

## CUALIFICACIÓN PROFESIONAL: Programación de sistemas informáticos

Familia Profesional:	<b>Informática y Comunicaciones</b>
Nivel:	<b>3</b>
Código:	<b>IFC303_3</b>
Estado:	<b>BOE</b>
Publicación:	<b>Orden PRE/1636/2015</b>
Referencia Normativa:	<b>RD 1201/2007, RD 616/2020</b>

### Competencia general

Desarrollar componentes software a partir de unas especificaciones concretas, proporcionando funciones de administración y supervisión del sistema operativo, para la gestión de los recursos de un sistema informático y la interacción con otros sistemas utilizando tecnologías de desarrollo orientadas a objetos y a componentes.

### Unidades de competencia

**UC0490\_3:** GESTIONAR SERVICIOS EN EL SISTEMA INFORMÁTICO

**UC0964\_3:** Crear elementos software para la gestión del sistema y sus recursos

**UC0965\_3:** Desarrollar elementos software con tecnologías de programación basada en componentes

### Entorno Profesional

#### Ámbito Profesional

Desarrolla su actividad profesional en el área de sistemas o de desarrollo dedicada a la programación y mantenimiento de sistemas informáticos, en entidades de naturaleza pública o privada, empresas de tamaño pequeño/mediano/grande o microempresas, tanto por cuenta propia como ajena, con independencia de su forma jurídica. Desarrolla su actividad dependiendo, en su caso, funcional y/o jerárquicamente de un superior. Puede tener personal a su cargo en ocasiones, por temporadas o de forma estable. En el desarrollo de la actividad profesional se aplican los principios de accesibilidad universal de acuerdo con la normativa aplicable.

#### Sectores Productivos

Se ubica sobre todo en el sector servicios, y principalmente en el subsector productivo de comercialización de servicios de análisis, diseño y construcción de aplicaciones informáticas o en cualquier otro sector productivo que utilice sistemas informáticos para su gestión formando parte del equipo de programación y mantenimiento.

#### Ocupaciones y puestos de trabajo relevantes

Los términos de la siguiente relación de ocupaciones y puestos de trabajo se utilizan con carácter genérico y omnicomprensivo de mujeres y hombres.

- Programadores de sistemas
- Programadores de componentes

## Formación Asociada (510 horas)

### Módulos Formativos

**MF0490\_3:** GESTIÓN DE SERVICIOS EN EL SISTEMA INFORMÁTICO (90 horas)

**MF0964\_3:** Desarrollo de elementos software para gestión de sistemas (210 horas)

**MF0965\_3:** Desarrollo de software basado en tecnologías orientadas a componentes (210 horas)

## UNIDAD DE COMPETENCIA 1

### GESTIONAR SERVICIOS EN EL SISTEMA INFORMÁTICO

Nivel: 3  
Código: UC0490\_3  
Estado: BOE

#### Realizaciones profesionales y criterios de realización

**RP1:** Gestionar la configuración del sistema para asegurar el rendimiento de los procesos según las necesidades de uso, considerando despliegues en arquitecturas dedicadas o distribuidas, con y sin virtualización y cumpliendo las directivas de la organización.

**CR1.1** Los procesos que intervienen en el sistema se identifican de forma que permitan evaluar parámetros de rendimiento, diferenciando los procesos que se encuentran repartidos en diferentes nodos, (si la arquitectura es distribuida) y/o si están asociados al software de gestión de la virtualización, al hipervisor de los host físicos o a los propios servicios virtualizados (si se trata de un modelo virtualizado).

**CR1.2** Los parámetros que afectan a los componentes del sistema: memoria, procesador y periféricos, entre otros, se ajustan a las necesidades de uso asignándoles la configuración que maximice el rendimiento.

**CR1.3** Las prioridades de ejecución de los procesos se adecuan en función de las especificaciones del plan de explotación de la organización (tipo de proceso, usuario, perfil, entre otros).

**CR1.4** Las herramientas de monitorización se implantan, configurándolas y determinando los niveles de las alarmas.

**CR1.5** La conectividad y el ancho de banda que se necesita en arquitecturas distribuidas, se proporcionan según las especificaciones y/o manuales de fabricantes y de la organización.

**CR1.6** La distribución de la información en arquitecturas distribuidas se gestiona, siguiendo las especificaciones y/o manuales de fabricantes y de la organización, para maximizar el rendimiento del sistema.

**CR1.7** El software de gestión de virtualización y el hipervisor, de los hosts físicos y los propios servicios virtualizados, en el caso de despliegues virtualizados, se gestiona, revisando la configuración y monitorizando el rendimiento, siguiendo las especificaciones y/o manuales de fabricantes y de la organización, y maximizando el rendimiento del sistema.

**RP2:** Administrar el almacenamiento según las necesidades de uso, considerando despliegues en arquitecturas dedicadas o distribuidas, con y sin virtualización y cumpliendo las directivas de la organización.

**CR2.1** Los dispositivos de almacenamiento se configuran para ser usados, asignando los parámetros propios del sistema operativo utilizado en el sistema informático.

**CR2.2** El almacenamiento se configura, teniendo en cuenta la posible necesidad de arquitecturas distribuidas que requieran distribución de la información, así como la necesidad de entornos virtualizados que requieren software de gestión de virtualización, hipervisores y los propios servicios virtualizados.

**CR2.3** La estructura de almacenamiento se define, implantándose, atendiendo a las necesidades de los sistemas de archivos y a las especificaciones de uso de la organización.

**CR2.4** Los requerimientos de nomenclatura de objetos y restricciones de uso del almacenamiento se documentan, siguiendo el formato (tipo de documento, tamaño, maquetación, tipografía, entre otros) y otras indicaciones establecidas por la organización.

**CR2.5** El almacenamiento se integra para ofrecer un sistema funcional al usuario, siguiendo las especificaciones de la organización, con independencia del tipo de arquitectura (distribuida o dedicada) y de la existencia o no de capa de virtualización.

**RP3:** Gestionar las tareas de usuarios para garantizar los accesos al sistema y la disponibilidad de los recursos según especificaciones de explotación del sistema informático.

**CR3.1** El acceso de los usuarios al sistema informático se configura, asignando métodos de autenticación y perfiles, entre otros, para garantizar la seguridad e integridad del sistema.

