



| Guía Docente | | | | |
|-----------------------|---|--------------------|--|----------|
| Datos Identificativos | | | | 2016/17 |
| Asignatura (*) | Deseño Software | Código | 614G01015 | |
| Titulación | Grao en Enxeñaría Informática | | | |
| Descritores | | | | |
| Ciclo | Período | Curso | Tipo | Créditos |
| Grao | 1º cuatrimestre | Segundo | Obrigatoria | 6 |
| Idioma | CastelánInglés | | | |
| Modalidade docente | Presencial | | | |
| Prerrequisitos | | | | |
| Departamento | Computación | | | |
| Coordinación | Mosqueira Rey, Eduardo | Correo electrónico | eduardo.mosqueira@udc.es | |
| Profesorado | Alonso Ríos, David Cabreiro Canosa, Mariano Javier Fontenla Romero, Oscar Hernandez Pereira, Elena Maria Monroy Camafreita, Juan Mosqueira Rey, Eduardo Pérez Sánchez, Beatriz Sanchez Maroño, Noelia | Correo electrónico | david.alonso@udc.es mariano.cabrero@udc.es oscar.fontenla@udc.es elena.hernandez@udc.es juan.monroy@udc.es eduardo.mosqueira@udc.es beatriz.perezs@udc.es noelia.sanchez@udc.es | |
| Web | | | | |
| Descrición xeral | <p>O Deseño Software é unha fase chave dentro do ciclo de vida do software que establece a ligazón entre os requisitos dun sistema e a súa posterior implementación. O deseño máis habitual hoxe en día é o deseño baseado na orientación a obxectos, que consiste en desenvolver un programa en base a obxectos que intercambian mensaxes.</p> <p>Esta materia introducirá aos alumnos nos elementos e propiedades básicas da orientación a obxectos usando unha linguaxe orientada a obxectos como Java. Aprenderase tamén a como reflectir os artefactos propios do deseño nunha linguaxe de modelado coma a Linguaxe Unificada de Modelado (UML).</p> <p>Finalmente presentaranse aqueles principios básicos que representan un bo deseño e aprenderase a identificar aqueles problemas típicos de deseño e as súas solucións máis comúns representadas como patróns de deseño.</p> | | | |

| Competencias / Resultados do título | |
|-------------------------------------|---|
| Código | Competencias / Resultados do título |
| A7 | Capacidade para deseñar, desenvolver, seleccionar e avaliar aplicacións e sistemas informáticos que aseguren a súa fiabilidade, seguranza e calidade, conforme a principios éticos e á lexislación e normativa vixente. |
| A13 | Coñecemento, deseño e utilización de forma eficiente dos tipos e estruturas de datos máis adecuados á resolución dun problema. |
| A14 | Capacidade para analizar, deseñar, construír e manter aplicacións de forma robusta, segura e eficiente, elixindo o paradigma e as linguaxes de programación máis adecuados. |
| B1 | Capacidade de resolución de problemas |
| B2 | Traballo en equipo |
| B3 | Capacidade de análise e síntese |
| B4 | Capacidade para organizar e planificar |
| C3 | Utilizar as ferramentas básicas das tecnoloxías da información e as comunicacións (TIC) necesarias para o exercicio da súa profesión e para a aprendizaxe ao longo da súa vida. |
| C6 | Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben afrontarse. |

| Resultados da aprendizaxe | |
|---------------------------|-------------------------------------|
| Resultados de aprendizaxe | Competencias / Resultados do título |
| | |



| | | | |
|---|------------------|----------------------|----------|
| Identificar o deseño software como unha das fases do ciclo de vida do software | A7 A13 A14 | B3 B4 | C3 |
| Coñecer os principios e propiedades básicas da orientación a obxectos | A7 A13 A14 | B1 B2 B3 B4 | C3 C6 |
| Plasmar un deseño software utilizando os artefactos propios dunha linguaxe de modelado coma UML | A7 A13 A14 | B1 B2 B3 B4 | C3 C6 |
| Coñecer os principios básicos que representan un bo deseño software | A7 A13 A14 | B1 B2 B3 B4 | C3 C6 |
| Identificar problemas típicos de deseño e as súas solucións máis comúns | A7 A13 A14 | B1 B2 B3 B4 | C3 C6 |
| Usar un deseño como guía para a implementación do software | A7 A13 A14 | B1 B2 B3 B4 | C3 C6 |

