CUALIFICACIÓN PROFESIONAL:
Despliegue de productos software en contenedores

Familia Profesional:	***Informática y Comunicaciones***
Nivel:	***3***
Código:	***IFC822_3***
Estado:	***BOE***
Publicación:	***RD 546/2023***

Competencia general

Administrar el despliegue continuo de aplicaciones en plataformas propias ("on premise") o en la nube, preparando entorno y plataforma, definiendo el flujo de procesos ("pipeline") de desarrollo y despliegue para su automatización, monitorizando y manteniendo en funcionamiento el sistema desplegado, en condiciones de seguridad, cumpliendo la normativa aplicable en materia de propiedad intelectual e industrial y la planificación de la actividad preventiva, así como los estándares de calidad.

Unidades de competencia

UC2743_3: Preparar el entorno de despliegue en contenedores
UC2744_3: Desplegar la plataforma de ejecución de contenedores
UC2745_3: Definir el flujo de procesos ("pipeline") del desarrollador en integración continua
UC2746_3: Definir el flujo de procesos ("pipeline") de despliegue continuo de contenedores
UC2747_3: Mantener el sistema de contenedores desplegado

Entorno Profesional

Ámbito Profesional

Desarrolla su actividad profesional en el área de desarrollo, de sistemas informáticos y/o telemáticos dedicado a la gestión de despliegues en infraestructuras locales ("on premise") o en la nube, 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 y diseño universal o diseño para todas las personas de acuerdo con la normativa aplicable.

Sectores Productivos

Se ubica en el sector servicios, en el subsector de los servicios de desarrollo, instalación, mantenimiento, gestión y asistencia técnica de sistemas informáticos y telemáticos, y en cualquier sector productivo que requiera los servicios anteriores.

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.

- Técnicos de integración y despliegue continuos
- Técnicos de despliegue en contenedores

- Técnicos de automatización de despliegues
- Desarrolladores Devops
- Administradores Devops

Formación Asociada (750 horas)

Módulos Formativos

MF2743_3: Preparación de entornos de despliegue en contenedores (120 horas)

MF2744_3: Despliegue de plataformas de ejecución de contenedores (150 horas)

MF2745_3: Definición de flujos de procesos ("pipelines") del desarrollador en integración continua (180 horas)

MF2746_3: Definición de flujos de procesos ("pipelines") de despliegue continuo de contenedores (180 horas)

MF2747_3: Mantenimiento del sistema de contenedores desplegado (120 horas)

UNIDAD DE COMPETENCIA 1

Preparar el entorno de despliegue en contenedores

Nivel: *3*
Código: *UC2743_3*
Estado: *Tramitación BOE*

Realizaciones profesionales y criterios de realización

RP1: Crear el acceso a los repositorios de código de infraestructura y/o servicios, o en su caso, solicitar la creación del mismo, verificando la conexión, comprobando las herramientas de gestión y validación, para su uso en el desarrollo y despliegue, siguiendo procedimientos, estándares y políticas de seguridad definidas por la entidad responsable del desarrollo, para facilitar a los usuarios y/o grupos de gestión el uso de los repositorios de código de infraestructura y/o servicios en las fases de desarrollo y despliegue.

CR1.1 El acceso a los recursos y herramientas para desplegar y/o gestionar los entornos de producción definidos en la arquitectura del proyecto se solicita al departamento o persona responsable del mismo, usando los canales de comunicación acordados entre desarrolladores y responsables de sistemas, o bien se verifica o bien se crea en su caso, para tener control de acceso y mantener el registro de acceso.

CR1.2 Los accesos al código fuente se configuran para ser gestionados a través de las herramientas proporcionadas por la entidad responsable del desarrollo, parametrizando la seguridad y las políticas de acceso, a través de los usuarios y/o grupos asignados.

CR1.3 Las herramientas de validación de calidad y seguridad del código, y dependencias de librerías externas o integraciones se verifican, previa instalación en su caso, comprobando su funcionalidad para ser usadas por los usuarios y/o grupos de gestión.

RP2: Validar la calidad y seguridad de las modificaciones previas al despliegue, usando las herramientas de gestión y validación y siguiendo procedimientos, estándares y políticas de seguridad definidas por la entidad responsable del desarrollo.

CR2.1 Las modificaciones del código fuente se validan, ejecutando pruebas de calidad mediante herramientas específicas, documentándolas con las guías de desarrollo y los flujos de trabajo y/o políticas definidas por la entidad responsable del desarrollo.

CR2.2 Las comprobaciones de parámetros de seguridad del código definidos en los estándares de la entidad responsable del desarrollo, se ejecutan mediante herramientas específicas, de forma periódica según el método definido por la propia entidad, para garantizar su buena praxis.

CR2.3 Las dependencias del código de librerías externas o integraciones se validan, mediante herramientas de comprobación, para verificar la seguridad y funcionalidad e integración con el resto de componentes del aplicativo antes del despliegue.

RP3: Crear o, en su caso, configurar entornos de revisión y/o producción, mediante ficheros de parámetros y creando flujos de trabajo, siguiendo procedimientos,

estándares y políticas de seguridad definidas por la entidad responsable del desarrollo para validar el despliegue.

CR3.1 Los ficheros de definición de infraestructura como código y SDH ("Software Define Hardware") se crean, usando el lenguaje propio de la plataforma de despliegue proporciona por la entidad responsable del desarrollo, previa identificación de los entornos de despliegue del código, artefactos e imágenes de contenedores.

CR3.2 Los flujos de trabajo para el despliegue del entorno "hardware" se crean, usando las credenciales proporcionadas, aplicando las configuraciones definidas, para crear el entorno de desarrollo o producción.

CR3.3 Las definiciones técnicas de uso de control y de seguridad se configuran para cada entorno, especificando:

- Roles de acceso del usuario.
- Roles de acceso del código.
- Acceso al almacenamiento.
- Acceso a las API ("Application Programming Interface"), entre otros recursos, siguiendo las indicaciones de la entidad responsable del desarrollo para su posterior gestión.

CR3.4 La infraestructura definida en los ficheros al efecto se crea dentro de la plataforma proporcionada por la entidad responsable del desarrollo, usando las credenciales facilitadas, para su posterior integración en los flujos de trabajo.

CR3.5 La infraestructura creada se verifica después del despliegue, comprobando que se obtiene acceso a todos los recursos de infraestructura.

RP4: Instalar en su caso el "software" de automatización y gestión de paquetes y dependencias, y de administración de la configuración del "software", creando flujos de mantenimiento y despliegue sobre las plataformas seleccionadas por la entidad responsable del desarrollo, configurando los parámetros de uso para preparar la automatización del despliegue.

CR4.1 Las herramientas de automatización y gestión de paquetes y dependencias se instalan en la infraestructura seleccionada, actualizándolas según las directrices de versión y documentación del fabricante.

CR4.2 Los flujos para mantenimiento se crean, usando las herramientas para la automatización de la administración de la configuración del "software" y servicios de los sistemas desarrollados.

CR4.3 Los flujos para mantenimiento se verifican, probando que las acciones automatizadas generan los resultados esperados según la documentación del proyecto.

RP5: Definir las variables de trabajo, configurándolas dentro de cada entorno para ser usadas por los contenedores que tienen el código y que se crean de forma dinámica y para ejecutar la aplicación.

CR5.1 Las reglas de seguridad, niveles de servicio y consumo definidas en el proyecto, se aplican como configuraciones dentro de las herramientas de despliegue, para parametrizar la ejecución de los contenedores.

CR5.2 Las métricas para elaborar informes de rendimiento y planificación ("capacity plan"), anticipar el crecimiento y facilitar la gestión de los recursos que requieren las herramientas se recogen, incluyéndolas en las herramientas de monitorización.

CR5.3 La documentación sobre el uso de los despliegues de código e imágenes de contenedores se redacta, para facilitar la reutilización de los componentes ya preparados, almacenándola en

los repositorios de código para servir como guía o modelo para posteriores desarrollos y despliegues.

CR5.4 Las acciones automáticas basadas en eventos disparados por errores tales como:
- Fallos de trabajo en los flujos.
- Disponibilidad de componentes de integración.
- Fallos de acceso.
- Estado de finalización.

Entre otros, se incorporan en las herramientas desplegadas, para ajustar el rendimiento y mantener los recursos utilizados en niveles óptimos.

Contexto profesional

Medios de producción

Conexión a la red. Equipamiento informático: equipos, componentes, periféricos, cableado, entre otros. Sistemas operativos. Navegadores. Lenguajes de "scripting". Lenguajes estructurados para automatizaciones. Lenguajes declarativos. Utilidades y aplicaciones incorporadas a los sistemas operativos. Herramientas de comunicación y colaboración en equipo. Sistemas gestores de repositorios de código fuente. Servicios de transferencia de ficheros y conexión remota. Herramientas de gestión y control de cambios, incidencias y configuración. Repositorio de artefactos/imágenes. Servidor de despliegues. Servidores de infraestructura. Sistema de monitorización. Sistema de orquestación de contenedores.

Productos y resultados

Accesos a los repositorios de código de infraestructura y/o servicios creados y verificados. Calidad y seguridad de las modificaciones previas al despliegue validadas. Entornos de revisión y/o producción creados y configurados. "Software" de automatización y gestión de paquetes y dependencias y de administración de la configuración del "software" instalado. Variables de trabajo definidas y configuradas.

Información utilizada o generada

Normas externas de trabajo (Normativa aplicable sobre prevención de riesgos laborales -ergonomía-; Normativa aplicable de protección de datos, propiedad intelectual e industrial). Normas internas de trabajo (guías de despliegue, instalación, configuración y actualización de los servicios de las aplicaciones desarrolladas; documentación de diseño y aprovisionamiento de los recursos; diseño y especificaciones de los servicios a desplegar y operar; plan de seguridad, operación y calidad; acuerdos de nivel de servicio -SLA-; documentación de configuración de sistemas y servicios; plan de pruebas e informe de fallos; especificaciones de la arquitectura de referencia de servicio corporativo; documentación asociada a los scripts desarrollados; documentación de las pruebas de funcionamiento de los servicios y aplicaciones desarrolladas). Documentación técnica (documentación técnica asociada a servicios informáticos; manuales de funcionamiento de las herramientas de gestión del ciclo de vida del "software", bases de datos de vulnerabilidades).

UNIDAD DE COMPETENCIA 2

Desplegar la plataforma de ejecución de contenedores

Nivel: *3*
Código: *UC2744_3*
Estado: *Tramitación BOE*

Realizaciones profesionales y criterios de realización

RP1: Crear las nubes virtuales privadas ("Virtual Private Cloud" -VPC-) donde se desplegará la infraestructura, mediante un hipervisor, configurando comunicaciones y seguridad, para permitir la gestión de la virtualización en todos sus niveles.

CR1.1 El hipervisor se instala, configurándolo con las opciones que permiten la virtualización de sistemas, teniendo en cuenta sus limitaciones y capacidades de configuración.

CR1.2 La VPC se crea, usando la consola del hipervisor, diferenciando las VPC de los entornos, realizando aislamientos y capacidades de configuración.

CR1.3 Los parámetros tales como IP, zona de disponibilidad en su caso y DNS se asignan, configurando las comunicaciones entre las zonas de un mismo entorno.

CR1.4 La seguridad se configura, desplegando capas de seguridad dependiendo de la exposición DMZ externa, DMZ interna y entorno interno y ejecutando pruebas de seguridad, aislamiento y separación de clientes.

RP2: Crear la infraestructura como código (IaC), seleccionando la infraestructura a desplegar y creando los códigos de despliegue, para permitir la automatización mediante procesos y adaptar las aplicaciones y servicios con mayor rapidez.

CR2.1 La infraestructura a desplegar tal como almacenamiento, capacidad de cómputo, memoria, entre otros, se selecciona, recogiendo los requerimientos para el servicio que va a alojar.