**CR3.2** El acceso de los usuarios a los recursos se administra mediante la asignación de permisos en función de las necesidades de la organización.

**CR3.3** Los recursos disponibles (dispositivos, espacio, número de conexiones, caudal/ancho de banda, entre otros) para los usuarios se limitan, usando las herramientas instaladas en el sistema, en base a lo especificado en las normas de uso de la organización.

**RP4:** Gestionar los servicios de red para asegurar la comunicación entre sistemas informáticos según necesidades de explotación.

**CR4.1** Los servicios de comunicación se establecen con un sistema de calidad de servicio, garantizándose las comunicaciones de los mismos.

**CR4.2** Los dispositivos de comunicaciones se verifican en lo que respecta a su configuración y rendimiento, siguiendo las especificaciones de la organización.

**CR4.3** Los consumos de recursos de los servicios de comunicaciones se analizan, verificando que se encuentran dentro de los límites permitidos por las especificaciones.

**CR4.4** Las incidencias detectadas en los servicios de comunicaciones se documentan para informar a los responsables de la explotación del sistema y de la gestión de las mismas según los protocolos de la organización indicando, entre otros, el momento, la descripción y la solución aplicadas al problema.

## Contexto profesional

### Medios de producción

Sistemas operativos. Herramientas de administración de usuarios y gestión de permisos a recursos. Herramientas de control de rendimiento. Herramientas de monitorización de procesos. Herramientas de monitorización de uso de memoria. Herramientas de monitorización de gestión de dispositivos de almacenamiento. Herramientas de gestión de usuarios.

### Productos y resultados

Dispositivos de almacenamiento configurados y estructurados. Sistema configurado y operando. Rendimiento del sistema según los parámetros de explotación. Usuarios gestionados. Sistema seguro e íntegro en el acceso y utilización de servicios y recursos. Servicios de comunicaciones en funcionamiento.

### Información utilizada o generada

Normas externas de trabajo (normativa aplicable de protección de datos y publicación de la información). Normas internas de trabajo (plan de explotación de la organización; gráficas y análisis de rendimiento; listados de acceso y restricciones de usuarios; informe de incidencias; protocolo de actuación ante incidencias). Documentaciones técnicas (manuales de explotación del sistema operativo y de los dispositivos; manuales de las herramientas de monitorización utilizadas).

## UNIDAD DE COMPETENCIA 2

### Crear elementos software para la gestión del sistema y sus recursos

Nivel: 3  
Código: UC0964\_3  
Estado: BOE

#### Realizaciones profesionales y criterios de realización

**RP1:** Desarrollar componentes software que implementen servicios y herramientas de gestión del sistema operativo, utilizando lenguajes orientados a la programación de sistemas, para soportar tareas administrativas según necesidades funcionales dadas.

**CR1.1** Los recursos a utilizar en el desarrollo de los componentes software o herramienta de gestión a implementar se definen, analizando las especificaciones técnicas y las necesidades funcionales detectadas.

**CR1.2** La diagramación y documentación previa al desarrollo del componente se elabora especificando los requisitos necesarios para su inclusión en librerías de modo que se facilite su reutilización.

**CR1.3** Los componentes software se codifican en función de las especificaciones técnicas del diseño suministrado, haciendo uso de herramientas de programación y depuración y reutilizando funciones de librería del sistema.

**CR1.4** El plan de pruebas se elabora de forma que se asegure la funcionalidad de los componentes desarrollados, según especificaciones y criterios de calidad establecidos.

**CR1.5** Los componentes software de servicios y de herramientas desarrollados se prueban y depuran, para corregir los errores utilizando las herramientas de depuración del entorno de programación según los criterios de calidad establecidos.

**CR1.6** La documentación de los componentes software de servicios y herramientas de gestión se elabora, siguiendo los patrones, normativa y procedimientos especificados por la organización.

**RP2:** Codificar las funciones de las librerías del sistema para su utilización en el desarrollo de componentes software y optimización de los desarrollos según especificaciones técnicas y funcionales.

**CR2.1** Las funciones documentadas de las librerías del sistema se identifican y catalogan, para facilitar la localización de la información de las mismas según necesidades de desarrollo.

**CR2.2** Las funciones de las librerías del sistema se utilizan en la elaboración de nuevos componentes software, para mejorar los rendimientos de los desarrollos mediante la reutilización del código escrito y probado, según las especificaciones técnicas de cada función y los requisitos de invocación de las mismas.

**CR2.3** Los componentes software se desarrollan con los requisitos especificados, para ser incluidos en librerías para su posterior uso y distribución, según necesidades y especificaciones técnicas.

**CR2.4** Las pruebas funcionales y estructurales del componente realizado se planifican y se realizan, para comprobar y asegurar los objetivos del desarrollo según especificaciones técnicas y de calidad de la organización.

**CR2.5** La documentación de las librerías del sistema operativo desarrolladas se realiza, para cumplimentar las necesidades de registro siguiendo los patrones, normativa y procedimientos especificados en el diseño.

**RP3:** Elaborar componentes software utilizando lenguajes orientados a la programación de sistemas, según especificaciones establecidas para manejar dispositivos hardware.

**CR3.1** Las características y los parámetros de los componentes software a desarrollar se definen en función de las especificaciones de los dispositivos hardware a manejar y del sistema operativo a utilizar, detallando los requisitos necesarios para su inclusión en librerías de modo que se facilite su reutilización.

**CR3.2** Los manejadores de dispositivos se codifican haciendo uso de herramientas de programación y depuración y utilizando funciones de librería del sistema ya existentes.

**CR3.3** El plan de pruebas se elabora de forma que asegure la funcionalidad del componente y la ausencia de conflictos con otros componentes del sistema.

**CR3.4** Los manejadores de dispositivo se prueban, siguiendo el plan elaborado y en los posibles escenarios en los que puede ser implantado.

**CR3.5** La documentación técnica y de usuario del manejador desarrollado se confecciona según los parámetros y la normativa de la organización.

**RP4:** Desarrollar componentes software que implementen servicios de comunicaciones, para enlazar distintos sistemas según estándares de desarrollo.

