

DISEÑO DE APLICACIONES ORIENTADAS A OBJETOS

Curso 2016/2017

(Código: 71022011)

1. PRESENTACIÓN DE LA ASIGNATURA

El objetivo de esta guía es orientar al alumno en el estudio de la asignatura. Se recomienda la lectura completa de la guía a comienzo del cuatrimestre para tener una idea completa de la temática de la asignatura y el calendario de prácticas, de forma que el alumno pueda planificar su trabajo para cumplir con las fechas de entrega.

La asignatura de Diseño de Aplicaciones Orientada a Objetos (DAOO en adelante) se imparte en el segundo cuatrimestre del segundo curso, consta de 6 créditos, con carácter de formación avanzada, para la titulación de grado en Informática.

2. CONTEXTUALIZACIÓN EN EL PLAN DE ESTUDIOS

Esta asignatura es una introducción al Diseño de Aplicaciones Orientadas a Objetos (DAOO) y tiene como objetivo principal el estudio y la puesta en práctica de los patrones de diseño como abstracción para la realización de aplicaciones y sistemas complejos basados en la metodología de orientación a objetos. Dado que el alumno ha tomado contacto con esta metodología en la asignatura de Programación Orientada a Objetos (POO), en este curso se profundizará en su uso y aplicación en escenarios más complejos. Así mismo, y dado que en POO el alumno ha desarrollado programas básicos utilizando el lenguaje de programación Java y la herramienta didáctica BlueJ, la asignatura DAOO mantendrá el uso de este lenguaje de programación motivando al alumno a implementar las actividades propuestas en el temario utilizando Eclipse como entorno de desarrollo estándar y mucho más extendido en la comunidad.

Respecto a la contextualización de las competencias especificadas en la memoria de verificación de ANECA se puede destacar las siguientes que corresponden a esta asignatura:

- (G.2) Competencias cognitivas superiores: selección y manejo adecuado de conocimientos, recursos y estrategias cognitivas de nivel superior apropiados para el afrontamiento y resolución de diversos tipos de tareas/problemas con distinto nivel de complejidad y novedad: Análisis y Síntesis. Aplicación de los conocimientos a la práctica Resolución de problemas en entornos nuevos o poco conocidos. Pensamiento creativo. Razonamiento crítico. Toma de decisiones.
- (G.4) Competencias de expresión y comunicación (a través de distintos medios y con distinto tipo de interlocutores): Comunicación y expresión escrita. Comunicación y expresión oral. Comunicación y expresión en otras lenguas (con especial énfasis en el inglés). Comunicación y expresión matemática, científica y tecnológica (cuando sea requerido y estableciendo los niveles oportunos).
- (FB.5) Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, así como de los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

3. REQUISITOS PREVIOS REQUERIDOS PARA CURSAR LA ASIGNATURA

Al tratarse de una continuación de asignatura POO, se requiere que el alumno sabe lo básico del diseño y desarrollo usando la orientación a objetos y como hacer programas con Java.



4.RESULTADOS DE APRENDIZAJE

Con el estudio de la asignatura el alumno debe adquirir una visión general de la DAOO. En concreto, se pretende que al finalizar el curso el alumno: a) Haya reforzado y ampliado sus conocimientos de programación orientada a objetos, b) Conozca los patrones de uso general de creación, estructura y comportamiento, c) Sea capaz de identificarlos en la etapa de diseño de una aplicación software, y; d) Conozca el modo en el que estos se implementan en el lenguaje Java.

Los resultados de aprendizaje que los alumnos van a conseguir en esta asignatura, según la memoria de verificación de ANECA son:

- R4. Realizar pruebas de validación y depuración de un programa dado.
- R5. Identificar clases de objetos con los datos de un problema.
- R6. Crear clases y objetos y manipularlos.
- R7. Identificar los métodos que caracterizan a una clase de objetos.
- R8. Utilizar el mecanismo de la herencia para crear clases.
- R9. Identificar las relaciones entre clases en distintos casos de uso.
- R10. Diseñar y documentar interfaces de usuario.
- R11. Usar Patrones de Diseño
- R12. Abordar metodológicamente proyectos de programación medios.
- R13. Dominar un lenguaje de programación acorde al paradigma del contenido de la materia.