CR2.2 El código que automatice el despliegue de la infraestructura seleccionada se crea usando un editor y el lenguaje fijado por la plataforma, en base a los requerimientos del servicio a alojar.

CR2.3 El código para el despliegue de la infraestructura se ejecuta, lanzando pruebas de servicio y chequeo de configuración.

CR2.4 Las configuraciones de seguridad se verifican, comprobando si el despliegue automatizado de IaC cumple con la normativa interna y las recomendaciones de seguridad en el entorno de la entidad responsable de la instalación.

RP3: Desplegar el orquestador, configurando proyecto y tareas y creando el "pipeline", para ayudar a realizar las funciones de integración continua y la automatización cuando un evento lo indica.

CR3.1 El "software" de orquestador se selecciona, escogiendo aquel con capacidad de llevar a cabo las pruebas de control de las variables o limitaciones del CI.

CR3.2 El "software" de orquestador se instala y/o se configura, parametrizando permisos, accesos, certificados y opciones de despliegue, entre otras.

CR3.3 Los "plugin" o complementos del orquestador se instalan, configurándolos para incorporar las necesidades de integración del orquestador con las tecnologías del entorno.

CR3.4 El proyecto se crea, definiendo las configuraciones base del orquestador de tareas, así como las credenciales para el acceso al repositorio.

CR3.5 Las tareas a realizar dentro del proyecto se crean, parametrizando las opciones tales como nombre de la tarea, máquinas en las que se procesan los trabajos del despliegue, direccionamiento, y condiciones de ejecución del despliegue tales como aseguramiento de la calidad y seguridad.

CR3.6 El "pipeline" se crea, parametrizándolo para encadenar procesos de modo que el resultado del primero desencadene una acción en el nido de trabajos definidos dentro de un proyecto, definiendo las etapas, secuencias y "script" que marquen los estados del proyecto.

CR3.7 El "pipeline" se prueba ejecutando los caminos posibles dentro del mismo, verificando que se gestionan los errores y salidas no esperadas.

CR3.8 La seguridad se configura, asignando los permisos que garanticen la ejecución de las tareas planificadas siguiendo el principio de "mínimo privilegio".

RP4: Desplegar la monitorización para el entorno, seleccionando y configurando el gestor de ingesta de datos y la seguridad para facilitar el control y gestión de problemas.

CR4.1 El gestor de ingesta de datos se selecciona, escogiendo un "software" con capacidad de recolección de datos y posterior visualización de estos.

CR4.2 El "software" de gestión de datos y métricas se configura, seleccionando el visualizador de métricas, configurando los paneles, alertas y envíos de los indicadores clave de rendimiento ("Key Performance Indicator" -KPI-).

CR4.3 La seguridad se configura, parametrizando comunicaciones, accesos y repositorios de datos.

Contexto profesional

Medios de producción

Conexión a la red. Equipamiento informático: equipos, componentes, periféricos, cableado, entre otros. Sistemas operativos. Navegadores. Editores e intérpretes de lenguajes de "scripting". Lenguajes estructurados para automatizaciones. Lenguajes declarativos. Utilidades y aplicaciones incorporadas a los sistemas operativos. Herramientas de comunicación y colaboración en equipo. Sistemas gestores de repositorios de código fuente. Servicios de transferencia de ficheros y conexión remota. Herramientas de gestión y control de cambios, incidencias y configuración. Repositorio de artefactos/imágenes. Servidor de despliegues. Servidores de infraestructura. Sistema de monitorización.

Productos y resultados

Nubes virtuales privadas creadas. Infraestructura como código (IaC) creada. Orquestador desplegado. Monitorización para el entorno desplegada.

Información utilizada o generada

Normas externas de trabajo (Normativa aplicable sobre prevención de riesgos laborales -ergonomía-; Normativa aplicable de protección de datos, propiedad intelectual e industrial). Normas internas de trabajo (guías de despliegue, instalación, configuración y actualización de los servicios de las aplicaciones desarrolladas; documentación de diseño y aprovisionamiento de los recursos; diseño y especificaciones de los servicios a desplegar y operar; plan de seguridad, operación y calidad; acuerdos de nivel de servicio -SLA-; documentación de configuración de sistemas y servicios; plan de pruebas e

informe de fallos; especificaciones de la arquitectura de referencia de servicio corporativo; documentación asociada a los scripts desarrollados; documentación de las pruebas de funcionamiento de los servicios y aplicaciones desarrolladas). Documentación técnica (documentación técnica asociada a servicios informáticos; manuales de funcionamiento de las herramientas de gestión del ciclo de vida del "software", bases de datos de vulnerabilidades).

UNIDAD DE COMPETENCIA 3

Definir el flujo de procesos ("pipeline") del desarrollador en integración continua

Nivel: *3*
Código: *UC2745_3*
Estado: *Tramitación BOE*

Realizaciones profesionales y criterios de realización

RP1: Gestionar los repositorios de código fuente del "software" y de los servicios asociados a las aplicaciones de los sistemas, según las necesidades de uso, directivas de calidad y seguridad de la entidad responsable del desarrollo, para facilitar su mantenimiento, recuperación y permitir la trazabilidad del sistema.

CR1.1 Los orígenes de código fuente se organizan con una estructura que permite su uso de forma consistente, definiendo ramas de código estable y validado y otras donde se recojan los cambios que están en proceso.

CR1.2 Los parámetros del sistema que afectan a la autenticación y autorización se configuran, ajustándolos a las necesidades de acceso, integración con herramientas y seguridad de la entidad responsable del desarrollo.

CR1.3 Las modificaciones sobre el código fuente se validan, evaluando de manera automatizada la sintaxis y la semántica del código, comprobando versiones de librerías externas y/o genéricas, siguiendo las guías de desarrollo y los flujos de trabajo y políticas tales como aprobación, asignación o revisión, entre otras, definidas por la entidad responsable del desarrollo.

CR1.4 La seguridad del código se comprueba, verificando mediante "software" específico que no contenga código malicioso y que no contenga vulnerabilidades.

CR1.5 Los procesos de copia de seguridad y recuperación del código fuente, se ejecutan de forma periódica, gestionando repositorios de gran tamaño.

RP2: Modificar el código fuente de integración y plantillas responsables de la creación de los servicios, definiendo los parámetros de los artefactos, cumpliendo las directivas de operación, calidad y seguridad de la entidad responsable del desarrollo, para simplificar la operación y la integración.

CR2.1 Los servicios requeridos para las aplicaciones se crean de forma automatizada, modificándolos, en su caso, empleando línea de comandos (CLI), API ("Application Programming Interface"), automatismos mediante lenguajes de programación, entre otras.

CR2.2 Los parámetros de los artefactos para el automatismo del ciclo de vida de los servicios se definen, considerando características propias del despliegue de las versiones de los datos de las aplicaciones, tales como creación de bases de datos, movimiento o transformación de la información y metadatos, entre otras.

CR2.3 Los parámetros de los artefactos para el automatismo del ciclo de vida de los servicios relacionados con las aplicaciones se definen, considerando características propias de la integración de las versiones del "software", tales como la gestión de la configuración de las aplicaciones, entre otras.

CR2.4 Los parámetros de los artefactos para el automatismo del ciclo de vida de los servicios relacionados con infraestructura se definen, considerando características propias de la

integración de las versiones del código fuente de las aplicaciones, tales como contenedores, máquinas virtuales, máquinas físicas, scripts, código binario, entre otros.

CR2.5 Los parámetros de los artefactos para el automatismo del ciclo de vida de los servicios se definen, considerando elementos que permitan su reutilización en futuros despliegues, tales como nombre del servicio, región geográfica, recursos asignados, permisos, confirmando que son únicos en su caso.

CR2.6 El código fuente de la integración, plantillas declarativas del servicio o cualquier proceso responsable de esta tarea se verifica que sea idempotente, siendo robusta su ejecución y proporcionando predictibilidad bajo circunstancias variables.

RP3: Configurar los servicios de comunicación y colaboración del grupo de personas del proyecto según las necesidades de uso, directivas de comunicación y adopción de la entidad responsable del desarrollo, para automatizar las interacciones con los repositorios de código fuente y las herramientas de gestión de proyectos.

CR3.1 Las plataformas de comunicación y herramientas de gestión de proyectos se emplean, configurando los repositorios de código fuente de modo que permitan la recepción automática de cambios de estado y contenido.

CR3.2 Las plataformas de comunicación empleadas se configuran para notificar a los responsables de los sistemas afectados acerca de métricas, alertas o reglas definidas en los repositorios de código fuente, estados de tareas, peticiones de cambios al sistema, entre otras.

CR3.3 Las plataformas de comunicación, documentación y herramientas de gestión de proyectos empleadas se configuran, conectándolas con los repositorios de código fuente, de tal modo que permitan relacionar errores ("bugs") con modificaciones de código fuente, entre otras.

RP4: Validar el resultado de los procesos de integración continua (CI) del código fuente de las aplicaciones desarrolladas, dentro del marco de las directivas de la entidad responsable del desarrollo sobre operación, calidad y seguridad para su publicación.

CR4.1 Los fallos de ejecución, calidad, seguridad y rendimiento de las aplicaciones del sistema se resuelven mediante automatización, incluyendo las pruebas de diagnóstico con las herramientas integradas, proporcionando información sobre resultados y acciones a los fallos diagnosticados siguiendo las estrategias del responsable del diseño de pruebas.

CR4.2 Los parámetros del sistema que afectan a la integración con dependencias externas en el proceso de compilación del código fuente se verifican, comprobando elementos tales como cobertura de código, pruebas de "software", análisis de seguridad, dependencias de librerías, entre otros.

CR4.3 El código fuente validado se publica en la rama estable, solucionando los conflictos que se notifiquen en el proceso, comprobando las fechas de modificación y contenidos modificados y cerrando el caso o incidencia en la herramienta de seguimiento.

Contexto profesional

Medios de producción

Conexión a la red. Equipamiento informático: componentes, periféricos, cableado y equipamiento para equipos portátiles, entre otros. Sistemas operativos. Navegadores. Lenguajes de "scripting". Lenguajes estructurados para automatizaciones. Lenguajes declarativos. Utilidades y aplicaciones incorporadas a

los sistemas operativos. Herramientas de comunicación y colaboración en equipo. Sistemas gestores de repositorios de código fuente. Servicios de transferencia de ficheros y conexión remota. Herramientas de copia de seguridad. Herramientas de gestión y control de cambios, incidencias y configuración. Aplicaciones de gestión de incidencias, código fuente, gestión de proyectos y comunicación/colaboración. Repositorio de artefactos/imágenes. Servidor de despliegues. Servidores de infraestructura.

Productos y resultados

Repositorios de código fuente del "software" y de los servicios asociados a las aplicaciones de los sistemas gestionados. Código fuente de integración y plantillas responsables de la creación de los servicios modificados. Servicios de comunicación y colaboración del grupo de personas del proyecto configurados. Resultado de los procesos de integración continua (CI) del código fuente de las aplicaciones desarrolladas validados.

Información utilizada o generada

Normas externas de trabajo (Normativa aplicable sobre prevención de riesgos laborales -ergonomía-; Normativa aplicable de protección de datos, propiedad intelectual e industrial). Normas internas de trabajo (guías de despliegue, instalación, configuración y actualización de los servicios de las aplicaciones desarrolladas; documentación de diseño y aprovisionamiento de los recursos; diseño y especificaciones de los servicios a desplegar y operar; plan de seguridad, operación y calidad; acuerdos de nivel de servicio -SLA-; documentación de configuración de sistemas y servicios; plan de pruebas e informe de fallos; especificaciones de la arquitectura de referencia de servicio corporativo; documentación asociada a los scripts desarrollados; documentación de las pruebas de funcionamiento de los servicios y aplicaciones desarrolladas). Documentación técnica (documentación técnica asociado a servicios informáticos; manuales de funcionamiento de las herramientas de gestión del ciclo de vida del "software").