**CR4.1** Las características de los componentes software a desarrollar (puertos de comunicaciones entre sistemas, protocolos estándares, entre otros) se definen en función de las especificaciones del servicio de comunicaciones a implementar y del sistema operativo a utilizar, detallando los requisitos necesarios para su inclusión en librerías de modo que se facilite su reutilización.

**CR4.2** Los componentes software de comunicación entre sistemas se codifican haciendo uso de herramientas de programación y depuración, y utilizando las funciones de librería del sistema ya existentes.

**CR4.3** El componente se somete a baterías de pruebas en réplicas de los posibles escenarios de su implantación posterior, para asegurar la funcionalidad e integridad según criterios de calidad y seguridad de la organización.

**CR4.4** La documentación del desarrollo y pruebas realizadas se confecciona siguiendo los patrones, normativa y procedimientos especificados en el diseño.

## Contexto profesional

### Medios de producción

Cortafuegos antivirus y servidores proxy. Entornos integrados de desarrollo. Equipos informáticos, periféricos y dispositivos hardware. Herramientas de control de versiones. Herramientas de prueba. Herramientas de depuración. Herramientas de desarrollo o entornos integrados (IDE). Herramientas de distribución de aplicaciones. Herramientas de documentación de elementos de programación. Herramientas ofimáticas. Lenguajes de modelización. Lenguajes estructurados. Lenguajes orientados a objetos. Lenguajes de programación concurrentes.

## Productos y resultados

Componentes software de servicios y herramientas del sistema operativo. Componentes software de manejo de dispositivos hardware. Componentes software de servicios de comunicaciones.

## Información utilizada o generada

Documentación sobre los casos y datos de prueba desarrollados. Documentación técnica del diseño del software a desarrollar. Documentación técnica y de usuario del software desarrollado. Manuales de funcionamiento del software. Manuales de interfaces de programación (API) del sistema operativo. Manuales de la herramienta de programación empleada. Manuales de uso del sistema operativo. Manuales del entorno de programación (IDE). Manuales del lenguaje de programación empleado. Manuales técnicos del dispositivo hardware a programar. Normas corporativas de desarrollo de software, de pruebas, de control de calidad. Sistemas de ayuda de las aplicaciones informáticas. Soportes técnicos para asistencia (telefónica, Internet, mensajería y foros, entre otros).

## UNIDAD DE COMPETENCIA 3

### Desarrollar elementos software con tecnologías de programación basada en componentes

Nivel: 3  
Código: UC0965\_3  
Estado: BOE

#### Realizaciones profesionales y criterios de realización

**RP1:** Diseñar el componente software, para asegurar su desarrollo según las especificaciones recibidas.

**CR1.1** La diagramación y documentación previa al desarrollo del componente se elabora según especificaciones técnicas de la arquitectura de componentes y las necesidades funcionales.

**CR1.2** Los interfaces del componente software a desarrollar se definen para la intercomunicación con el resto de componentes del sistema según especificaciones técnicas de la arquitectura de componentes y necesidades funcionales.

**CR1.3** La estructura del componente se diseña utilizando los estándares de creación de componentes, para facilitar y asegurar la integración en la arquitectura y los procedimientos de desarrollo, según especificaciones técnicas de la arquitectura utilizada y necesidades funcionales.

**CR1.4** La documentación del diseño realizado se elabora, siguiendo los patrones, normas y procedimientos especificados por la organización.

**RP2:** Desarrollar el componente software para su integración en una arquitectura definida, según el diseño realizado y especificaciones recibidas.

**CR2.1** El componente se codifica haciendo uso de los lenguajes soportados por la arquitectura utilizada, según las especificaciones del diseño, utilizando herramientas de programación y depuración.

**CR2.2** La interfaz del componente se realiza 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.

**CR2.3** El componente se somete a baterías de prueba para verificar su funcionalidad según criterios de calidad y seguridad de la organización.

**CR2.4** La documentación del desarrollo y pruebas se elaboran siguiendo los patrones, normas y procedimientos especificados por la organización.

**CR2.5** Las herramientas de control de versiones centralizadas o distribuidas se utilizan para garantizar el adecuado control de cambios en el software facilitando el desarrollo y las pruebas de las modificaciones relativas a nuevas funcionalidades y corrección de errores.

**RP3:** Realizar el despliegue y la implantación de los componentes desarrollados para su integración en el sistema, según especificaciones técnicas de la arquitectura.

**CR3.1** La estructura del componente se prueba verificando que se comunica con el resto de componentes y que no produce conflictos, según criterios de calidad y seguridad de la organización.

**CR3.2** Los procedimientos de despliegue se definen según requisitos del componente desarrollado y siguiendo criterios de calidad, seguridad de la organización y especificaciones de la arquitectura, para asegurar la implantación del mismo con la provisión de sus funcionalidades y la ausencia de conflictos.

**CR3.3** El rendimiento de los componentes desarrollados se monitoriza para asegurar su integración en el sistema según criterios de calidad y seguridad de la organización.

**CR3.4** La documentación del despliegue e implantación se elabora, siguiendo los patrones, normas y procedimientos especificados por la organización.

## Contexto profesional

### Medios de producción

Entornos integrados de desarrollo. Equipos informáticos y periféricos de comunicaciones. Herramientas de control de versiones. Herramientas de depuración. Herramientas de desarrollo o entornos integrados (IDE). Herramientas de distribución de aplicaciones. Herramientas de documentación de elementos de programación. Herramientas de gestión de cambios, incidencias y configuración. Herramientas de prueba. Herramientas ofimáticas. Lenguajes de modelización. Lenguajes de manipulación de datos. Lenguajes estructurados. Lenguajes orientados a objetos. Servicios de transferencia de ficheros y mensajería. Sistemas operativos y parámetros de configuración.

### Productos y resultados

Diseño y desarrollo de componentes software a integrar en arquitectura definida. Componente software integrado en el sistema y en funcionamiento.

