

DATOS IDENTIFICATIVOS DE LA UNIDAD FORMATIVA

UNIDAD FORMATIVA	IMPLEMENTACIÓN E INTEGRACIÓN DE ELEMENTOS SOFTWARE CON TECNOLOGÍAS BASADAS EN COMPONENTES	Duración	90
		Condicionada	
Código	UF1290		
Familia profesional	INFORMÁTICA Y COMUNICACIONES		
Área Profesional	Sistemas y telemática		
Certificado de profesionalidad	Programación de sistemas informáticos	Nivel	3
Módulo formativo	Desarrollo de software basado en tecnologías orientadas a componentes	Duración	210
Resto de unidades formativas que completan el módulo	Diseño de elementos software con tecnologías basadas en componentes	Duración	90
	Despliegue y puesta en funcionamiento de componentes software		30

Apartado A: REFERENTE DE COMPETENCIA

Esta unidad formativa se corresponde con la RP2 de la UC0965_3 DESARROLLAR ELEMENTOS SOFTWARE CON TECNOLOGÍAS DE PROGRAMACIÓN BASADA EN COMPONENTES.

Apartado B: ESPECIFICACIÓN DE LAS CAPACIDADES Y CONTENIDOS

Capacidades y criterios de evaluación

C1: Construir elementos software a partir de las especificaciones de necesidades y con las condiciones de desarrollo de la tecnología de componentes utilizada.

CE1.1 Explicar los enfoques de desarrollo e implementación para la creación de componentes aplicando el principio de reutilización.

CE1.2 Describir el proceso de adaptación de un componente existente para incluirlo en la arquitectura en la que se quiere reutilizar, según especificaciones técnicas de la tecnología de componentes utilizada.

CE1.3 Enunciar las características del proceso de diseño de un nuevo componente para incluirlo en la arquitectura en la que se quiere utilizar, garantizando su futura reutilización.

CE1.4 Clasificar las herramientas de programación y depuración para optimizar la fase de desarrollo de los componentes según unas especificaciones dadas.

CE1.5 Identificar los elementos y parámetros de la interfaz del componente, para su desarrollo con herramientas y lenguajes específicos, para implementar la vía de comunicaciones con el resto de componentes según los estándares de definición de interfaces de la arquitectura.

CE1.6 Realizar los procesos de instalación del componente, comprobando que ejecuta las acciones requeridas y su disponibilidad para las aplicaciones que lo invoquen, según especificaciones técnicas de la arquitectura.

CE1.7 En un caso práctico para desarrollar componentes software reutilizables, dadas unas especificaciones funcionales y técnicas:

- Diseñar el nuevo componente para que cumpla las especificaciones funcionales dadas.
- Comprobar que la funcionalidad del componente diseñado puede ser extendida para futuras reutilizaciones.
- Implementar el componente utilizando herramientas y lenguajes orientados a objeto.
- Depurar y probar el componente desarrollado utilizando las herramientas disponibles.
- Documentar el componente y sus interfaces para facilitar su futura reutilización.

CE1.8 En un caso práctico para desarrollar software reutilizando componentes, dadas unas especificaciones funcionales y técnicas:

- Utilizar repositorios de componentes para localizar aquellos que se ajusten a las especificaciones dadas y puedan ser reutilizados.
- Diseñar las modificaciones que se van a realizar sobre el componente existente para que cumpla las especificaciones dadas.
- Modificar el componente utilizando herramientas y lenguajes orientados a objeto.
- Depurar y probar el componente modificado utilizando las herramientas disponibles.
- Documentar las modificaciones realizadas sobre el componente y sus interfaces para facilitar su futura reutilización.

CE1.9 Interpretar la documentación técnica asociada a las herramientas de programación, incluso si está editada en la lengua extranjera de uso más frecuente en el sector, utilizándola de ayuda en el desarrollo.

Contenidos

1. Desarrollo de componentes

- o Lenguajes de desarrollo de componentes.
 - Comparativa con lenguajes orientados a objetos
 - Lenguajes orientados a componentes:

- Descripción de interfaces
- Ensamblado
- Descripción de arquitectura
- Requisitos principales del desarrollo orientado a componentes:
 - Modularidad
 - Despliegue independiente
 - Reemplazabilidad
 - Seguridad
 - Separación entre interfaz e implementación
- Infraestructuras (frameworks) de componentes:
 - Modelos de infraestructuras de componentes:
 - Orientados a conexión
 - Orientados a contexto
 - Orientados a aspectos
 - Descripción de las infraestructuras de componentes de uso común:
 - OMG: CORBA, OMA
 - Java: JavaBeans, EJBs
 - Microsoft: COM, OLE/ActiveX, .NET
- Métodos de desarrollo de componentes:
 - Uso de lenguajes orientados a objetos
 - Selección de infraestructuras de componentes
- Construcción de software mediante componentes:
 - Definición de interfaces. Lenguajes de descripción de interfaces
 - Reutilización de componentes.
 - Técnicas de ensamblado en infraestructuras de uso común
- Técnicas específicas de desarrollo:
 - Componentes en la capa de servidor web. Páginas dinámicas
 - Componentes en la capa de servidor de aplicaciones.
 - Componentes en la capa de aplicación cliente:
 - Componentes de interfaz gráfico
 - Componentes orientados a documento
 - Componentes en la capa de servicios web
 - Componentes para dispositivos móviles
- Herramientas para el desarrollo de componentes:
 - Entornos integrados de desarrollo de componentes
 - Configuración e instalación de herramientas de uso común:
 - Entorno Java
 - Entorno .NET
 - Gestión del ciclo de vida en el desarrollo de componentes mediante herramientas de uso común:
 - Uso de repositorios de componentes. Registro de componentes
 - Reutilización de componentes para la construcción de sistemas software
 - Definición de metadatos de componente. Descriptores de interfaces
 - Modelo de seguridad
 - Instalación de componentes
 - Depuración y prueba de componentes

2. Componentes distribuidos

- Programación distribuida en infraestructuras de uso común:
 - Programación multihilo (multithreading)
 - Comunicaciones síncronas y asíncronas
- Modelos de intercambio:
 - Llamadas a procedimientos remotos
 - Orientados a mensajes
 - Orientados a recursos

Apartado C: REQUISITOS Y CONDICIONES

Deberá cumplir alguno de los requisitos siguientes:

- Estar en posesión del título de Bachiller
- Estar en posesión de algún certificado de profesionalidad de nivel 3.

- Estar en posesión de un certificado de profesionalidad de nivel 2 de la misma familia y área profesional
- Cumplir el requisito académico de acceso a los ciclos formativos de grado superior o haber superado las correspondientes pruebas de acceso a ciclos de grado superior
- Tener superada la prueba de acceso a la universidad para mayores de 25 años y/o de 45 años
- Tener, de acuerdo con la normativa que se establezca, los conocimientos formativos o profesionales suficientes que permitan cursar con aprovechamiento la formación

Para cursar esta unidad formativa se debe haber superado la UF1289: Diseño de elementos software con tecnologías basadas en componentes.

En relación con las exigencias de los formadores o de las formadoras, instalaciones y equipamientos se atenderá las exigencias solicitadas para el propio certificado de profesionalidad: Programación de sistemas informáticos.