UNIDAD DE COMPETENCIA 4

Definir el flujo de procesos ("pipeline") de despliegue continuo de contenedores

Nivel: *3*
Código: *UC2746_3*
Estado: *Tramitación BOE*

Realizaciones profesionales y criterios de realización

RP1: Crear el paquete de "software" que se va a desplegar, utilizando la versión estable del código fuente que indique el responsable de las versiones, según las necesidades de uso, directivas de calidad y seguridad de la entidad responsable del desarrollo, para facilitar su despliegue y permitir la trazabilidad del sistema.

CR1.1 El código fuente se obtiene de la rama de trabajo del repositorio, utilizando los procesos de acceso, gestión y trazabilidad definidos en la entidad responsable del desarrollo.

CR1.2 La calidad del código se valida, usando herramientas de comprobación semántica y sintáctica y de seguridad sobre el código desarrollado y librerías de terceros asociadas.

CR1.3 El paquete de "software" se crea, incluyendo los elementos requeridos tales como aplicaciones, librerías y/o "script" de instalación, entre otros, para un despliegue automático en cualquier entorno, utilizando herramientas de arquitectura, versionado, entornos y trazabilidad.

CR1.4 El paquete de "software" se comprueba, verificando que contiene los elementos, tales como versión anterior de la aplicación, los "script" de instalación y los "script" para el ajuste de datos, que permitan dar marcha atrás del proceso y actualizar el "software" a la versión anterior en caso de que haya algún problema durante la validación.

CR1.5 El paquete de "software" a desplegar se comprueba, verificando que incluye elementos para la ejecución de pruebas funcionales y no funcionales.

CR1.6 Los resultados de las pruebas de "software" a desplegarse se almacenan en el aplicativo que defina la entidad responsable del desarrollo para su posterior reutilización, seguimiento y cualquier actividad que pueda ser requerida por el responsable de versionado.

CR1.7 El paquete de "software" a desplegar se almacena en el aplicativo que defina la entidad responsable del desarrollo para su posterior reutilización, seguimiento y cualquier actividad que pueda ser requerida por el responsable de versionado.

RP2: Preparar el entorno mediante la validación, creación o modificación de las variables de entorno requeridas, para el despliegue del paquete creado para cada aplicativo o servicio.

CR2.1 La existencia de los parámetros requeridas para desplegar en cada entorno se valida de forma automática por el "pipeline", mediante configuración, según los procesos definidos en la creación de la infraestructura.

CR2.2 Los valores de los parámetros a utilizar en cada entorno se recopilan, obteniéndolos de la aplicación definida por la entidad responsable del desarrollo durante la creación de la infraestructura.

CR2.3 Los parámetros de los entornos recopilados se verifica que se han incluido en el "software" a desplegar, ejecutando el despliegue y comprobando la ausencia de errores, según los procesos definidos por la entidad responsable del desarrollo.

CR2.4 Los errores detectados durante el despliegue se recopilan, comunicándolos al desarrollador, deteniendo la "pipeline" y destruyendo todos los objetos intermedios creados hasta el instante de la ejecución.

RP3: Desplegar la nueva versión del "software" en el entorno definido por el responsable de versiones, utilizando el paquete creado por el "pipeline", para que se pueda validar antes de la puesta en funcionamiento.

CR3.1 Las aplicaciones adicionales relacionadas y previas al despliegue se instalan como parte del paquete o, en caso de ser algo estático, de modo que se pueda acceder al repositorio del "software" para proceder a su instalación.

CR3.2 La nueva versión del "software" y aquellos artefactos que se requieran para realizar las tareas de integración con otros sistemas se instalan, ejecutado "script" de validación de la instalación.

CR3.3 El "software" desplegado se comprueba, garantizando que se integra de manera automática con el resto de las aplicaciones de la solución, ejecutado los "script" de prueba que realizan la tarea.

RP4: Validar el nuevo "software" instalado, comprobando que cumple todos los requerimientos y directivas de la entidad responsable del desarrollo sobre pruebas no funcionales, funcionales y rendimiento, resolviendo los fallos detectados y actualizando los repositorios de versiones para garantizar un despliegue libre de errores.

CR4.1 El "software" se valida automáticamente, utilizando bien las herramientas definidas en el paquete o bien el "software" de pruebas en caso de que dicho "software" esté predefinido por la entidad responsable del desarrollo.

CR4.2 El "pipeline" se comprueba, verificando que accede a los flujos de trabajo y datos de prueba de cada uno de los entornos de ejecución, incluyendo pruebas no funcionales, funcionales, de rendimiento y de integración con otras aplicaciones relacionadas.

CR4.3 Los resultados de las pruebas se almacenan, guardándolos en las aplicaciones definidas por la entidad responsable del desarrollo, para su acceso y posterior uso en la toma de decisiones del responsable de versionado.

CR4.4 Los fallos de validación del nuevo código se resuelven, mediante la actualización del entorno con la versión estable anterior, efectuándola de manera automática tanto para código fuente como para datos.

CR4.5 El paquete de "software" se actualiza en el repositorio de versiones de los entornos, en caso de no detectarse fallos, incorporándolo según la operativa que disponga la herramienta de versiones, para que el responsable tenga el conocimiento de la situación de cada entorno.

CR4.6 Las dependencias entre aplicaciones y versiones se actualizan de manera automática, según la operativa que disponga la herramienta de versiones.

CR4.7 La información que produce el "software" desplegado se comprueba, garantizando que se envía al sistema de monitorización existente, revisándolo en el propio "software" de monitorización.

Contexto profesional

Medios de producción

Conexión a la red. Equipamiento informático: equipos, componentes, periféricos, cableado, entre otros. Sistemas operativos. Navegadores. Lenguajes de "scripting". Lenguajes estructurados para

automatizaciones. Lenguajes declarativos. Utilidades y aplicaciones incorporadas a los sistemas operativos. Herramientas de comunicación y colaboración en equipo. Sistemas gestores de repositorios de código fuente. Servicios de transferencia de ficheros y conexión remota. Herramientas de gestión y control de cambios, incidencias y configuración. Repositorio de artefactos/imágenes. Servidor de despliegues. Servidores de infraestructura. Sistema de monitorización.

Productos y resultados

Paquete de "software" que se va a desplegar, creado. Entorno de despliegue preparado. Nueva versión del "software" desplegada en el entorno. Nuevo "software" instalado y validado. Repositorios de versiones actualizados.

Información utilizada o generada

Normas externas de trabajo (Normativa aplicable sobre prevención de riesgos laborales -ergonomía-; Normativa aplicable de protección de datos, propiedad intelectual e industrial). Normas internas de trabajo (guías de despliegue, instalación, configuración y actualización de los servicios de las aplicaciones desarrolladas; documentación de diseño y aprovisionamiento de los recursos; diseño y especificaciones de los servicios a desplegar y operar; plan de seguridad, operación y calidad; acuerdos de nivel de servicio -SLA-; documentación de configuración de sistemas y servicios; plan de pruebas e informe de fallos; especificaciones de la arquitectura de referencia de servicio corporativo; documentación asociada a los scripts desarrollados; documentación de las pruebas de funcionamiento de los servicios y aplicaciones desarrolladas). Documentación técnica (documentación técnica asociado a servicios informáticos; manuales de funcionamiento de las herramientas de gestión del ciclo de vida del "software", bases de datos de vulnerabilidades).

UNIDAD DE COMPETENCIA 5

Mantener el sistema de contenedores desplegado

Nivel: *3*
Código: *UC2747_3*
Estado: *Tramitación BOE*

Realizaciones profesionales y criterios de realización

RP1: Integrar la conectividad a través de la red de datos entre el contenedor y los sistemas de monitorización y alarmas asociados, asegurando el flujo de información, cumpliendo las especificaciones y criterios de seguridad para permitir su interconexión.

CR1.1 La integración del contenedor con el sistema de monitorización se verifica, probando las comunicaciones y comprobando la integridad de los datos enviados y recibidos y que dichos datos son almacenados en el sistema remoto.

CR1.2 Los umbrales de los contadores y las cadenas de texto requeridas en los eventos se configuran, especificando valores recogidos en la documentación para generar diferentes tipos de indicadores.

CR1.3 Las reglas de agregado y correlación de contadores se configuran, aplicando la parametrización que figura en la documentación del proyecto, para crear nuevos indicadores.

CR1.4 Los eventos generados en el contenedor se integran en el sistema de gestión de alarmas comprobando su activación y/o recuperación, probando las comunicaciones y verificando que se almacenan en el sistema remoto para su gestión.

CR1.5 Los eventos recibidos en el sistema de gestión de alarmas se categorizan, agrupándolos usando parámetros tales como fecha de creación, origen, criticidad, entre otros, para la posterior notificación y tratamiento de la alarma.

CR1.6 Las comunicaciones entre el contenedor y los sistemas de monitorización y alarmas se auditan, verificando que sólo los protocolos y puertos requeridos para dicha comunicación están habilitados.

CR1.7 Las reglas de protección y seguridad entre el contenedor y el entorno de monitorización y alarmas se configuran, comprobando que no es posible otra comunicación distinta en dicho canal de comunicaciones.

RP2: Validar el sistema desplegado, comprobando su estado, indicadores y el rendimiento esperado, según especificaciones para monitorizar su funcionalidad y calidad de servicio.

CR2.1 Las métricas proporcionadas por la aplicación se documentan, explicando cada contador y su referencia asociada, verificando que se incluyen tanto indicadores de capacidad como de rendimiento y calidad.

CR2.2 Los indicadores se implementan, realizando las fórmulas y cálculos sobre los contadores proporcionados por el sistema, especificando los umbrales para cada tipo de indicador, documentando los valores máximos, mínimo y recomendado por el sistema.

CR2.3 Las posibles alarmas generadas por el sistema se documentan, indicando el origen, posible fallo e impacto en servicio derivado de cada alarma, adjuntando las guías para su manejo, indicando los pasos a seguir para el análisis y resolución de las mismas.

CR2.4 Las pruebas del aplicativo o componente se ejecutan, incluyendo pruebas funcionales, de aseguramiento de calidad del servicio, rendimiento y seguridad de la aplicación y de estrés, verificando que todos los parámetros están dentro de los umbrales marcados por las especificaciones.

CR2.5 Las pruebas ejecutadas se documentan, almacenando los resultados y evidencias de ejecución de cada caso, junto con los indicadores y archivos de registro.

CR2.6 Los indicadores y su evolución se monitorizan usando el sistema de monitorización, revisándolas periódicamente, comprobando que el rendimiento de la aplicación no se excede de los umbrales marcados y sus indicadores funcionan acorde con la carga de trabajo requerida a la aplicación.

RP3: Extraer datos e información, ejecutándolo manualmente o previa programación del proceso en su caso, para el control y toma de decisiones.

CR3.1 Los archivos de registro de la aplicación se observan periódicamente, chequeando que no existen errores internos derivados de algún posible fallo "software", salida inesperada de la aplicación o reinicio y reportando dichos fallos en caso de ocurrir.

CR3.2 Los accesos al sistema se monitorizan, detectando intentos con contraseña equivocada y alertando de conexiones por fuerza bruta, bloqueando los no permitidos para proteger la integridad de la aplicación.

CR3.3 Los datos generados por fallos o reinicios de la aplicación tales como volcados de memoria ("crashdumps"), registros de error ("log"), entre otros se revisan de forma periódica, monitorizando su aparición y reportando al diseñador del contenedor su frecuencia y contenido, para inspeccionar la aplicación y corregir el origen del fallo.

CR3.4 El rendimiento del equipo se comprueba, evaluando y comparando indicadores tales como uso de CPU, ocupación de memoria, acceso a disco y otros, para comprobar que se cumplen las especificaciones del proyecto.

RP4: Ejecutar procesos de "backup", programándolos y ejecutándolos para garantizar la integridad y disponibilidad de la aplicación.

CR4.1 Los "backup" de la aplicación se programan, configurándolos para que sean ejecutados periódicamente, verificando su ejecución para que se asegure la recuperación del sistema en caso de fallo.