### Información utilizada o generada

Documentación sobre los casos y datos de prueba desarrollados. Documentación técnica del diseño del software a desarrollar. Documentación técnica y de usuario del software desarrollado. Manuales de interfaces de programación (API) del sistema operativo. Manuales de uso del sistema operativo. Manuales de uso y funcionamiento de los sistemas informáticos. Manuales del entorno de programación (IDE). Manuales del lenguaje de programación. Manuales del lenguaje de programación empleado. Manuales técnicos de los dispositivos de comunicaciones sobre los que se vaya a programar. Normas corporativas de desarrollo de software, de pruebas, de control de calidad. Soportes técnicos de asistencia (telefónica, Internet, mensajería y foros, entre otros).

## MÓDULO FORMATIVO 1

### GESTIÓN DE SERVICIOS EN EL SISTEMA INFORMÁTICO

Nivel:	3
Código:	MF0490_3
Asociado a la UC:	UC0490_3 - GESTIONAR SERVICIOS EN EL SISTEMA INFORMÁTICO
Duración (horas):	90
Estado:	BOE

#### Capacidades y criterios de evaluación

- C1:** Analizar procesos del sistema, asegurando un rendimiento acorde a los parámetros especificados en el plan de explotación considerando despliegues en arquitecturas dedicadas o distribuidas, con y sin capa de virtualización.
- CE1.1** Identificar procesos del sistema, analizando los parámetros que los caracterizan (procesos padre, estado del proceso, consumo de recursos, prioridades y usuarios afectados entre otros) para determinar su influencia en el rendimiento del sistema.
  - CE1.2** Describir cada una de las herramientas provistas por el sistema para la gestión de procesos, con objeto de permitir la intervención en el rendimiento general del sistema explicando sus características y funciones.
  - CE1.3** Explicar técnicas de monitorización y herramientas destinadas a evaluar el rendimiento del sistema, indicando qué parámetros se miden y qué funciones se controlan.
  - CE1.4** En un supuesto práctico de análisis del rendimiento de un sistema informático con una carga de procesos concreta:
    - Utilizar herramientas del sistema, monitorizando sus parámetros para identificar cuantos procesos activos existen y las características particulares de alguno de ellos.
    - Realizar operaciones de activación, desactivación y modificación de prioridad entre otras con un proceso, utilizando las herramientas del sistema.
    - Monitorizar el rendimiento del sistema, mediante herramientas específicas y definir alarmas, que indiquen situaciones de riesgo.
- C2:** Aplicar procedimientos de administración del almacenamiento para ofrecer al usuario un sistema de registro de la información íntegro, confidencial y disponible.
- CE2.1** Identificar sistemas de archivo utilizables en un dispositivo de almacenamiento dado, para optimizar los procesos de registro y acceso a los mismos.
  - CE2.2** Explicar las características de un sistema de archivo, en función de la arquitectura hardware (dedicada o distribuida), los dispositivos de almacenamiento y sistemas operativos empleados.
  - CE2.3** Describir la estructura general de almacenamiento asociando, para cada nodo o sistema informático final, los dispositivos con los sistemas de archivos existentes.
  - CE2.4** Describir la distribución del almacenamiento en nodos, dispositivos y sistemas de archivo, comprobando que se garantice la funcionalidad y el rendimiento del conjunto.
  - CE2.5** En un supuesto práctico de aplicación de procedimientos de administración de almacenamiento de la información con varios dispositivos:

- Particionar los dispositivos, en los casos que se requiera distribuir la información de manera separada, generando la infraestructura de los sistemas de archivo a instalar.
- Distribuir la información en diferentes nodos, integrándolos en un sistema de almacenamiento común, garantizando las comunicaciones y el rendimiento cuando la distribución del almacenamiento sea un requisito de implementación.
- Implementar la estructura general de almacenamiento, integrando todos los nodos, dispositivos y sus correspondientes sistemas de archivos.
- Documentar los requerimientos y restricciones de cada sistema de archivos implantado, indicando la restricción o el requerimiento y el tipo de dispositivo afectado.
- Aplicar los puntos anteriores sobre sistemas virtualizados.

**C3:** Administrar accesos al sistema y a los recursos para asegurarlos, restringiendo su uso en función del perfil de acceso.

**CE3.1** Identificar posibilidades de acceso al sistema, distinguiendo los accesos remotos de los accesos locales.

**CE3.2** Describir herramientas que se utilizan en la gestión de permisos a usuarios para el uso de los recursos del sistema.

**CE3.3** En un supuesto práctico de administración del acceso al sistema en el que se cuenta con derecho de administración de usuarios:

- Identificar los posibles accesos de un usuario al sistema, monitorizando mediante visionado de log o usando herramienta software.
- Modificar los permisos de utilización de un recurso del sistema a un usuario, estableciendo otros que se hayan solicitado.
- Definir limitaciones de uso de un recurso del sistema a los usuarios, verificando dicha limitación simulando el acceso.

**C4:** Evaluar el uso y rendimiento de los servicios de comunicaciones para mantenerlos dentro de los parámetros especificados.

**CE4.1** Explicar parámetros de configuración y funcionamiento de los dispositivos de comunicaciones, indicando los servicios afectados por cada uno para asegurar su funcionalidad dentro del sistema.

**CE4.2** Relacionar servicios de comunicaciones activos en el sistema con los dispositivos utilizados por ellos, analizando y evaluando el rendimiento.

**CE4.3** En un supuesto práctico de evaluación de uso y rendimiento de un sistema informático conectado con el exterior por medio de varias líneas de comunicaciones:

- Identificar los dispositivos de comunicaciones, describiendo sus características.
- Verificar el estado de los servicios de comunicaciones, comprobando su funcionalidad.
- Evaluar el rendimiento de los servicios de comunicaciones, midiendo los parámetros de conectividad y caudal.
- Detectar las incidencias producidas en el sistema, documentando las que se produzcan.

## Capacidades cuya adquisición debe ser completada en un entorno real de trabajo

C1 respecto a CE1.4; C2 respecto a CE2.5; C3 respecto a CE3.3; C4 respecto a CE4.3.

### Otras Capacidades:

Mantener el área de trabajo con el grado apropiado de orden y limpieza.

Demstrar cierto grado de autonomía en la resolución de contingencias relacionadas con su actividad.

