). Mathematical Logic for Computer Science. Springer</li><li>- Ron Patton (2001). Software testing. Sams</li><li>- Peter Farrell-Vinay (2008). Manage software testing. Auerbach</li><li>- Kent Beck (2002). Test Driven Development (By Example). Addison-Wesley</li><li>- Gerard J. Holzmann (2003). The SPIN model checker: primer and reference manual. Addison-Wesley</li><li>- Mordechai Ben-Ari (2001). Mathematical Logic for Computer Science. Springer</li><li>- Zohar Manna and Amir Pnueli (1991). The Temporal Logic of Reactive and Concurrent Systems. Specification. Springer</li><li>- Zohar Manna and Amir Pnueli (1995). The Temporal Logic of Reactive and Concurrent Systems. Safety. Springer</li></ul> |
|---------------|--|

## Complementaria

## Recomendaciones

### Asignaturas que se recomienda haber cursado previamente

Diseño Software/614G01015

Concurrencia y Paralelismo/614G01018

Proceso Software/614G01019

Arquitectura del Software/614G01221

Ingeniería de Requisitos/614G01222

Aseguramiento de la Calidad/614G01223

### Asignaturas que se recomienda cursar simultáneamente

Representación del Conocimiento y Razonamiento Automático/614G01036

Teoría de la computación/614G01039

Metodologías de Desarrollo/614G01051

### Asignaturas que continúan el temario

Proyectos de Desarrollo Software/614G01226

### Otros comentarios

(\*) La Guía Docente es el documento donde se visualiza la propuesta académica de la UDC. Este documento es público y no se puede modificar, salvo cosas excepcionales bajo la revisión del órgano competente de acuerdo a la normativa vigente que establece el proceso de elaboración de guías