CR4.2 Los ficheros de "backup" producidos por la aplicación se exportan a un medio externo, comprobando que son almacenados de acuerdo a las políticas de almacenamiento, rotación y limpieza establecidas en las especificaciones del proyecto.

CR4.3 El mantenimiento preventivo de los "backup" se realiza de forma periódica, asegurando que la última copia de seguridad puede ser restaurada, instanciando dicha copia en una plataforma de pruebas y verificando el despliegue del contenedor.

RP5: Instalar actualizaciones de nuevas versiones de "software", comprobando la existencia de parches de mantenimiento y de corrección de posibles vulnerabilidades para garantizar la seguridad y funcionalidad.

CR5.1 El "software" de base de la aplicación se mantiene, revisando en el repositorio la aparición de nuevas versiones que solucionen problemas encontrados durante las pruebas o reportados durante la vida de la aplicación e instalándolas en su caso.

CR5.2 Las vulnerabilidades sobre los componentes usados internamente por el contenedor o la aplicación se examinan, a partir de escaneados de código, binarios y librerías o buscando información en foros del programador del componente, estudiando su posible afectación y consecuencias e implementando posibles soluciones recomendadas si existieran.

CR5.3 La actualización de los paquetes de "software" se realiza de forma periódica, descargando de un repositorio propio o externo, instalando los cambios y verificando que no se produce pérdida de los datos previamente almacenados para asegurar la integridad de la aplicación y la disponibilidad de la información.

CR5.4 Los problemas que aparezcan durante el proceso de actualización "software" se resuelven en su caso, en entornos previos, realizando un análisis del problema, identificando su naturaleza, en los márgenes de tiempo y el nivel de calidad requerido y reportando al diseñador de la aplicación.

CR5.5 El funcionamiento de la aplicación una vez actualizada o corregida y la integridad de los datos se chequea, ejecutando las pruebas acordadas en entornos previos y producción, según las especificaciones del proyecto, para verificar su funcionamiento.

CR5.6 La actualización de las versiones "software" se documenta, actualizando el repositorio de incidencias, incluyendo información tal como el nombre y versión de la aplicación actualizada, información acerca de las incidencias generadas e incompatibilidades detectadas, incremento o decremento de rendimiento, garantizando la trazabilidad de los procesos, de cara a facilitar su seguimiento.

RP6: Ejecutar tareas de terminación del contenedor, eliminando programas y datos relacionados en condiciones de seguridad, para reutilizar el almacenamiento.

CR6.1 La aplicación "software" desplegada en contenedor se termina cuando no va a ser usada, verificando que todos los recursos usados por el contenedor en la infraestructura han quedado liberados, tales como CPU, memoria y espacio de almacenamiento en disco, entre otros.

CR6.2 Las copias de seguridad ("backup") y los archivos de registro almacenados en servidores externos se eliminan, sobreescribiendo el espacio de disco usado previamente o utilizando alguna técnica similar, asegurando que no es posible su restauración mediante técnicas de recuperación de datos.

CR6.3 Los datos en posibles bases de datos internas de la aplicación que contengan información confidencial o sensible, se borran destruyendo su contenido acorde con los procedimientos que garantizan la protección de datos personales.

CR6.4 La conectividad e integración del contenedor con el entorno de monitorización y alarmas se desconfigura, eliminando la conectividad a través de la red de datos y todas las referencias introducidas en los sistemas de monitorización y alarmas relativas a la aplicación "software" del contenedor.

Contexto profesional

Medios de producción

Conexión a la red. Equipamiento informático: equipos, componentes, periféricos, cableado, entre otros. Sistemas operativos. Navegadores. Lenguajes de "scripting". Lenguajes estructurados para automatizaciones. Lenguajes declarativos. Utilidades y aplicaciones incorporadas a los sistemas operativos. Herramientas de comunicación y colaboración en equipo. Sistemas gestores de repositorios de código fuente. Servicios de transferencia de ficheros y conexión remota. Herramientas de copia de seguridad. Herramientas de gestión y control de cambios, incidencias y configuración. Repositorio de artefactos/imágenes. Servidor de despliegues. Servidores de infraestructura. Sistema de monitorización.

Productos y resultados

Conectividad a través de la red de datos integrada. Sistema desplegado validado. Datos e información para el control y toma de decisiones extraídos. Procesos de "backup" programados y ejecutados. Actualizaciones de nuevas versiones de "software" instaladas. Tareas de terminación del contenedor ejecutadas.

Información utilizada o generada

Normas externas de trabajo (Normativa aplicable sobre prevención de riesgos laborales -ergonomía-; Normativa aplicable de protección de datos, propiedad intelectual e industrial). Normas internas de trabajo (guías de despliegue, instalación, configuración y actualización de los servicios de las aplicaciones desarrolladas; documentación de diseño y aprovisionamiento de los recursos; diseño y especificaciones de los servicios a desplegar y operar; plan de seguridad, operación y calidad; acuerdos de nivel de servicio -SLA-; documentación de configuración de sistemas y servicios; plan de pruebas e informe de fallos; especificaciones de la arquitectura de referencia de servicio corporativo; documentación asociada a los scripts desarrollados; documentación de las pruebas de funcionamiento de los servicios y aplicaciones desarrolladas). Documentación técnica (documentación técnica asociado a servicios informáticos; manuales de funcionamiento de las herramientas de gestión del ciclo de vida del "software", bases de datos de vulnerabilidades).

MÓDULO FORMATIVO 1

Preparación de entornos de despliegue en contenedores

Nivel: *3*
Código: *MF2743_3*
Asociado a la UC: *UC2743_3 - Preparar el entorno de despliegue en contenedores*
Duración (horas): *120*
Estado: *Tramitación BOE*

Capacidades y criterios de evaluación

C1: Aplicar procedimientos de creación del acceso a los repositorios de código de infraestructura y/o servicios, comprobando las herramientas de gestión y validación, para su uso en el desarrollo y despliegue, siguiendo estándares y políticas de seguridad, para facilitar a los usuarios y/o grupos de gestión el uso de los repositorios de código de infraestructura y/o servicios en las fases de desarrollo y despliegue.

CE1.1 Describir el proceso de despliegue en contenedores de aplicaciones, explicando el desarrollo e integración continuas (CI/CD) como parte de una metodología.

CE1.2 Enumerar herramientas relacionadas con el despliegue en contenedores, describiendo sus objetivos, características y funcionamiento.

CE1.3 Clasificar herramientas de validación de calidad y seguridad del código y dependencias de librerías externas o integraciones, describiendo sus características y aplicaciones.

CE1.4 Describir el proceso de habilitación del acceso a los recursos y herramientas para desplegar y/o gestionar los entornos de producción definidos en la arquitectura, definiendo usuarios y perfiles que permitan y limiten.

CE1.5 Explicar el procedimiento de configuración de los accesos a un código fuente para ser gestionados a través de herramientas, identificando los parámetros de seguridad y políticas de acceso, a través de los usuarios y/o grupos asignados.

CE1.6 En un supuesto práctico de aplicación de procedimientos de creación del acceso a los repositorios de código de infraestructura y/o servicios, comprobando las herramientas de gestión y validación, para su uso en el desarrollo y despliegue, siguiendo estándares y políticas de seguridad, para facilitar a los usuarios y/o grupos de gestión el uso de los repositorios de código de infraestructura y/o servicios en las fases de desarrollo y despliegue:

- Crear accesos a unos recursos y herramientas para desplegar y/o gestionar los entornos definidos en la arquitectura del proyecto, definiendo usuarios y roles, verificando su funcionalidad, para tener control de acceso y mantener el registro de acceso.
- Configurar accesos al código fuente se configuran para ser gestionados a través de las herramientas proporcionadas por la entidad responsable del desarrollo, parametrizando la seguridad y las políticas de acceso, a través de los usuarios y/o grupos asignados.
- Las herramientas de validación de calidad y seguridad del código y dependencias de librerías externas o integraciones se verifican, comprobando su funcionalidad para ser usadas por los usuarios y/o grupos de gestión.

C2: Aplicar técnicas de validación de la calidad y seguridad de unas modificaciones previas al despliegue, usando herramientas de gestión y validación, y siguiendo estándares y políticas de seguridad para detectar posibles fallos.

CE2.1 Describir el procedimiento de uso de una herramienta de validación del código fuente, clasificando las pruebas de calidad a ejecutar.

CE2.2 Explicar el procedimiento de uso de herramientas específicas de comprobación de parámetros de seguridad del código, identificando parámetros de configuración y soluciones a adoptar.

CE2.3 En un supuesto práctico de aplicación de técnicas de validación de la calidad y seguridad de unas modificaciones previas al despliegue, usando herramientas de gestión y validación y siguiendo estándares y políticas de seguridad para detectar posibles fallos:

- Validar unas modificaciones del código fuente, ejecutando pruebas de calidad mediante herramientas específicas, documentándolas con unas guías de desarrollo y flujos de trabajo y/o políticas.
- Ejecutar comprobaciones de parámetros de seguridad del código mediante herramientas específicas, de forma periódica, siguiendo las recomendaciones del fabricante.
- Validar dependencias del código de librerías externas o integraciones, mediante herramientas de comprobación, para verificar la seguridad y funcionalidad e integración con el resto de componentes del aplicativo antes del despliegue.

C3: Aplicar procedimientos de configuración de entornos de revisión, mediante ficheros de parámetros y creando flujos de trabajo, siguiendo estándares y políticas de seguridad para comprobar el despliegue.

CE3.1 Describir ficheros de definición de infraestructura como código (IaC - "Infraestructure as Code"), identificando contenidos y utilidad y explicando el procedimiento de creación usando el lenguaje propio de una plataforma de despliegue, previa identificación de los entornos de despliegue del código, artefactos e imágenes de contenedores.

CE3.2 Explicar el proceso de creación de flujos de trabajo para el despliegue del entorno "hardware", usando las credenciales proporcionadas, aplicando unas configuraciones definidas, para crear el entorno de desarrollo o producción.

CE3.3 Enumerar los elementos que se especifican al configurarlas definiciones técnicas de uso de control y de seguridad para cada entorno, tales como:

- Roles de acceso del usuario.
- Roles de acceso del código.
- Acceso al almacenamiento.
- Acceso a las API ("Application Programming Interface"), entre otros recursos, identificando parámetros de configuración.

CE3.4 Detallar el procedimiento de creación dentro de la plataforma de la infraestructura definida en los ficheros, usando las credenciales facilitadas, para su posterior integración en los flujos de trabajo.

CE3.5 En un supuesto práctico de aplicación de procedimientos de configuración de entornos de revisión, mediante ficheros de parámetros y creando flujos de trabajo, siguiendo estándares y políticas de seguridad para comprobar el despliegue:

- Crear unos ficheros de definición de infraestructura como código (IaC - "Infraestructure as Code"), usando el lenguaje propio de la plataforma de despliegue, previa identificación de los entornos de despliegue del código, artefactos e imágenes de contenedores.
- Crear unos flujos de trabajo para el despliegue de infraestructura, usando las credenciales proporcionadas, aplicando las configuraciones definidas, para crear el entorno.

- Las definiciones técnicas de uso de control y de seguridad se configuran para cada entorno, especificando: roles de acceso del usuario, roles de acceso del código, acceso al almacenamiento, acceso a las API ("Application Programming Interface"), entre otros recursos, para su posterior gestión.
- Crear la infraestructura definida en los ficheros al efecto dentro de la plataforma proporcionada por la entidad responsable del desarrollo, usando las credenciales facilitadas, para su posterior integración en los flujos de trabajo.
- Verificar la infraestructura creada después del despliegue, comprobando que se obtiene acceso a todos los recursos de infraestructura.

C4: Aplicar procedimientos de instalación del "software" de automatización y gestión de paquetes y dependencias y de administración de la configuración del "software", creando flujos de mantenimiento y despliegue sobre las plataformas, configurando los parámetros de uso para preparar la automatización del despliegue.