Demostrar creatividad en el desarrollo del trabajo que realiza.  
Interpretar y ejecutar instrucciones de trabajo.  
Demostrar resistencia al estrés, estabilidad de ánimo y control de impulsos.  
Valorar el talento y el rendimiento profesional con independencia del sexo.  
Mostrar una actitud de respeto hacia los compañeros, procedimientos y normas de la empresa.  
Cumplir las medidas que favorezcan el principio de igualdad de trato y de oportunidades entre hombres y mujeres.  
Aplicar de forma efectiva el principio de igualdad de trato y no discriminación en las condiciones de trabajo entre mujeres y hombres.

## Contenidos

### 1 Procesos en el sistema informático

Estados de un proceso.  
Manejo de señales entre procesos.  
Administración de procesos.  
Cambio de prioridades.  
Monitorización de procesos.  
Gestión del consumo de recursos.

### 2 Almacenamiento de información en la gestión de servicios

Dispositivos de almacenamiento.  
Sistemas de archivo.  
Estructura general de almacenamiento.  
Herramientas del sistema para gestión del almacenamiento.

### 3 Gestión de usuarios en la gestión de servicios

Acceso al sistema.  
Permisos y acceso a los recursos.  
Limitaciones de uso de recursos.

### 4 Servicios de comunicaciones en la gestión de servicios

Dispositivos de comunicaciones.  
Protocolos de comunicaciones.  
Servicios de comunicaciones.  
Rendimientos de los servicios de comunicaciones.

## Parámetros de contexto de la formación

### Espacios e instalaciones

Los talleres e instalaciones darán respuesta a las necesidades formativas de acuerdo con el contexto profesional establecido en la unidad de competencia asociada, teniendo en cuenta la normativa aplicable del sector productivo, prevención de riesgos, salud laboral, accesibilidad universal y protección medioambiental. Se considerará con carácter orientativo como espacios de uso:

- Instalación de 2 m<sup>2</sup> por alumno o alumna.

### Perfil profesional del formador o formadora:

1. Dominio de los conocimientos y las técnicas relacionados con la gestión de servicios en el sistema informático, que se acreditará mediante una de las dos formas siguientes:

- Formación académica de nivel 2 (Marco Español de Cualificaciones para la Educación Superior), Ingeniero Técnico, Diplomado o de otras de superior nivel relacionadas con el campo profesional.
  - Experiencia profesional de un mínimo de 2 años en el campo de las competencias relacionadas con este módulo formativo.
2. Competencia pedagógica acreditada de acuerdo con lo que establezcan las Administraciones competentes.

## MÓDULO FORMATIVO 2

### Desarrollo de elementos software para gestión de sistemas

Nivel:	3
Código:	MF0964_3
Asociado a la UC:	UC0964_3 - Crear elementos software para la gestión del sistema y sus recursos
Duración (horas):	210
Estado:	BOE

### Capacidades y criterios de evaluación

- C1:** Analizar las especificaciones técnicas proporcionadas para el desarrollo a realizar, comprobando su validez y optimización.
- CE1.1** Clasificar las principales arquitecturas de sistemas operativos (monolítico, modular, 'microkernel', sistemas distribuidos) y sus características, para identificar las necesidades de gestión y administración de dichos sistemas según especificaciones técnicas.
  - CE1.2** Explicar los apartados de un análisis de requisitos, detallando los diagramas básicos utilizados para la especificación funcional y de datos según metodologías y estándares de diseño.
  - CE1.3** Analizar las especificaciones técnicas del servicio o herramienta de gestión que se desea implementar, para identificar los recursos requeridos del componente según las especificaciones funcionales dadas.
  - CE1.4** Realizar los diagramas previos a la fase de desarrollo siguiendo las especificaciones técnicas y criterios de calidad especificados.
  - CE1.5** En un supuesto práctico, de interpretación de un análisis de requisitos dado para el desarrollo de un elemento software, teniendo en cuenta las necesidades de administración de los recursos del sistema informático:
    - Definir los módulos software a realizar a partir de las especificaciones técnicas y el catálogo de requisitos.
    - Establecer las relaciones entre módulos determinando entradas, salidas y flujos de datos según el diseño funcional y las especificaciones del sistema.
    - Determinar las estructuras necesarias para representar la información especificada en los requisitos.
    - Crear las estructuras necesarias para realizar el seguimiento de requisitos durante la codificación del elemento software según unos criterios de calidad especificados.
- C2:** Desarrollar elementos software destinados a la gestión de los recursos del sistema, mediante herramientas y lenguajes de programación de sistemas.
- CE2.1** Enumerar las herramientas y lenguajes estructurados, orientados al desarrollo de programas para la gestión de recursos del sistema, detallando sus características.
  - CE2.2** Describir las técnicas de funcionamiento y principios de los sistemas de memoria, detallando su organización en jerarquías, para desarrollar elementos software que las utilicen según las especificaciones funcionales aportadas.

**CE2.3** Clasificar las arquitecturas de entrada/salida, de buses y de microprocesadores en sistemas, explicando las técnicas y procesos funcionales utilizados para el desarrollo de los elementos software, según unas especificaciones funcionales dadas.

**CE2.4** Distinguir las funciones de las librerías del sistema para la elaboración de nuevos componentes software, reutilizando el código ya desarrollado.

**CE2.5** Describir como incorporar nuevos elementos en las librerías del sistema, según unos criterios de optimización y calidad especificados.

**CE2.6** Describir los tipos de pruebas, tanto funcionales como estructurales, y los procesos de depuración a los que debe ser sometido un componente desarrollado y las herramientas utilizadas, para verificar su funcionalidad e integración con el resto de componentes del sistema, según unos criterios de calidad especificados.

**CE2.7** Clasificar las herramientas utilizadas en las fases de desarrollo: generación del código, creación de los módulos ejecutables, control de versiones, depuración y pruebas, documentación y empaquetado para su distribución para optimizar la generación y asegurar la calidad de los productos del desarrollo, según unos criterios de seguridad y calidad especificados.