5.CONTENIDOS DE LA ASIGNATURA

Los contenidos de este curso se organizan en dos partes:

Parte 1. El diseño orientado a objetos

Capítulo 1. Análisis y diseño basado en objetos

Capítulo 2. Identificación de clases. Escenarios. Documentación. Cooperación. Prototipos. Diseño de la persistencia

Capítulo 3. Diseño del interfaz de usuario. Componentes y gestores de disposición.

Capítulo 4: Programación orientada a eventos. Visores de imágenes

Capítulo 5: Crecimiento: desarrollo iterativo y reusabilidad

Parte 2. Los patrones de diseño

Capítulo 6. Patrones de creación.

- Abstract Factory (Fábrica Abstracta)
- Builder (Constructor)
- Factory Method (Método de Fabricación)
- Prototype (Prototipo)
- Singleton (Único)

Capítulo 7. Patrones estructurales.

- Adapter (Adaptador)
- Bridge (Puente)
- Composite (Compuesto)
- Decorator (Decorador)
- Facade (Fachada)
- Flyweight (Peso Ligero)
- Proxy (Apoderado)

Capítulo 8. Patrones de comportamiento.



- Chain of Responsibility (Cadena de Responsabilidad)
- Command (Orden)
- Interpreter (Intérprete)
- Iterator (Iterador)
- Mediator (Mediador)
- Memento (Recuerdo)
- Observer (Observador)
- State (Estado)
- Strategy (Estrategia)
- Template Method (Método Plantilla)
- Visitor (Visitante)

Capítulo 9. Casos de estudio.

6.EQUIPO DOCENTE

- [TIMOTHY MARTIN READ](#)
- [JUAN MANUEL CIGARRAN RECUERO](#)

7.METODOLOGÍA Y ACTIVIDADES DE APRENDIZAJE

La modalidad y tipo de actividades que se contemplan incluye: trabajo con contenidos teórico-prácticos utilizando la bibliografía y el material complementario. Trabajo autónomo con las actividades de ejercicios y pruebas de autoevaluación disponibles, y realización de una práctica bajo la supervisión del tutor, con las herramientas y directrices preparadas por el equipo docente. La interacción con el equipo docente y los tutores se describe más adelante.

Para solicitar plaza/turno de prácticas de laboratorio/experimentales, el estudiante tendrá que acceder a la aplicación de prácticas desde su escritorio. En estas imágenes puede ver desde dónde se puede realizar el acceso a dicha aplicación: Imágenes. Si al acceder a ella no encuentra ninguna oferta, deberá ponerse en contacto con el centro asociado donde está matriculado.

8.EVALUACIÓN

Evaluación continua

- La autoevaluación supone un aspecto muy importante dentro de un proceso general de aprendizaje. En esta asignatura, planteamos a los alumnos un proceso de autoevaluación basado en la realización de preguntas tipo test y en la resolución de problemas de examen de convocatorias anteriores. Para ello, en el módulo de contenidos dentro del entorno virtual CiberUNED los alumnos pueden encontrar dos apartados de "Autoevaluación" correspondientes a cada una de las Unidades Didácticas del curso. En dichos apartados el alumno podrá encontrar tanto cuestiones y problemas resueltos como sus soluciones, con las que podrá realizar una autoevaluación de sus conocimientos.
- Práctica obligatoria. Realización de una práctica obligatoria a lo largo del cuatrimestre, bajo la supervisión del tutor. La práctica aporta 10% de la nota final y requiere la presencialidad del alumno en el Centro Asociado para su realización (en una única sesión presencial) y corrección por parte del tutor. La nota mínima para aprobar la práctica es 5.0.