CE4.1 Enumerar herramientas de automatización y gestión de paquetes y dependencias, identificando sus características y funcionalidades.

CE4.2 Describir el procedimiento de instalación de las herramientas en la infraestructura seleccionada, actualizándolas según las directrices de versión y documentación del fabricante.

CE4.3 Explicar el proceso de creación de flujos para mantenimiento, usando las herramientas para la automatización de la administración de la configuración del "software" y servicios de los sistemas desarrollados.

CE4.4 Detallar el proceso de verificación de los flujos para mantenimiento, indicando cómo probar que las acciones automatizadas generan los resultados esperados.

CE4.5 En un supuesto práctico de aplicación de procedimientos de instalación del "software" de automatización y gestión de paquetes y dependencias y de administración de la configuración del "software", creando flujos de mantenimiento y despliegue sobre las plataformas, configurando los parámetros de uso para preparar la automatización del despliegue:
- Instalar unas herramientas de automatización y gestión de paquetes y dependencias en la infraestructura seleccionada, actualizándolas según las directrices de versión y documentación del fabricante.
- Crear los flujos para mantenimiento, usando las herramientas para la automatización de la administración de la configuración del "software" y servicios de los sistemas desarrollados.
- Verificar los flujos para mantenimiento, probando que las acciones automatizadas generan los resultados esperados según la documentación del proyecto.

C5: Aplicar procedimientos de definición de las variables de trabajo de un entorno, configurándolas para ser usadas por los contenedores que tienen el código y que se crean de forma dinámica y para ejecutar la aplicación.

CE5.1 Definir procedimientos de parametrización de reglas de seguridad, niveles de servicio y consumo, identificando su aplicación como configuraciones dentro de las herramientas de despliegue.

CE5.2 Enumerar métricas disponibles en una plataforma, clasificándolas y definiendo su utilidad, explicando el proceso para incluirlas en las herramientas de monitorización.

CE5.3 Describir el contenido de posibles informes de rendimiento y planificación ("capacity planning"), indicando la información a mostrar, para anticipar el crecimiento y facilitar la gestión de los recursos que requieren las herramientas.

CE5.4 Explicar el proceso de generación de acciones automáticas basadas en eventos disparados por errores, describiendo cómo se incorporan en las herramientas desplegadas, para ajustar el rendimiento y mantener los recursos utilizados en niveles óptimos.

CE5.5 En un supuesto práctico de aplicación de procedimientos de definición de las variables de trabajo de un entorno, configurándolas para ser usadas por los contenedores que tienen el código y que se crean de forma dinámica y para ejecutar la aplicación:

- Aplicar reglas de seguridad, niveles de servicio y consumo como configuraciones dentro de unas herramientas de despliegue, para parametrizar la ejecución de los contenedores.
- Incluir en las herramientas de monitorización las métricas para elaborar informes de rendimiento y planificación ("capacity planning"), anticipar el crecimiento y facilitar la gestión de los recursos que requieren las herramientas, previa recopilación y usando las instrucciones y facilidades de la herramienta.
- Redactar la documentación sobre el uso de los despliegues de código e imágenes de contenedores, para facilitar la reutilización de los componentes ya preparados, almacenándola en un repositorio de código para servir como guía o modelo para posteriores desarrollos y despliegues.
- Incorporar en las herramientas desplegadas las acciones automáticas basadas en eventos disparados por errores tales como fallos de trabajo en los flujos, disponibilidad de componentes de integración, fallos de acceso, estado de finalización, entre otros, usando las facilidades de cada herramienta, para ajustar el rendimiento y mantener los recursos utilizados en niveles óptimos.

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

C1 respecto a CE1.6; C2 respecto a CE2.3; C3 respecto a CE3.5; C4 respecto a CE4.5 y C5 respecto a CE5.5.

Otras Capacidades:

Comunicarse eficazmente con las personas adecuadas en cada momento, respetando los canales establecidos en la organización.
Adaptarse a la organización, a sus cambios organizativos y tecnológicos, así como a situaciones o contextos nuevos.
Mantener una actitud asertiva, empática y conciliadora con las personas demostrando cordialidad y amabilidad en el trato.
Mostrar iniciativa en la búsqueda de soluciones y en la resolución de problemas.
Aplicar de forma efectiva el principio de igualdad de trato y no discriminación en las condiciones de trabajo entre mujeres y hombres.

Contenidos

1 Metodologías de integración y despliegue continuos (DevOps) aplicadas a la preparación del entorno

Metodologías de desarrollo: Cascada ("Waterfall") vs ágiles (Agile). Ciclo de vida del desarrollo.
Herramientas de gestión de proyectos de desarrollo.
Integración Continua (CI).
Despliegue continuo (CD).

2 Acceso a repositorios de código de infraestructura y/o servicios

Despliegue de aplicaciones en contenedores de aplicaciones. Desarrollo e integración continuas (CI/CD).
Herramientas relacionadas con el despliegue en contenedores. Objetivos, características y funcionamiento.
Herramientas de validación de calidad y seguridad del código y dependencias de librerías externas o integraciones. Características y aplicaciones.
Habilitación del acceso a los recursos y herramientas para desplegar y/o gestionar los entornos de producción. Usuarios y perfiles.
Accesos a un código fuente a través de herramientas. Configuración. Parámetros de seguridad y políticas de acceso.

3 Validación de la calidad y seguridad de modificaciones previas despliegues

Herramientas de validación del código fuente. Pruebas de calidad.
Herramientas específicas de comprobación de parámetros de seguridad del código. Parámetros de configuración. Soluciones a adoptar.

4 Configuración de entornos de revisión

Ficheros de definición de infraestructura como código. Contenidos y utilidad. Procedimiento de creación usando el lenguaje propio de una plataforma de despliegue. Identificación de los entornos de despliegue del código, artefactos e imágenes de contenedores.
Creación de flujos de trabajo. Configuraciones para crear un entorno.
Procedimientos de configuración de las definiciones técnicas de uso de control y de seguridad para un entorno: roles de acceso de usuarios, roles de acceso del código, acceso al almacenamiento, acceso a las API ("Application Programming Interface"). Parámetros de configuración.
Creación de la infraestructura definida en ficheros dentro de una plataforma. Integración en los flujos de trabajo.

5 Instalación del "software" de automatización y gestión de paquetes y dependencias y de administración de la configuración del "software"

Herramientas de automatización y gestión de paquetes y dependencias. Características y funcionalidades. Instalación en una infraestructura.
Creación de flujos para mantenimiento. Herramientas para la automatización de la administración de la configuración del "software" y servicios de los sistemas desarrollados.
Pruebas de verificación de los flujos para mantenimiento.

6 Definición de variables de trabajo de un entorno

Reglas de seguridad, niveles de servicio y consumo. Parametrización y aplicación como configuraciones dentro de las herramientas de despliegue.
Métricas de monitorización. Tipos y utilidad. Uso en herramientas de monitorización.
Generación de acciones automáticas basadas en eventos disparados por errores. Tipos de error (fallos de trabajo en los flujos, disponibilidad de componentes de integración, fallos de acceso, estado de finalización, entre otros). Incorporación en herramientas desplegadas.

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 laborales, accesibilidad universal, igualdad de género y protección medioambiental. Se considerará con carácter orientativo como espacios de uso:

- Instalación de 2 m² por alumno o alumna.

Perfil profesional del formador o formadora:

1. Dominio de los conocimientos y las técnicas relacionados con la preparación de entornos de despliegue en contenedores, que se acreditará simultáneamente mediante las dos formas siguientes:
- Formación académica de nivel 2 (Marco Español de Cualificaciones para la Educación Superior) o de otras de superior nivel relacionadas con el campo profesional.
- Experiencia profesional de un mínimo de 3 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

Despliegue de plataformas de ejecución de contenedores

Nivel: *3*
Código: *MF2744_3*
Asociado a la UC: *UC2744_3 - Desplegar la plataforma de ejecución de contenedores*
Duración (horas): *150*
Estado: *Tramitación BOE*

Capacidades y criterios de evaluación

C1: Aplicar procedimientos de creación de la nube virtual privada ("Virtual Private Cloud" -VPC-) donde se desplegará una infraestructura, mediante un hipervisor, configurando comunicaciones y seguridad, para permitir la gestión de la virtualización en todos sus niveles.

CE1.1 Clasificar hipervisores, describiendo sus características y funcionalidades.

CE1.2 Describir el proceso de instalación de un hipervisor, identificando las opciones que permiten la virtualización de sistemas, teniendo en cuenta sus limitaciones, sistemas operativos base, sistemas operativos soportados y capacidades de configuración.

CE1.3 Describir las aplicaciones y características de una VPC (Red virtual Privada), identificando sus funcionalidades.

CE1.4 Explicar el proceso de creación de una VPC, usando la consola de un hipervisor, diferenciando las VPC de los entornos, realizando aislamientos y capacidades de configuración.

CE1.5 Identificar parámetros tales como IP, zona de disponibilidad en su caso y DNS, explicando cómo configurarlos para establecer comunicaciones entre las zonas de un mismo entorno.

CE1.6 Explicar el nivel de exposición DMZ externa, DMZ interna y entorno interno, describiendo el proceso de configuración de capas de seguridad en función de la zona de exposición y accesos externos.

CE1.7 Describir el procedimiento de ejecución de pruebas de seguridad, aislamiento y separación de clientes, garantizando que se cumplen unos requisitos y corrigiendo la configuración en su caso para alcanzarlos.

CE1.8 En un supuesto práctico de aplicación de procedimientos de creación de la VPC donde se desplegará una infraestructura, mediante un hipervisor, configurando comunicaciones y seguridad, para permitir la gestión de la virtualización en todos sus niveles:

- Instalar un hipervisor, configurándolo con las opciones que permiten la virtualización de sistemas, teniendo en cuenta sus limitaciones y capacidades de configuración.
- Crear una VPC, usando la consola del hipervisor, diferenciando las VPC de los entornos, realizando aislamientos y capacidades de configuración.
- Asignar parámetros tales como IP, zona de disponibilidad en su caso y DNS, configurando las comunicaciones entre las zonas de un mismo entorno.
- Configurar la seguridad se configura, desplegando capas de seguridad dependiendo de la exposición DMZ externa, DMZ interna y entorno interno y ejecutando pruebas que la garanticen, asegurando el aislamiento y separación de clientes.

C2: Aplicar procedimientos de creación de una infraestructura como código (IaC), seleccionando la infraestructura a desplegar y creando los códigos de despliegue, para permitir la automatización mediante procesos y adaptar las aplicaciones y servicios con mayor rapidez.

CE2.1 Enumerar elementos de una infraestructura a desplegar tal como almacenamiento, capacidad de cómputo, memoria, autoescalado, entre otros, identificando los valores que se ajustan a unos requerimientos en función del servicio que va a alojar.

CE2.2 Identificar paradigma, estructuras de control y sentencias del código usado por la plataforma para la automatización del despliegue de la infraestructura, describiendo el proceso de programación usando un editor, en base a unos requerimientos del servicio a alojar.

CE2.3 Explicar el proceso de lanzamiento de pruebas de servicio y chequeo de configuración de un código para el despliegue de la infraestructura, identificando resultados y soluciones de corrección en su caso.

CE2.4 Describir el procedimiento de verificación de las configuraciones de seguridad, explicando los criterios para comprobar si el despliegue automatizado de IaC cumple unas recomendaciones de seguridad en el entorno, normativa interna/externa y recomendaciones de organismos internacionales (guías de configuración de entornos).