**CE2.8** En varios supuestos prácticos de diseño y codificación de elementos software para la gestión de los recursos del sistema, dadas unas especificaciones técnicas:

- Diseñar los algoritmos asociados al elemento software utilizando técnicas de desarrollo estructurado.
- Codificar los módulos software a partir de los algoritmos diseñados utilizando herramientas y lenguajes estructurados.
- Codificar estructuras de datos utilizando las funcionalidades proporcionadas por el lenguaje estructurado y utilizando los recursos y librerías disponibles en las herramientas de desarrollo para realizar la codificación de los algoritmos.
- Diseñar y codificar los manejadores de errores necesarios para garantizar el óptimo funcionamiento del módulo software.
- Diseñar y codificar componentes que permitan el acceso concurrente a los recursos del sistema.
- Documentar los módulos desarrollados para facilitar su revisión y futuras modificaciones y ampliaciones.

**CE2.9** En varios supuestos prácticos de prueba y modificación de elementos software para la gestión de los recursos del sistema, dado un elemento codificado, la documentación del mismo y unas especificaciones técnicas:

- Depurar los módulos desarrollados utilizando las herramientas disponibles.
- Diseñar y aplicar baterías de pruebas sobre los módulos desarrollados para comprobar su correcto funcionamiento y documentar los resultados obtenidos.
- Desarrollar los cambios necesarios en la codificación del elemento para solucionar los posibles errores de funcionamiento encontrados.
- Documentar los cambios realizados en los módulos desarrollados para facilitar su revisión y futuras modificaciones y ampliaciones.
- Realizar la implantación de los módulos documentando el proceso y las incidencias detectadas.

**C3:** Utilizar las técnicas y estándares utilizadas en el desarrollo, distribución e implantación de manejadores de dispositivos (drivers), para la integración de periféricos en el sistema informático según especificaciones técnicas y funcionales dadas.

**CE3.1** Utilizar las técnicas, estándares y herramientas de programación para desarrollar el manejador de dispositivo y depurar los posibles errores en el código desarrollado, según especificaciones técnicas de las herramientas y necesidades funcionales dadas.

**CE3.2** Realizar las pruebas del manejador del dispositivo elaborado en los posibles escenarios en los que puede ser implantado, para asegurar su funcionalidad y la ausencia de conflictos con el resto de los elementos del sistema según especificaciones técnicas y normativa de calidad dadas.

**CE3.3** Confeccionar la documentación técnica y de usuario del manejador desarrollado según unos parámetros y una normativa dadas.

**CE3.4** En un supuesto práctico de modificación de un manejador de dispositivo en el que se va a realizar un cambio en su diseño, según unas especificaciones funcionales dadas:

- Modificar el código para incorporar los cambios necesarios siguiendo los criterios de calidad especificados.

- Comprobar, mediante la realización de pruebas, que la modificación ha sido incorporada con éxito y que no ha alterado la funcionalidad del resto de manejadores, en particular, y del sistema en general.

- Realizar la actualización de toda la documentación que se vea afectada por el cambio propuesto.

**CE3.5** En un supuesto práctico, debidamente caracterizado, de desarrollo de un manejador de dispositivo a partir de unas especificaciones funcionales dadas:

- Diseñar el manejador del dispositivo utilizando técnicas de desarrollo estructurado y los estándares adecuados para el tipo de dispositivo.

- Codificar el manejador de dispositivo utilizando la herramienta seleccionada, aplicando el diseño previamente obtenido y utilizando lenguajes estructurados.

- Implantar el manejador del dispositivo documentando el código generado, los resultados obtenidos y las incidencias detectados.

**C4:** Implementar servicios de comunicaciones entre sistemas aplicando las técnicas y estándares de desarrollo de elementos software, de acuerdo a con unas especificaciones técnicas y funcionales dadas.

**CE4.1** Clasificar las arquitecturas de servicios de comunicaciones para distinguir servicios prestados en entornos cliente/servidor de entornos entre iguales (punto a punto).

**CE4.2** Describir los protocolos y puertos utilizados para la comunicación entre sistemas, teniendo en cuenta el soporte que ofrecen a los servicios de comunicaciones.

**CE4.3** Identificar las principales API's (Application Program Interface) y librerías y su uso para el desarrollo mediante programación estructurada.

**CE4.4** Codificar el componente utilizando herramientas de programación y depuración adecuadas para optimizar la fase de desarrollo según unas especificaciones técnicas dadas.

**CE4.5** Someter al componente a baterías de pruebas en réplicas de los posibles escenarios de su implantación posterior, para verificar la ausencia de conflictos y su integración con el resto de componentes del sistema, según unos criterios de calidad y seguridad dados.

**CE4.6** Clasificar los estándares definidos para el desarrollo de servicios de comunicaciones entre sistemas según diferentes criterios: organizaciones de estandarización, tipos de servicios y protocolos soportados, entre otros.

**CE4.7** Enumerar los principales problemas de seguridad en el ámbito de las comunicaciones y describir las estrategias a aplicar, para el desarrollo de componentes que implementen servicios seguros según estándares y especificaciones dadas.

**CE4.8** En varios supuestos prácticos de desarrollo de componentes para el establecimiento de servicios de comunicaciones entre sistemas, dadas unas especificaciones técnicas:

- Diseñar el componente utilizando técnicas de desarrollo estructurado y los estándares definidos.
- Codificar el elemento software utilizando la herramienta idónea, aplicando el diseño previamente obtenido y utilizando lenguajes estructurados.
- Depurar y probar el componente garantizando su óptimo funcionamiento.
- Diseñar baterías de posibles ataques contra el servicio y probarlas para detectar posibles vulnerabilidades.
- Implantar los componentes para verificar el servicio de comunicaciones documentando los resultados e incidencias detectados.
- Documentar el código desarrollado, las pruebas realizadas y el resultado de los procesos de implantación de los componentes.

## Capacidades cuya adquisición debe ser completada en un entorno real de trabajo

C1 respecto a CE1.5; C2 respecto a CE2.8 y CE2.9; C3 respecto a CE3.4 y CE3.5; C4 respecto a CE4.8.

### Otras Capacidades:

Demostrar un buen hacer profesional.

Proponerse objetivos retadores que supongan un nivel de rendimiento y eficacia superior al alcanzado previamente.

Demostrar creatividad en el desarrollo del trabajo que realiza.

Demostrar cierto grado de autonomía en la resolución de contingencias relacionadas con su actividad.

Emplear tiempo y esfuerzo en ampliar conocimientos e información complementaria para utilizarlos en su trabajo.

Aprender nuevos conceptos o procedimientos y aprovechar eficazmente la formación utilizando los conocimientos adquiridos.

