

AD07.- Creación de componentes visuales.

Orientaciones para la tutoría

Esta es la última Unidad de Trabajo del módulo.

Es una unidad de finalización en la que se cierra el estudio de la teoría de acceso a datos con su inclusión en componentes.

En concreto, esta unidad se centra en determinar en qué casos puede ser útil crear un componente de acceso a datos, que características debe tener y el proceso de creación del componente en sí teniendo en cuenta todas sus propiedades, usando herramientas de diseño y con un ejemplo práctico al final de la unidad donde el alumnado podrá crear componentes con el Entorno de Desarrollo NetBeans.

Datos generales de la Unidad de Trabajo

Nombre completo del MP	Acceso a Datos.	Siglas MP	AD
Nº y título de la UT	07.- Creación de componentes visuales.	Horas	21/125
Índice o tabla de contenidos	<ul style="list-style-type: none">1.- Concepto de componente. Características.2.- Propiedades y atributos.3.- Editores de propiedades.4.- Eventos. Asociación de acciones a eventos.5.- Introspección. Reflexión.6.- Persistencia del componente.7.- Propiedades simples e indexadas.8.- Propiedades simples e indexas.9.- Herramientas para el desarrollo de componentes visuales.10.- Empaquetado de los componentes.11.- Elaboración de un componente de ejemplo.<ul style="list-style-type: none">11.1.- Modelo o base de datos.11.2.- Controlador o lógico del modelo.11.3.- Estructura del JavaBean.11.4.- Creación del componente.11.5.- Añadir propiedades.11.6.- Implementar el comportamiento.11.7.- Gestión de eventos.11.8.- Uso de componentes previamente elaborados en NetBeans.Anexo.- Licencias de recursos.		
Objetivos	<ul style="list-style-type: none">a. Seleccionar y emplear lenguajes, herramientas y librerías, interpretando las especificaciones para desarrollar aplicaciones multiplataforma con acceso a bases de datos.		

	<div>b. Gestionar la información almacenada, planificando e implementando sistemas de formularios e informes para desarrollar aplicaciones de gestión.</div> <div>c. Seleccionar y utilizar herramientas específicas, lenguajes y librerías, evaluando sus posibilidades y siguiendo un manual de estilo, para manipular e integrar en aplicaciones multiplataforma contenidos gráficos y componentes multimedia.</div> <div>d. Emplear herramientas de desarrollo, lenguajes y componentes visuales, siguiendo las especificaciones y verificando interactividad y usabilidad, para desarrollar interfaces gráficos de usuario en aplicaciones multiplataforma.</div> <div>e. Seleccionar y emplear técnicas y herramientas, evaluando la utilidad de los asistentes de instalación generados, para empaquetar aplicaciones.</div> <div>f. Establecer procedimientos, verificando su funcionalidad, para desplegar y distribuir aplicaciones.</div>
<div>Consejos y recomendaciones</div>	<div>En esta unidad el alumno o alumna puede encontrar dificultades debido a la complejidad de los conceptos que se utilizan.</div> <div>Si las tiene, deberá repasarse los contenidos y el ejemplo final de la unidad, sobre todo en cuanto a la creación del componente y las órdenes en Java para gestionar los eventos y definir propiedades.</div> <div>Respecto de la tarea, el principal inconveniente es la creación de múltiples proyectos y tener en cuenta la gestión de las bibliotecas jar.</div> <div>Los demás pasos son de realización automática y no deberían presentar ningún problema.</div>
<div>Tarea</div>	<div>Criterios de puntuación</div> <div><div>✓ Añadir tabla a la base de datos y rellenarla. 1 punto.</div><div>✓ Adición de las propiedades de una matrícula con sus correspondientes getters y setters. 1 punto.</div><div>✓ Añadir las estructuras internas para gestionar los datos. 1 punto.</div><div>✓ Implementar la diferencia entre el modo de listado normal y listado por DNI. 1 punto.</div><div>✓ Añadir los métodos para hacer listados (completos y por DNI). 1 punto.</div><div>✓ Crear la clase que hereda de EventObject para que se puedan crear los eventos a lanzar. 1 punto.</div><div>✓ Añadir el código necesario para generar los eventos cuando se cargan las matrículas. 2 puntos.</div><div>✓ Generar el ejemplo de prueba que realice un listado completo, otro por DNI y añadir un registro a la tabla. 1 punto.</div><div>✓ Añadir el código necesario para programar la captura de los eventos que se generan. 1 punto.</div></div> <div>Criterios de corrección</div> <div>Se aconseja ser estricto o estricta en la realización de la tarea en todos sus puntos, pues se trata de repetir lo estudiado en la unidad.</div> <div>Solución propuesta</div>

	La solución completa y correcta a la tarea propuesta, desarrollada por el autor o la autora de la misma, la podrás encontrar como enlace en la página de inicio de la unidad de trabajo, justo debajo del enlace a las orientaciones para la tutoría.
Temas de debate	Ventajas que aporta el uso de componentes en programación.