CE2.5 En un supuesto práctico de aplicación de procedimientos de creación de una infraestructura como código, seleccionando la infraestructura a desplegar y creando los códigos de despliegue, para permitir la automatización mediante procesos y adaptar las aplicaciones y servicios con mayor rapidez:

- Seleccionar una infraestructura a desplegar tal como almacenamiento, capacidad de cómputo, memoria, autoescalado, entre otros, ajustada a unos requerimientos para el servicio que va a alojar.
- Crear un código que automatice el despliegue de la infraestructura seleccionada, usando un editor y el lenguaje fijado por la plataforma, en base a los requerimientos del servicio a alojar.
- Ejecutar el código para el despliegue de la infraestructura, lanzando pruebas de servicio y chequeo de configuración.
- Verificar las configuraciones de seguridad, comprobando si el despliegue automatizado de IaC cumple con unas recomendaciones de seguridad en el entorno.

C3: Aplicar procedimientos para desplegar el orquestador, configurando un proyecto y tareas y creando el "pipeline", para ayudar a realizar las funciones de integración continua y la automatización cuando un evento lo indica.

CE3.1 Enumerar herramientas "software" de orquestador, identificando sus capacidades para llevar a cabo las pruebas de control de las variables o limitaciones del CI.

CE3.2 Describir el proceso de selección e instalación de un "software" de orquestador, identificando los parámetros de configuración tales como permisos, accesos, certificados y opciones de despliegue, entre otras.

CE3.3 Identificar "plugin" o complementos del orquestador para incorporar las necesidades de integración del orquestador con las tecnologías del entorno, explicando el proceso de instalación y configuración.

CE3.4 Explicar el proceso de creación de un proyecto, identificando las configuraciones base del orquestador de tareas, así como las credenciales para el acceso al repositorio.

CE3.5 Describir el proceso de creación de tareas a realizar dentro de un proyecto, identificando parámetros de configuración tales como nombre de la tarea, máquinas en las que se procesan los trabajos del despliegue, direccionamiento, y condiciones de ejecución del despliegue tales como aseguramiento de la calidad y seguridad.

CE3.6 Detallar el procedimiento de creación de un "pipeline", identificando los parámetros para encadenar procesos de modo que el resultado del primero desencadene una acción en el nido de trabajos definidos dentro de un proyecto, definiendo las etapas, secuencias y "script" que marquen los estados del proyecto.

CE3.7 Explicar el proceso de configuración de la seguridad, describiendo los pasos para asignar los permisos que garanticen la ejecución de las tareas planificadas siguiendo el principio de "mínimo privilegio".

CE3.8 En un supuesto práctico de aplicación de procedimientos para desplegar el orquestador, configurando un proyecto y tareas y creando el "pipeline", para ayudar a realizar las funciones de integración continua y la automatización cuando un evento lo indica:

- Seleccionar un "software" de orquestador, escogiendo aquel con capacidad de llevar a cabo las pruebas de control de las variables o limitaciones del CI.
- Instalar y/o configurar el "software" de orquestador seleccionado, parametrizando permisos, accesos, certificados y opciones de despliegue, entre otras.
- Instalar unos "plugin" o complementos del orquestador, configurándolos para incorporar las necesidades de integración del orquestador con las tecnologías del entorno.
- Crear un proyecto, definiendo las configuraciones base del orquestador de tareas, así como las credenciales para el acceso al repositorio.
- Crear unas tareas a realizar dentro del proyecto, parametrizando las opciones tales como nombre de la tarea, máquinas en las que se procesan los trabajos del despliegue, direccionamiento, y condiciones de ejecución del despliegue tales como aseguramiento de la calidad y seguridad.
- Crear un "pipeline", parametrizándolo para encadenar procesos de modo que el resultado del primero desencadene una acción en el nido de trabajos definidos dentro de un proyecto, definiendo las etapas, secuencias y "script" que marquen los estados del proyecto.
- Probar el "pipeline" ejecutando los caminos posibles dentro del mismo, verificando que se gestionan los errores y salidas no esperadas.
- Configurar la seguridad, asignando los permisos que garanticen la ejecución de las tareas planificadas siguiendo el principio de "mínimo privilegio".

C4: Aplicar procedimientos de despliegue de la monitorización para un entorno, seleccionando y configurando el gestor de ingesta de datos y la seguridad para facilitar el control y gestión de problemas.

CE4.1 Enumerar gestores de ingesta de datos, describiendo las opciones de recolección de datos y posterior visualización de estos.

CE4.2 Explicar el proceso de configuración de un "software" de gestión de datos y métricas, describiendo cómo seleccionar las métricas a mostrar e identificando parámetros de configuración de paneles, alertas y envíos de los indicadores clave de rendimiento ("Key Performance Indicator" -KPI-).

CE4.3 Describir el proceso de configuración de la seguridad, identificando parámetros para habilitar la comunicación en tránsito segura, accesos y repositorios de datos encriptados.

CE4.4 En un supuesto práctico de aplicar procedimientos de despliegue de la monitorización para un entorno, seleccionando y configurando el gestor de ingesta de datos y la seguridad para facilitar el control y gestión de problemas:

- Seleccionar un gestor de ingesta de datos, escogiendo un "software" con capacidad de recolección de datos y posterior visualización de estos.
- Configurar un "software" de gestión de datos y métricas, seleccionando el visualizador de métricas, configurando los paneles, alertas y envíos de los indicadores clave de rendimiento ("Key Performance Indicator" -KPI-).

- Configurar la seguridad, parametrizando comunicaciones, accesos y repositorios de datos.

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

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

Otras Capacidades:

Comunicarse eficazmente con las personas adecuadas en cada momento, respetando los canales establecidos en la organización.
Adaptarse a la organización, a sus cambios organizativos y tecnológicos, así como a situaciones o contextos nuevos.
Mantener una actitud asertiva, empática y conciliadora con las personas demostrando cordialidad y amabilidad en el trato.
Mostrar iniciativa en la búsqueda de soluciones y en la resolución de problemas.
Aplicar de forma efectiva el principio de igualdad de trato y no discriminación en las condiciones de trabajo entre mujeres y hombres.

Contenidos

1 Metodologías de integración y despliegue continuos (DevOps) aplicadas al despliegue de plataformas de ejecución de contenedores

Metodologías de desarrollo: Cascada ("Waterfall"), ágiles (Agile). Ciclo de vida del desarrollo.
Herramientas de gestión de proyectos de desarrollo.
Integración Continua (CI).
Despliegue Continuo (CD).

2 Creación de nubes virtuales privadas ("Virtual Private Cloud" -VPC-) para el despliegue de infraestructuras en contenedores

Hipervisores. Clasificación, características y funcionalidades. Instalación.
Consola del hipervisor. Creación de VPC.
Parámetros de comunicaciones: (IP, zona de disponibilidad en su caso y DNS).
Configuración de capas de seguridad en función del nivel de exposición (DMZ externa, DMZ interna y entorno interno).
Pruebas de seguridad, aislamiento y separación de clientes.

3 Creación de una infraestructura como código (IaC) para la automatización del despliegue en contenedores

Elementos de infraestructura a desplegar: almacenamiento, capacidad de cómputo, memoria, autoescalado, entre otros.
Automatización del despliegue de la infraestructura. Lenguajes de codificación ("script") de la plataforma.
Pruebas de servicio y chequeo de configuración.
Verificación de la seguridad. Recomendaciones de organismos internacionales. Normativa aplicable.

4 Despliegue del orquestador del despliegue en contenedores

Herramientas "software" de orquestador. Capacidades para llevar a cabo las pruebas de control de las variables o limitaciones del CI. Instalación y configuración. Parámetros (permisos, accesos, certificados y opciones de despliegue, entre otras).

"Plugin" o complementos del orquestador. Configuración.
Creación de proyectos. Configuraciones base del orquestador de tareas. Credenciales para el acceso al repositorio.
Creación de tareas a realizar dentro de proyectos. Parámetros de configuración.
Creación de "pipeline". Etapas, secuencias y "script" que marcan los estados del proyecto.
Configuración de la seguridad. Permisos. Principio de "mínimo privilegio".

5 Despliegue de la monitorización para un entorno

Gestores de ingesta de datos. Opciones de recolección de datos y posterior visualización.
"Software" de gestión de datos y métricas. Configuración. Parámetros de configuración de paneles, alertas y envíos de los indicadores clave de rendimiento ("Key Performance Indicator" -KPI-).
Configuración de la seguridad en comunicaciones, accesos y repositorios de datos encriptados.

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 laborales, accesibilidad universal, igualdad de género y protección medioambiental. Se considerará con carácter orientativo como espacios de uso:

- Instalación de 2 m² por alumno o alumna.

Perfil profesional del formador o formadora:

1. Dominio de los conocimientos y las técnicas relacionados con el despliegue de plataformas de ejecución de contenedores, que se acreditará simultáneamente mediante las dos formas siguientes:
- Formación académica de nivel 2 (Marco Español de Cualificaciones para la Educación Superior) o de otras de superior nivel relacionadas con el campo profesional.
- Experiencia profesional de un mínimo de 3 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

Definición de flujos de procesos ("pipelines") del desarrollador en integración continua

Nivel:	*3*
Código:	*MF2745_3*
Asociado a la UC:	*UC2745_3 - Definir el flujo de procesos ("pipeline") del desarrollador en integración continua*
Duración (horas):	*180*
Estado:	*Tramitación BOE*

Capacidades y criterios de evaluación

C1: Aplicar técnicas de gestión de un repositorio de código fuente del "software" y de los servicios asociados a las aplicaciones de los sistemas, según las necesidades de uso, en condiciones de calidad y seguridad, para facilitar su mantenimiento, recuperación y permitir la trazabilidad del sistema.

CE1.1 Enumerar repositorios de código, describiendo las funcionalidades que proveen para gestionar los cambios en un proyecto software y los interfaces de uso.

CE1.2 Describir procedimientos de organización de un repositorio de código fuente, explicando cómo organizar los orígenes de modo que se permita su uso de forma consistente, definiendo ramas de código estable y validado y otras donde se recojan los cambios que están en proceso.

CE1.3 Explicar el proceso de definición y configuración de parámetros de un sistema que afectan a la autenticación y autorización, ajustándolos a las necesidades de acceso, integración con herramientas y seguridad.

CE1.4 Clasificar procedimientos de validación de un código fuente, para evaluar las modificaciones de manera automatizada, comprobando la sintaxis y la semántica del código, versiones de librerías externas y/o genéricas, en base a las guías de desarrollo y los flujos de trabajo y unas políticas tales como aprobación, asignación o revisión, entre otras.

CE1.5 Identificar mecanismos de comprobación de la seguridad de un código, verificando mediante "software" específico que no contenga código malicioso y que no contenga vulnerabilidades.

CE1.6 Aplicar procesos de copia de seguridad y recuperación del código fuente, programándolos para su ejecución de forma periódica, gestionando repositorios de gran tamaño.

CE1.7 Explicar el proceso de publicación del código fuente validado en una rama estable, solucionando los conflictos que se notifiquen en el proceso, comprobando las fechas de modificación y contenidos modificados.

CE1.8 En un supuesto practico de aplicación de técnicas de gestión de un repositorio de código fuente del "software" y de los servicios asociados a las aplicaciones de los sistemas, según las necesidades de uso, en condiciones de calidad y seguridad, para facilitar su mantenimiento, recuperación y permitir la trazabilidad del sistema:

- Organizar unos orígenes de código fuente con una estructura que permita su uso de forma consistente, definiendo ramas de código estable y validado, y otras donde se recojan los cambios que están en proceso.

- Configurar parámetros del sistema que afectan a la autenticación y autorización, ajustándolos a las necesidades de acceso, integración con herramientas y seguridad.
- Validar unas modificaciones sobre el código fuente, evaluando de manera automatizada la sintaxis y la semántica del código, comprobando versiones de librerías externas y/o genéricas, siguiendo las guías de desarrollo y los flujos de trabajo y unas políticas tales como aprobación, asignación o revisión, entre otras.
- Comprobar la seguridad del código, verificando mediante "software" específico que no contenga código malicioso y que no contenga vulnerabilidades.
- Publicar el código fuente validado en la rama estable, solucionando los conflictos que se notifiquen en el proceso, comprobando las fechas de modificación y contenidos modificados.
- Ejecutar un proceso de copia de seguridad y recuperación del código fuente, programándolo para aplicarlo de forma periódica, gestionando repositorios de gran tamaño.