## Contenidos

### 1 Características de la arquitectura de sistemas operativos

Principales arquitecturas de sistemas operativos y sus características.

Gestión de los recursos.

Librerías y funciones del sistema.

### 2 Análisis de requisitos del desarrollo

El ciclo de desarrollo del software en gestión de sistemas.

Análisis, diseño y programación para la gestión de sistemas.

Modelos de procesos: nomenclatura y representación.

Modelos de datos: nomenclatura y representación.

Herramientas de diagramación y documentación.

Métricas y estándares de calidad para desarrollo de software.

### 3 Desarrollo de elementos software para la gestión del sistema y sus recursos

Herramientas de desarrollo: editores, compiladores.

Técnicas y herramientas de programación estructurada orientadas a la programación de sistemas: tipos y estructuras de datos, operadores, estructuras lógicas, sentencias, recursividad, programación concurrente, excepciones.

Llamadas a funciones del sistema.

#### 4 Pruebas y documentación de elementos software para la gestión del sistema y sus recursos

Planificación y diseño de pruebas.

Técnicas y herramientas de depuración de software.

Pruebas y verificación del desarrollo.

Técnicas y herramientas de documentación del software desarrollado.

Herramientas de compresión y empaquetamiento de software.

Técnicas y sistemas de distribución y despliegue de software.

Sistemas de control de desarrollo (CVS).

#### 5 Manejadores de dispositivos (drivers)

Arquitectura de los subsistemas del equipo informático.

Gestión de buses y puertos de los sistemas operativos.

Gestión de los dispositivos de los sistemas operativos.

Modos específicos y estándares de desarrollo de manejadores de dispositivos.

Herramientas de desarrollo y verificación de manejadores de dispositivos.

Procedimientos de distribución e instalación de manejadores de dispositivos en sistemas operativos.

#### 6 Desarrollo de procesos de comunicaciones entre sistemas

Acceso concurrente a los recursos del sistema.

Protocolos de comunicación.

Dispositivos físicos y puertos de comunicación.

Servicios de comunicaciones.

Rendimientos de los servicios de comunicaciones.

Programación en red.

Aplicaciones cliente/servidor.

'Sockets'.

### Parámetros de contexto de la formación

#### Espacios e instalaciones

Los espacios e instalaciones darán respuesta, en forma de aula, aula-taller, taller de prácticas, laboratorio o espacio singular, a las necesidades formativas, de acuerdo con el Contexto Profesional establecido en la Unidad de Competencia asociada, teniendo en cuenta la normativa aplicable del sector productivo, prevención de riesgos, salud laboral, accesibilidad universal y protección medioambiental.

#### Perfil profesional del formador o formadora:

1. Dominio de los conocimientos y las técnicas relacionados con la creación de elementos software para la gestión del sistema y sus recursos, que se acreditará mediante una de las dos formas siguientes:

- Formación académica de nivel 2 (Marco Español de Cualificaciones para la Educación Superior), Ingeniero Técnico, Diplomado o de otras de superior nivel relacionadas con el campo profesional.
- Experiencia profesional de un mínimo de 2 años en el campo de las competencias relacionadas con este módulo formativo.

2. Competencia pedagógica acreditada de acuerdo con lo que establezcan las Administraciones competentes.



## MÓDULO FORMATIVO 3

### Desarrollo de software basado en tecnologías orientadas a componentes

Nivel:	3
Código:	MF0965_3
Asociado a la UC:	UC0965_3 - Desarrollar elementos software con tecnologías de programación basada en componentes
Duración (horas):	210
Estado:	BOE

#### Capacidades y criterios de evaluación

**C1:** Identificar las características y arquitecturas de las tecnologías de desarrollo, orientadas a componentes para la creación y modificación de elementos software integrados en estos entornos, según estándares y normalizaciones existentes.

**CE1.1** Describir las técnicas y métodos de desarrollo involucrados en el paradigma del desarrollo, orientado a componentes para la confección y modificación elementos software, según los estándares de esta tecnología.

**CE1.2** Clasificar las herramientas y lenguajes orientados a objetos utilizados en el desarrollo orientado a componentes, describiendo sus características para identificar las que son específicas para la creación o modificación de los elementos software, según las especificaciones funcionales dadas.

**CE1.3** Clasificar los estándares de modelos de componentes, describiendo las pasarelas para interoperar entre componentes heterogéneos, para realizar las tareas de integración de los elementos desarrollados según especificaciones funcionales y técnicas.

**CE1.4** Identificar las técnicas de diagramación y documentación para el desarrollo de software basado en tecnologías orientadas a componentes, según estándares de diseño de metodologías orientadas a componentes.

**CE1.5** En un supuesto práctico de diseño de componentes dentro de una arquitectura dada y contando con unas especificaciones funcionales precisas:

- Realizar la diagramación y documentación previa al desarrollo del componente, para optimizar los procesos de creación del componente según especificaciones recibidas.
- Identificar los diferentes interfaces y técnicas utilizadas para la intercomunicación de componentes, para poder aplicarlas al desarrollo de nuevos componentes.
- Definir los interfaces del componente software a desarrollar para la intercomunicación con el resto de componentes del sistema, según especificaciones técnicas de la arquitectura de componentes y necesidades funcionales.
- Diseñar la estructura del componente utilizando los estándares de creación de componentes, según especificaciones técnicas de la arquitectura utilizada y necesidades funcionales.
- Confeccionar la documentación del diseño realizado siguiendo los patrones, normas y procedimientos especificados.

**C2:** Construir elementos software a partir de las especificaciones de necesidades y con las condiciones de desarrollo de la tecnología de componentes utilizada.

**CE2.1** Explicar los enfoques de desarrollo e implementación para la creación de componentes aplicando el principio de reutilización.

**CE2.2** 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.

**CE2.3** 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.

**CE2.4** Clasificar las herramientas de programación y depuración para optimizar la fase de desarrollo de los componentes según unas especificaciones dadas.

**CE2.5** 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.

**CE2.6** Aplicar técnicas de instalación de componentes, comprobando que ejecuta las acciones requeridas y su disponibilidad para las aplicaciones que lo invoquen, según especificaciones técnicas de la arquitectura.