| Contidos | |
|--|---|
| Temas | Subtemas |
| 1. Introducción | ? Deseño software ? Deseño orientado a obxectos |
| 2. Elementos Básicos da Orientación a Obxectos | ? Clases e obxectos ? Identidade de obxectos ? Estado de obxectos ? Comportamento de obxectos |
| 3. Características Básicas da Orientación a Obxectos | ? Abstracción e encapsulamento ? Modularidade ? Xerarquía ? Polimorfismo ? Tipificación ? Ligadura Dinámica |
| 4. Linguaxe Unificada de Modelado (UML) | ? Introducción ? Bloques básicos do UML ? Deseño estático: Diagramas de clases ? Deseño dinámico: Diagramas de interacción ? Outros diagramas |
| 5. Principios de Deseño | ? Calidade no deseño ? Principios SOLID ? Tipos de herdanza |



| | |
|----------------------|---|
| 6. Padróns de Deseño | ? Introducción aos patróns de deseño ? Padróns elementais ? Deseños adaptables aos cambios ? Padróns e coleccións de obxectos ? Deseños débilmente acoplados ? Outros patróns e principios |
|----------------------|---|

| Planificación | | | | |
|--------------------------|---------------------------------|---|-------------------------|--------------|
| Metodoloxías / probas | Competencias / Resultados | Horas lectivas (presenciais e virtuais) | Horas traballo autónomo | Horas totais |
| Sesión maxistral | A7 A13 A14 B1 B3 C6 | 30 | 45 | 75 |
| Prácticas de laboratorio | A7 A13 A14 B1 B2 B3 B4 C3 C6 | 20 | 30 | 50 |
| Seminario | A7 A13 A14 B1 B2 B3 B4 C3 C6 | 10 | 10 | 20 |
| Proba obxectiva | A7 A13 A14 B1 B3 C6 | 3 | 0 | 3 |
| Atención personalizada | | 2 | 0 | 2 |

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

| Metodoloxías | |
|--------------------------|---|
| Metodoloxías | Descrición |
| Sesión maxistral | Clases expositivas de presentación dos coñecementos teóricos empregando diferentes recursos: pizarra, proxección de material en formato electrónico, apuntes en formato electrónico e os recursos facilitados pola equipa docente da asignatura no Campus Virtual da UDC. |
| Prácticas de laboratorio | Prácticas deseñadas pola equipa docente da asignatura baseadas nos coñecementos que cada estudante vai adquirindo. Estes traballos serán desenvolvidos preferiblemente en grupo (que serán como máximo de dúas persoas). Empregarase unha ferramenta de modelado para construír os artefactos de deseño e aplicarase unha linguaxe orientada a obxectos (Java) para realizar a implementación dos mesmos. |
| Seminario | Seminarios nos que se proporán actividades relacionadas cos coñecementos adquiridos en teoría ou práctica |
| Proba obxectiva | Proba escrita mediante a que se valoran os coñecementos adquiridos polo estudantado. Cada estudante deberá aplicar tanto os seus coñecementos tanto a nivel teórico coma a nivel práctico. |

| Atención personalizada | |
|---------------------------------------|--|
| Metodoloxías | Descrición |
| Prácticas de laboratorio Seminario | A atención personalizada ao estudantado comprende non só as titorías, presenciais ou virtuais, para a discusión de dúbidas, senón tamén as seguintes actuacións: <ul style="list-style-type: none"> - Seguemento do labor realizado nas prácticas de laboratorio propostas pola equipa docente. - Avaliación dos resultados obtidos nas prácticas, participación en seminarios realizados por cada estudante. - Encontros personalizados para resolver dúbidas sobre os contidos da asignatura. |