C2: Aplicar técnicas de modificación del código fuente de integración y plantillas responsables de la creación de los servicios, definiendo los parámetros de los artefactos, en condiciones de calidad y seguridad, para simplificar la operación y la integración.

CE2.1 Explicar el procedimiento de creación de servicios requeridos para las aplicaciones de forma automatizada, describiendo cómo modificarlos, en su caso, empleando línea de comandos (CLI), API ("Application Programming Interface") y/o, automatismos mediante lenguajes de programación, entre otras.

CE2.2 Clasificar los parámetros para el automatismo del ciclo de vida, diferenciándolos en función de los objetivos.

CE2.3 Describir el proceso de definición de parámetros de los artefactos para el automatismo del ciclo de vida de los servicios, considerando características propias del despliegue de las versiones de los datos de las aplicaciones, tales como creación de bases de datos, movimiento o transformación de la información y metadatos, entre otras.

CE2.4 Detallar el proceso de definición de parámetros de los artefactos para el automatismo del ciclo de vida de los servicios relacionados con las aplicaciones, considerando características propias de la integración de las versiones del "software", tales como la gestión de la configuración de las aplicaciones, entre otras.

CE2.5 Enumerar parámetros de los artefactos para el automatismo del ciclo de vida de los servicios relacionados con infraestructura, explicando cómo se definen, considerando características propias de la integración de las versiones del código fuente de las aplicaciones, tales como contenedores, máquinas virtuales, máquinas físicas, scripts, código binario, entre otros.

CE2.6 Enumerar parámetros de los artefactos para el automatismo del ciclo de vida de los servicios, describiendo cómo se definen, considerando elementos que permitan su reutilización en futuros despliegues, tales como nombre del servicio, región geográfica, recursos asignados, permisos, confirmando que son únicos en su caso.

CE2.7 En un supuesto práctico de aplicación de técnicas de modificación del código fuente de integración y plantillas responsables de la creación de los servicios, en condiciones cumpliendo directivas de operación y en condiciones de calidad y seguridad, para simplificar la operación y la integración:
- Crear los servicios requeridos para unas aplicaciones de forma automatizada, modificándolos, en su caso, empleando línea de comandos (CLI), API ("Application Programming Interface"), automatismos mediante lenguajes de programación, entre otras.
- Definir los parámetros de los artefactos para el automatismo del ciclo de vida de los servicios, considerando características propias del despliegue de las versiones de los datos de las

aplicaciones, tales como creación de bases de datos, movimiento o transformación de la información y metadatos, entre otras.
- Definir los parámetros de los artefactos para el automatismo del ciclo de vida de los servicios relacionados con las aplicaciones, considerando características propias de la integración de las versiones del "software", tales como la gestión de la configuración de las aplicaciones, entre otras.
- Definir los parámetros de los artefactos para el automatismo del ciclo de vida de los servicios relacionados con infraestructura, considerando características propias de la integración de las versiones del código fuente de las aplicaciones, tales como contenedores, máquinas virtuales, máquinas físicas, scripts, código binario, entre otros.
- Definir los parámetros de los artefactos para el automatismo del ciclo de vida de los servicios, considerando elementos que permitan su reutilización en futuros despliegues, tales como nombre del servicio, región geográfica, recursos asignados, permisos, confirmando que son únicos en su caso.
- Verificar que el código fuente de la integración, plantillas declarativas del servicio o cualquier proceso responsable de esta tarea sea idempotente, siendo robusta su ejecución y proporcionando predictibilidad bajo circunstancias variables.

C3: Aplicar procedimientos de configuración de los servicios de comunicación y colaboración del grupo de personas de un proyecto según unas necesidades de uso, directivas de comunicación y adopción, para automatizar las interacciones con los repositorios de código fuente y las herramientas de gestión de proyectos.

CE3.1 Enumerar plataformas de comunicación y herramientas de gestión de proyectos, describiendo cómo configurar los repositorios de código fuente de modo que permitan la recepción automática de cambios de estado y contenido.

CE3.2 Describir procedimientos de configuración de unas plataformas de comunicación para notificaciones acerca de métricas, alertas o reglas definidas en los repositorios de código fuente, estados de tareas, peticiones de cambios al sistema, entre otras.

CE3.3 Explicar el proceso de configuración de unas plataformas de comunicación, documentación y herramientas de gestión de proyectos, describiendo cómo conectarlas con los repositorios de código fuente, de tal modo que permitan relacionar los errores ("bugs") con modificaciones de código fuente, entre otras.

CE3.4 En un supuesto práctico de aplicación de procedimientos de configuración de los servicios de comunicación y colaboración del grupo de personas de un proyecto según unas necesidades de uso, directivas de comunicación y adopción, para automatizar las interacciones con los repositorios de código fuente y las herramientas de gestión de proyectos:
- Emplear unas plataformas de comunicación y herramientas de gestión de proyectos, configurando los repositorios de código fuente de modo que permitan la recepción automática de cambios de estado y contenido.
- Configurar las plataformas de comunicación empleadas para notificar acerca de métricas, alertas o reglas definidas en los repositorios de código fuente, estados de tareas, peticiones de cambios al sistema, entre otras.
- Configurar las plataformas de comunicación, documentación y herramientas de gestión de proyectos empleadas, conectándolas con los repositorios de código fuente, de tal modo que permitan relacionar errores ("bugs") con modificaciones de código fuente, entre otras.

C4: Aplicar procedimientos de validación del resultado de los procesos de integración continua (CI) del código fuente de las aplicaciones desarrolladas, dentro del

marco de unas directivas sobre operación, calidad y seguridad para su publicación.

CE4.1 Enumerar posibles fallos de ejecución, calidad, seguridad y rendimiento de las aplicaciones del sistema, proponiendo soluciones que las resuelvan.

CE4.2 Describir procedimientos de ejecución de pruebas de diagnóstico con unas herramientas integradas, proporcionando información sobre resultados y acciones a los fallos diagnosticados.

CE4.3 Enumerar elementos que afectan a la integración con dependencias externas en el proceso de compilación del código fuente, tales como cobertura de código, pruebas de "software", análisis de seguridad, dependencias de librerías, entre otros.

CE4.4 Definir el proceso de verificación de los parámetros del sistema que afectan a la integración con dependencias externas en el proceso de compilación del código fuente, explicando los pasos para comprobar elementos que afectan en este ámbito, tales como cobertura de código, pruebas de "software", análisis de seguridad, dependencias de librerías, entre otros.

CE4.5 En un supuesto práctico de aplicación de procedimientos de validación del resultado de los procesos de integración continua (CI) del código fuente de las aplicaciones desarrolladas, dentro del marco de unas directivas sobre operación, calidad y seguridad:

- Resolver unos fallos de ejecución, calidad, seguridad y rendimiento de las aplicaciones del sistema mediante automatización, incluyendo las pruebas de diagnóstico con las herramientas integradas, proporcionando información sobre resultados y acciones a los fallos diagnosticados.
- Verificar parámetros del sistema que afectan a la integración con dependencias externas en el proceso de compilación del código fuente, comprobando elementos tales como cobertura de código, pruebas de "software", análisis de seguridad, dependencias de librerías, entre otros.
- Publicar el código fuente validado en la rama estable, solucionando los conflictos que se notifiquen en el proceso, comprobando las fechas de modificación y contenidos modificados.

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

C1 respecto a CE1.8; C2 respecto a CE2.7; C3 respecto a CE3.4 y C4 respecto a CE4.5.

Otras Capacidades:

Comunicarse eficazmente con las personas adecuadas en cada momento, respetando los canales establecidos en la organización.

Adaptarse a la organización, a sus cambios organizativos y tecnológicos, así como a situaciones o contextos nuevos.

Mantener una actitud asertiva, empática y conciliadora con las personas demostrando cordialidad y amabilidad en el trato.

Mostrar iniciativa en la búsqueda de soluciones y en la resolución de problemas.

Aplicar de forma efectiva el principio de igualdad de trato y no discriminación en las condiciones de trabajo entre mujeres y hombres.

Contenidos

1 Metodologías de integración y despliegue continuos (DevOps) en el flujo del desarrollador

Metodologías de desarrollo: Cascada ("Waterfall"), ágiles (Agile). Ciclo de vida del desarrollo.

Herramientas de gestión de proyectos de desarrollo.

Integración Continua (CI).

2 Gestión de repositorios de código fuente del "software" y de los servicios asociados a las aplicaciones de los sistemas

Repositorios de código. Herramientas. Características y funcionalidades.
Organización de un repositorio de código fuente. Fuentes/orígenes de código. Gestión de cambios: ramas o bifurcaciones.
Autenticación y autorización. Parámetros. Configuración. Integración.
Validación automatizada del código fuente. Comprobaciones de sintaxis y semántica del código. Comprobaciones de versiones de librerías externas y/o genéricas.
Comprobación de la seguridad del código. Herramientas "software" externas.
Procedimientos de copia de seguridad y recuperación del código fuente.
Publicación del código fuente validado. Solución de conflictos en el proceso.

3 Técnicas de modificación del código fuente de integración y plantillas responsables de la creación de los servicios

Creación de servicios de forma automatizada para las aplicaciones. Línea de comandos (CLI), API ("Application Programming Interface"), automatismos mediante lenguajes de programación.
Clasificación de parámetros para el automatismo del ciclo de vida, diferenciándolos en función de los objetivos.
Automatismo del ciclo de vida de los servicios. Parámetros de los artefactos. Características propias del despliegue de las versiones de los datos de las aplicaciones: creación de bases de datos, movimiento o transformación de la información y metadatos, entre otras.
Automatismo del ciclo de vida de los servicios relacionados con las aplicaciones. Parámetros de los artefactos. Características propias de la integración de las versiones del "software": gestión de la configuración de las aplicaciones, entre otras.
Automatismo del ciclo de vida de los servicios relacionados con infraestructura. Parámetros de los artefactos. Características propias de la integración de las versiones del código fuente de las aplicaciones: contenedores, máquinas virtuales, máquinas físicas, scripts, código binario, entre otros.
Automatismo del ciclo de vida de los servicios. Parámetros de los artefactos. Elementos que permiten su reutilización: nombre del servicio, región geográfica, recursos asignados, permisos.

4 Configuración de los servicios de comunicación y colaboración

Plataformas de comunicación y herramientas de gestión de proyectos. Configuración de repositorios de código fuente para la recepción automática de cambios de estado y contenido.
Configuración de plataformas de comunicación para notificaciones. Métricas, alertas o reglas en los repositorios de código fuente, estados de tareas, peticiones de cambios al sistema, entre otras.
Configuración de plataformas de comunicación, documentación y herramientas de gestión de proyectos. Conexión con los repositorios de código fuente.

5 Validación del resultado de los procesos de integración continua (CI) del código fuente

Fallos de ejecución, calidad, seguridad y rendimiento de las aplicaciones del sistema. Herramientas integradas. Diagnóstico y soluciones.
Integración con dependencias externas en el proceso de compilación del código fuente. Cobertura de código, pruebas de "software", análisis de seguridad, dependencias de librerías, entre otros. Parámetros del sistema que afectan a la integración.

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 laborales, accesibilidad universal, igualdad de género y protección medioambiental. Se considerará con carácter orientativo como espacios de uso:

- Instalación de 2 m² por alumno o alumna.