**CE2.7** En un supuesto práctico de desarrollo de componentes software reutilizables, dado un diseño de su estructura y unas especificaciones funcionales y técnicas:

- Comprobar que la funcionalidad del componente diseñado puede ser extendida para futuras reutilizaciones, modificando sino el mismo para lograrlo.
- Desarrollar el componente utilizando herramientas y lenguajes orientados a objeto y según las especificaciones de diseño resultantes de la comprobación anterior.
- Depurar y probar el componente desarrollado utilizando las herramientas disponibles.
- Documentar el componente y sus interfaces para facilitar su futura reutilización.

**CE2.8** En un supuesto práctico de desarrollo de 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.
- Utilizar herramientas de control de versiones centralizadas o distribuidas para garantizar el adecuado control de cambios en el software facilitando el desarrollo y las pruebas de las modificaciones relativas a nuevas funcionalidades y corrección de errores.

**C3:** Aplicar los procedimientos de despliegue e integración del componente en un entorno de tecnología orientada a componentes, según especificaciones técnicas de la arquitectura.

**CE3.1** Describir las técnicas disponibles para realizar el proceso de búsqueda de componentes, que satisfagan los requisitos impuestos en el diseño inicial.

**CE3.2** Clasificar los métodos de evaluación y selección de componentes, basándose en una serie de requisitos impuestos por las especificaciones iniciales de diseño.

**CE3.3** Realizar las pruebas estructurales para verificar que el componente seleccionado se comunica con el resto de componentes y que no produce conflictos, según criterios de calidad y seguridad dados, y del diseño preliminar.

**CE3.4** Definir los procedimientos para el despliegue y adaptación para realizar la implantación del elemento software, según requisitos del componente desarrollado y siguiendo criterios de calidad, seguridad y especificaciones de la arquitectura.

**CE3.5** Describir los procesos de integración, configuración e interconexión de los componentes seleccionados, para que permitan construir la aplicación final según los criterios de calidad y seguridad especificados en el diseño inicial.

**CE3.6** Monitorizar el rendimiento de los componentes desarrollados o seleccionados para asegurar su integración en el sistema, siguiendo criterios de calidad y seguridad.

**CE3.7** Documentar el despliegue y la implantación siguiendo los patrones, normas y procedimientos especificados.

**CE3.8** En un supuesto práctico debidamente caracterizado, de despliegue de un componente software ya desarrollado, según unas especificaciones funcionales y técnicas dadas:

- Aplicar técnicas de prueba estructural al componente desarrollado según normas y criterios de calidad establecidos.
- Incorporar el despliegue, adaptación, configuración e integración del componente según especificaciones técnicas y de implantación del desarrollo.
- Trazar las pruebas del componente según normas de calidad y seguridad dadas.
- Configurar la herramienta para la realización de baterías de pruebas automáticas según normas y criterios de calidad dados.
- Elaborar documentación mediante las plantillas facilitadas o incorporadas en la propia herramienta.
- Generar informes de calidad y métricas, e interpretar los resultados.

## Capacidades cuya adquisición debe ser completada en un entorno real de trabajo

C1 respecto a CE1.5; C2 respecto a CE2.7 y CE2.8; C3 respecto a CE3.8.

### Otras Capacidades:

Demostrar un buen hacer profesional.

Proponerse objetivos retadores que supongan un nivel de rendimiento y eficacia superior al alcanzado previamente.

Demostrar creatividad en el desarrollo del trabajo que realiza.

Demostrar cierto grado de autonomía en la resolución de contingencias relacionadas con su actividad.

Emplear tiempo y esfuerzo en ampliar conocimientos e información complementaria para utilizarlos en su trabajo.

Aprender nuevos conceptos o procedimientos y aprovechar eficazmente la formación utilizando los conocimientos adquiridos.

## Contenidos

### 1 Programación orientada a objetos

El paradigma de la orientación a objetos.

Objetos: relaciones, atributos y métodos.

Clases y tipos de datos.

Desarrollo orientado a objetos.

Técnicas de representación de la programación orientada a objetos.

Herramientas de desarrollo de programación orientada a objetos.

### 2 Fundamentos conceptuales de orientación de componentes

El concepto de componente: componentes y objetos.  
La reutilización de código como objetivo.  
Dependencias del contexto.  
Componentes e Interfaces.  
Estados del componente.  
Especificación e implementación del componente.

### 3 Arquitecturas de componentes

Características de las arquitecturas de componentes.  
Arquitecturas actuales.  
Integración y despliegue de componentes.

### 4 Programación orientada a componentes

Programación orientada a componentes frente a programación orientada a objetos.  
Conceptos de la programación orientada a componentes.  
Lenguajes de descripción de interfaces (IDL).

### 5 Programación distribuida

Planificación y diseño de pruebas de software con tecnologías de programación basadas en componentes.  
Técnicas y herramientas de depuración de software.  
Pruebas y verificación del desarrollo.  
Técnicas y herramientas de documentación del software desarrollado.  
Herramientas de compresión y empaquetamiento de software.  
Técnicas y sistemas de distribución y despliegue de software.  
Sistemas de control de desarrollo (CVS).

## Parámetros de contexto de la formación

### Espacios e instalaciones

Los espacios e instalaciones darán respuesta, en forma de aula, aula-taller, taller de prácticas, laboratorio o espacio singular, a las necesidades formativas, de acuerdo con el Contexto Profesional establecido en la Unidad de Competencia asociada, teniendo en cuenta la normativa aplicable del sector productivo, prevención de riesgos, salud laboral, accesibilidad universal y protección medioambiental.

### Perfil profesional del formador o formadora:

1. Dominio de los conocimientos y las técnicas relacionados con el desarrollo de elementos software con tecnologías de programación basada en componentes, que se acreditará mediante una de las dos formas siguientes:
  - Formación académica de nivel 2 (Marco Español de Cualificaciones para la Educación Superior), Ingeniero Técnico, Diplomado o de otras de superior nivel relacionadas con el campo profesional.
  - Experiencia profesional de un mínimo de 2 años en el campo de las competencias relacionadas con este módulo formativo.
2. Competencia pedagógica acreditada de acuerdo con lo que establezcan las Administraciones competentes.