Nota para los alumnos de esta asignatura (DAOO) que no han cursado previamente la asignatura de Programación Orientada a Objetos (POO)

El propósito de esta asignatura es profundizar en la aplicación de POO en escenarios más complejos. Las asignaturas han sido diseñadas así con ese objetivo. Por este motivo, en la primera parte del DAOO se plantea un repaso de los contenidos teóricos de POO, incidiendo en los aspectos gráficos y de interfaz de usuario, con el fin de aplicar dichos conocimientos sobre una práctica con una cierta complejidad gráfica.

El equipo docente recomienda, evidentemente, que el alumno curse POO antes de DAOO. No obstante, los alumnos que no



pasen por POO, por el motivo que sea, pueden cursar DAOO, pero el equipo docente no proporcionará resúmenes de la parte teórica de POO ni tampoco una solución a la práctica de dicha asignatura. Aunque el enunciado de la práctica de DAOO sea similar al de la práctica de POO del año anterior, el alumno debe tener claro que el enfoque es totalmente diferente ya que, en esta asignatura, se pretenden aplicar los conocimientos de patrones de diseño para su rediseño completo.

El hecho de no haber realizado la practica de POO por cualquier razón en el curso pasado no supone una desventaja con respecto al resto de los alumnos por las siguientes razones: 1) El hecho de tener convalidada la asignatura de POO supone que el alumno dispone de las competencias necesarias como para abordar cualquiera de los proyectos planteados en dicha asignatura, 2) En el enunciado de la practica de DAOO se hace referencia a la practica de POO del año anterior para poner en contexto al alumno. De hecho, en todos sus apartados menos en el último se pide que el usuario trabaje sobre una implementación limpia de patrones, es decir, no integrada en el código de la práctica de POO del año pasado, 3) solo en el último apartado, que es voluntario, se pide integrar los resultados obtenidos en el código de la práctica de POO del año anterior. Respecto a este último punto decir que, en la mayor parte de las ocasiones, esto va a suponer que el alumno rehaga una buena parte de la práctica ya que la implementación realizada en la asignatura de POO no se adecuaba a una aproximación basada en patrones. En cualquier caso, cualquier alumno de DAOO puede aprobar la parte práctica sin necesidad de llevar a cabo el ultimo punto de integración.

La evaluación final

Se llevará a cabo a partir de las siguientes pruebas:

- Calificación de la práctica obligatoria, ya que es necesario aprobarla para la superación de la asignatura.
- Realización de un examen teórico/práctico que aporta 90% de la nota final. La nota mínima para aprobar el examen es 5.0.

El examen constará de dos partes, una teórica formada por preguntas tipo test y que será eliminatoria, y una segunda parte práctica formada por un problema de programación con varios apartados y en los que el alumno demostrará el nivel de los conocimientos adquiridos.

Se incluirán también preguntas sobre la práctica obligatoria.

Criterios de Calificación

Para que el examen de un alumno sea calificado deberá haber asistido, como mínimo, a dos sesiones presenciales de prácticas en su centro asociado y haber entregado y aprobado la práctica obligatoria.

Criterios de Evaluación

Para que un alumno pueda aprobar la asignatura deberá haber superado un mínimo de preguntas establecido en la parte teórica (tipo test) del examen. En la evaluación de la asignatura se tendrán en cuenta especialmente los aspectos relativos al diseño más que a los detalles propios de la implementación.

9. BIBLIOGRAFÍA BÁSICA

ISBN(13): 9788478290598
Título: PATRONES DE DISEÑO
Autor/es: Gamma, Erich ; Otros ;
Editorial: PEARSON ADDISON-WESLEY

Buscarlo en Editorial UNED

Buscarlo en librería virtual UNED

Buscarlo en bibliotecas UNED

Buscarlo en la Biblioteca de Educación



Comentarios y anexos:

Este libro no trata de una introducción a la tecnología orientada a objetos ni al diseño orientado a objetos. Es un libro de patrones de diseño que describe soluciones simples y elegantes para problemas específicos del diseño de software orientado a objetos.