| Avaliación | | | |
|--------------|---------------------------|------------|---------------|
| Metodoloxías | Competencias / Resultados | Descrición | Cualificación |
| | | | |



| | | | |
|--------------------------|---------------------------------|--|----|
| Prácticas de laboratorio | A7 A13 A14 B1 B2 B3 B4 C3 C6 | Dous boletíns de exercicios baseados na programación en Java, na orientación a obxectos e o deseño de probas. Unha práctica de deseño centrada no uso de principios e patróns de deseño. Prácticas copiadas poderán supor un cero na nota, tanto para o orixinal como para a copia | 40 |
| Seminario | A7 A13 A14 B1 B2 B3 B4 C3 C6 | Os seminarios son prácticas de laboratorio desenroladas polos alumnos coa axuda directa do profesor que, ao final dos mesmos, mostrará cal era a solución a que os alumnos tiñan que chegar. Os seminarios están directamente relacionados coa teoría e a práctica e a súa avaliación farase nestos apartados | 0 |
| Proba obxectiva | A7 A13 A14 B1 B3 C6 | Proba escrita realizada ao final do curso sobre contidos teórico-prácticos. A proba obxectiva é obrigatoria para aprobar a materia e tamén é obrigatoria obter unha nota mínima de 4 sobre 10 para poder facer media cos outros elementos avaliados. | 60 |

Observacións avaliación

En caso de non chegar á nota mínima na proba obxectiva, en calquera das oportunidades, implicará que non se poida obter máis dun 4,5 na nota final da materia.

Aspectos a ter en conta para a avaliación de segunda oportunidade (Xullo):

A nota das prácticas de laboratorio serán as obtidas na primeira oportunidade (non haberá entrega de prácticas na segunda oportunidade). Aspecto a ter en conta en caso de matrícula a tempo parcial:

Elimínase a obrigatoriedade de asistir as actividades nas que se poida esixir presencialidade.

Fontes de información

| | |
|------------------------------------|---|
| Bibliografía básica | <ul style="list-style-type: none"> - Sierra, K., Bates, B. (2005). Head First Java. O'Reilly - Eckel, B. (2007). Piensa en Java (4ª ed.). Thinking in Java (4th ed.). Prentice-Hall - Booch J.; Rumbaugh J. y Jacobson I. (2006). El Lenguaje Unificado de Modelado (2ª ed.) The Unified Modeling Language (2nd ed.). Addison Wesley - Martin, R.C. (2004). UML para programadores Java. UML for Java Programmers. Pearson - Gamma, E.; Helm, R.; Johnson, R. y Vlissides J. (1996). Design Patterns: Elements of Reusable Object-oriented Software.. Addison Wesley |
| Bibliografía complementaria | <ul style="list-style-type: none"> - Arnold K., Gosling J. y Holmes D. (2005). The Java Programming Language. Prentice-Hall - Rumbaugh, J.; Jacobson, I. y Booch, J. (2004). The Unified Modeling Language Reference Manual. Addison Wesley - Stevens, P. y Pooley, R. (2006). Using UML. Software Engineering with Objects and Components. Addison Wesley - Freeman, E., Freeman, E., Bates, B. (2004). Head First Design Patterns. O'Reilly - Grand M. (2002). Patterns in Java. John Wiley & Sons |

Recomendacións

Materias que se recomenda ter cursado previamente

Programación I/614G01001

Programación II/614G01006

Materias que se recomenda cursar simultaneamente

Paradigmas de Programación/614G01014

Materias que continúan o temario



Proceso Software/614G01019

Interfaces Home Máquina/614G01022

Internet e sistemas distribuidos/614G01023

Observacións

A materia asume que os estudantes saben programar e coñecen as estruturas de datos (Programación II) aínda que nunca utilizaron unha linguaxe orientada a obxectos. Ao principio do curso, e segundo vanse introducindo os conceptos propios da orientación a obxectos, os alumnos familiarízanse cos aspectos básicos da linguaxe de programación Java.

(*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