En este libro, Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides introducen los principios de los patrones de diseño y ofrecen un catálogo de dichos patrones. Así, este libro realiza dos importantes contribuciones. En primer lugar, muestra el papel que los patrones pueden desempeñar diseñando la arquitectura de sistemas complejos. En segundo lugar, proporciona una referencia práctica de un conjunto de excelentes patrones que el desarrollador puede aplicar para construir sus propias aplicaciones.

10. BIBLIOGRAFÍA COMPLEMENTARIA

Comentarios y anexos:

Como material complementario, el equipo docente recomienda la consulta de los siguientes documentos:

- Patterns in Java: A Catalog of Reusable Design Patterns Illustrated with UML, 2nd Edition, Volume 1. Marc Grand. Wiley. 2002
- Pattern Hatching: Design Patterns Applied. John Vlissides. Addison-Wesley. 1998.
- Design Patterns Explained: A New Perspective on Object-Oriented Design. Alan Shalloway, James R. Trott. Addison-Wesley. 2001.
- Resumen de Luis Fernandez (<http://software.guisho.com/patrones-de-diseno>)

11. RECURSOS DE APOYO

Como materiales adicionales para el estudio de la asignatura se ofrece en el curso virtual:

- Esta guía de estudio y una versión extendida de la misma, la guía didáctica.
- Enunciados y soluciones de ejercicios teórico-prácticos que el alumno puede usar como ejercicios de autoevaluación.
- Exámenes resueltos de anteriores convocatorias.
- Lista de preguntas frecuentes, que recogen dudas de años anteriores.

Además, a través de CiberUNED se pondrán a disposición de los alumnos herramientas necesarias para el desarrollo de la práctica. Los alumnos pueden acceder al curso virtual en este entorno con su identificador y clave de acceso. En los centros asociados los alumnos dispondrán de ordenadores en donde el entorno de desarrollo BlueJ deberá estar instalado. Además, los alumnos que dispongan de un ordenador personal podrán instalarse dicho entorno de desarrollo. En el entorno CiberUNED se encontrará este paquete software y las instrucciones para su instalación. El entorno virtual se usará como medio para que los estudiantes puedan acceder a material complementario de estudio. El equipo docente añadirá aquellos documentos o referencias que considere adecuados para que los alumnos puedan complementar los conocimientos adquiridos a través del estudio de la bibliografía básica.

12. TUTORIZACIÓN

La enseñanza a distancia posee unas características que la diferencian de la presencial. Sin embargo, esto no impide que el alumno pueda disponer de la ayuda y los recursos necesarios para cursar las asignaturas en las que se matricule. Los mecanismos de los que dispone el alumno para facilitar el aprendizaje requerido en la asignatura son los siguientes:

- Tutores en los centros asociados. Los tutores serán los encargados de desarrollar las sesiones presenciales de



seguimiento y control de las prácticas. Para la realización de las prácticas el alumno debe ponerse en contacto con el tutor correspondiente, para conocer cuanto antes el horario y las sesiones previstas.

- Tutorías presenciales o virtuales en el centro asociado correspondiente.
- Entorno Virtual. A través de CiberUNED el equipo docente de la asignatura pondrá a disposición de los alumnos diverso material de apoyo en el estudio, así como el enunciado de la práctica obligatoria. Dispone además de foros donde los alumnos podrán plantear sus dudas para que sean respondidas por los tutores o por el propio equipo docente. Es el soporte fundamental de la asignatura, y supone la principal herramienta de comunicación entre el equipo docente, los tutores y los alumnos, así como de los alumnos entre sí.

13.Revisión de calificaciones

Se podrá solicitar revisión de las calificaciones en el plazo y forma establecidos por la UNED.