Perfil profesional del formador o formadora:

1. Dominio de los conocimientos y las técnicas relacionados con la definición de flujos de procesos ("pipelines") del desarrollador en integración continua, que se acreditará simultáneamente mediante las dos formas siguientes:
- Formación académica de nivel 2 (Marco Español de Cualificaciones para la Educación Superior) 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 4

Definición de flujos de procesos ("pipelines") de despliegue continuo de contenedores

Nivel: *3*
Código: *MF2746_3*
Asociado a la UC: *UC2746_3 - Definir el flujo de procesos ("pipeline") de despliegue continuo de contenedores*
Duración (horas): *180*
Estado: *Tramitación BOE*

Capacidades y criterios de evaluación

C1: Aplicar procedimientos de creación de un paquete de "software" que se va a desplegar, utilizando una versión estable del código fuente, según las necesidades de uso y en condiciones de calidad y seguridad, para facilitar su despliegue y permitir la trazabilidad del sistema.

CE1.1 Enumerar herramientas de comprobación de la calidad del código fuente, describiendo las capacidades de validación sintáctica y semántica y de seguridad sobre el código desarrollado y librerías de terceros asociadas.

CE1.2 Describir herramientas de arquitectura para crear paquetes de "software", explicando la manera de incluir elementos requeridos tales como aplicaciones, librerías y/o "script" de instalación, entre otros, para un despliegue automático en cualquier entorno.

CE1.3 Detallar procedimientos de comprobación del contenido de un paquete "software", incluyendo los pasos a seguir para verificar que contiene los elementos requeridos, tales como versión anterior de la aplicación, los "script" de instalación y los "script" para el ajuste de datos, que permitan dar marcha atrás del proceso y actualizar el "software" a la versión anterior en caso de que haya algún problema durante la validación.

CE1.4 Clasificar pruebas sobre un paquete a desplegar, explicando el proceso de comprobación según el tipo (pruebas funcionales y no funcionales), indicando los pasos para verificar que incluye elementos para la ejecución de dichas pruebas.

CE1.5 Describir procedimientos para almacenamiento de resultados de pruebas sobre un paquete a desplegar, describiendo su procedimiento de uso, para la posterior reutilización, seguimiento y cualquier otra actividad relacionada con el versionado.

CE1.6 Enumerar aplicativos o soluciones de almacenamiento y recuperación de paquetes "software" en aplicativos, explicando cómo proceder en ambos casos, para su posterior reutilización, seguimiento y cualquier actividad que pueda ser requerida por el responsable de versionado.

CE1.7 En un supuesto práctico de aplicación de procedimientos de creación de un paquete de "software" que se va a desplegar, utilizando una versión estable del código fuente, según las necesidades de uso y en condiciones de calidad y seguridad, para facilitar su despliegue y permitir la trazabilidad del sistema:

- Obtener un código fuente de la rama de trabajo del repositorio, utilizando procesos de acceso, gestión y trazabilidad.

- Validar la calidad del código, usando herramientas de comprobación semántica y sintáctica y de seguridad sobre el código desarrollado y librerías de terceros asociadas.
- Crear el paquete de "software", incluyendo todos los elementos requeridos tales como aplicaciones, librerías y/o 'script' de instalación, entre otros, para un despliegue automático en cualquier entorno, utilizando herramientas de arquitectura, versionado, entornos y trazabilidad.
- Comprobar el paquete de "software", verificando que contiene los elementos, tales como versión anterior de la aplicación, los "script" de instalación y los "script" para el ajuste de datos, que permitan dar marcha atrás del proceso y actualizar el "software" a la versión anterior en caso de que haya algún problema durante la validación.
- Comprobar el paquete de "software" a desplegar, verificando que incluye elementos para la ejecución de pruebas funcionales y no funcionales.
- Almacenar los resultados de las pruebas de "software" a desplegarse en un aplicativo, para su posterior reutilización, seguimiento y cualquier actividad relacionada con el versionado.
- Almacenar el paquete de "software" a desplegar en un aplicativo para su posterior reutilización, seguimiento y cualquier actividad relacionada con el versionado.

C2: Aplicar técnicas de validación, creación o modificación de las variables de entorno requeridas, para preparar un entorno de despliegue del paquete creado para cada aplicativo o servicio.

CE2.1 Describir procedimientos de validación automática por el "pipeline" de la existencia de los parámetros requeridos para desplegar en cada entorno, explicando su configuración, según los procesos definidos en la creación de la infraestructura.

CE2.2 Enumerar los posibles errores que puedan producirse durante el despliegue, explicando cómo detener la "pipeline" y revertir el proceso.

CE2.3 En un supuesto práctico de aplicación de técnicas de validación, creación o modificación de las variables de entorno requeridas, para preparar un entorno de despliegue del paquete creado para cada aplicativo o servicio:
- Validar la existencia de unos parámetros requeridos para desplegar en cada entorno de forma automática por el "pipeline", mediante configuración, según los procesos definidos en la creación de la infraestructura.
- Recopilar valores de los parámetros a utilizar en cada entorno, obteniéndolos de la aplicación definida durante la creación de la infraestructura.
- Verificar que los parámetros de los entornos recopilados se han incluido en el "software" a desplegar, ejecutando el despliegue y comprobando la ausencia de errores.

C3: Aplicar procedimientos para desplegar una nueva versión del "software" en un entorno, utilizando el paquete creado por el "pipeline", para que se pueda validar antes de la puesta en funcionamiento.

CE3.1 Describir procedimientos de instalación de aplicaciones adicionales relacionadas y previas al despliegue como parte de un paquete o, en caso de ser algo estático, de modo que se pueda acceder al repositorio del "software" para proceder a su instalación.

CE3.2 Explicar el proceso para la instalación de una nueva versión del "software" y aquellos artefactos que se requieran para realizar las tareas de integración con otros sistemas, ejecutado "script" de validación de la instalación.

CE3.3 Detallar el procedimiento para las pruebas de integración, tal como ejecutar "script" de prueba, para comprobar que el "software" desplegado se integra de manera automática con el resto de las aplicaciones de la solución.

CE3.4 Enumerar los posibles errores que puedan producirse durante el despliegue, explicando cómo detener la "pipeline" y revertir el proceso.

CE3.5 En un supuesto práctico de aplicación de procedimientos para desplegar una nueva versión del "software" en un entorno, utilizando el paquete creado por el "pipeline", para que se pueda validar antes de la puesta en funcionamiento:

- Instalar unas aplicaciones adicionales relacionadas y previas al despliegue como parte del paquete o, en caso de ser algo estático, de modo que se pueda acceder al repositorio del "software" para proceder a su instalación.
- Instalar una nueva versión del "software" y aquellos artefactos que se requieran para realizar las tareas de integración con otros sistemas, ejecutado "script" de validación de la instalación.
- Comprobar que el "software" desplegado se integra de manera automática con el resto de las aplicaciones de la solución, ejecutado "script" de prueba que realicen la tarea.
- Recopilar los errores detectados durante el despliegue, deteniendo la "pipeline" y destruyendo todos los objetos intermedios creados hasta el instante de la ejecución.

C4: Aplicar procedimientos de validación de un nuevo "software" instalado, comprobando que cumple todos los requerimientos, efectuando pruebas no funcionales, funcionales y rendimiento, resolviendo los fallos detectados y actualizando los repositorios de versiones para garantizar un despliegue libre de errores.

CE4.1 Enumerar herramientas definidas en el paquete para pruebas y validación automatizada del "software", explicando su configuración y gestión.

CE4.2 Enumerar herramientas externas para pruebas y validación automatizada del "software", explicando su configuración y gestión.

CE4.3 Reconocer la tipología de pruebas de verificación del "pipeline", incluyendo pruebas no funcionales, funcionales, de rendimiento y de integración con otras aplicaciones relacionadas, explicando los pasos para comprobar en cada caso que accede a los flujos de trabajo y datos de prueba de cada uno de los entornos de ejecución.

CE4.4 Describir procedimientos para almacenamiento de resultados de pruebas sobre un "software" instalado, describiendo su utilización, para su acceso y posterior uso en la toma de decisiones del responsable de versionado.

CE4.5 Explicar el proceso de actualización o retorno a la versión estable anterior para aplicarla en casos de fallos de validación del nuevo código, describiendo cómo efectuarla de manera automática tanto para código fuente como para los datos.

CE4.6 Detallar el proceso de verificación de la monitorización del sistema desplegado, explicando el proceso para comprobar que la información que produce el "software" es supervisada.

CE4.7 En un supuesto práctico de aplicación de procedimientos de validación de un nuevo "software" instalado, comprobando que cumple todos los requerimientos, efectuando pruebas no funcionales, funcionales y rendimiento, resolviendo los fallos detectados y actualizando los repositorios de versiones para garantizar un despliegue libre de errores:

- Validar un "software" automáticamente, utilizando bien las herramientas definidas en el paquete o bien un "software" de pruebas.
- Comprobar el "pipeline", verificando que accede a los flujos de trabajo y datos de prueba de cada uno de los entornos de ejecución, incluyendo pruebas no funcionales, funcionales, de rendimiento y de integración con otras aplicaciones relacionadas.
- Almacenar los resultados de las pruebas realizadas, guardándolos en las aplicaciones al efecto, para su acceso y posterior uso según necesidades del versionado.

- Resolver los fallos de validación del nuevo código, mediante la actualización del entorno con la versión estable anterior, efectuándola de manera automática tanto para código fuente como para datos.
- Actualizar el paquete de "software" creado en el repositorio de versiones, en caso de no detectarse fallos, incorporándolo según la operativa de versiones.
- Actualizar las dependencias entre aplicaciones y versiones de manera automática según la operativa que disponga la herramienta de versiones.
- Comprobar que la información que produce el "software" desplegado se envía al sistema de monitorización existente, revisándolo en el propio "software" de monitorización.

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

C1 respecto a CE1.7; C2 respecto a CE2.3; C3 respecto a CE3.5 y C4 respecto a CE4.7.

Otras Capacidades:

Comunicarse eficazmente con las personas adecuadas en cada momento, respetando los canales establecidos en la organización.
Adaptarse a la organización, a sus cambios organizativos y tecnológicos, así como a situaciones o contextos nuevos.
Mantener una actitud asertiva, empática y conciliadora con las personas demostrando cordialidad y amabilidad en el trato.
Mostrar iniciativa en la búsqueda de soluciones y en la resolución de problemas.
Aplicar de forma efectiva el principio de igualdad de trato y no discriminación en las condiciones de trabajo entre mujeres y hombres.

Contenidos

1 Metodologías aplicadas al despliegue continuo (DevOps)

Metodologías de desarrollo: Cascada ("Waterfall"), ágiles (Agile). Ciclo de vida del desarrollo.
Herramientas de gestión de proyectos de desarrollo.
Despliegue Continuo (CD).

2 Creación de paquetes de "software" para desplegar

Herramientas de comprobación de la calidad del código fuente. Validación sintáctica y semántica y de seguridad.
Herramientas de arquitectura para crear paquetes de "software" y despliegue automático. Pruebas de contenido del paquete y marcha atrás del proceso.
Tipología de pruebas sobre un paquete a desplegar. Pruebas funcionales y no funcionales. Aplicativos para almacenamiento de resultados de pruebas.
Aplicativos y procedimientos de almacenamiento y recuperación de paquetes "software" en aplicativos de versionado.

3 Preparación del entorno de despliegue

Validación, creación o modificación de las variables de entorno requeridas según cada aplicativo o servicio.
Validación automática de la existencia de los parámetros requeridos por el "pipeline". Configuración. Recopilación de valores de parámetros.
Detección y resolución de errores. Reversión del proceso y retorno a la versión anterior.

4 Despliegue de nuevas versiones del "software"

Aplicaciones adicionales relacionadas y previas al despliegue. Instalación.
Instalación de nueva versión del "software" y artefactos de integración con otros sistemas. Validación de la instalación.
Pruebas de integración.

5 Validación del nuevo "software" instalado

Herramientas definidas en el paquete para pruebas y validación automatizada. Configuración y gestión.
Herramientas externas para pruebas y validación automatizada del "software". Configuración y gestión.
Pruebas de verificación del "pipeline". Tipología. Pruebas no funcionales, funcionales, de rendimiento y de integración con otras aplicaciones.
Aplicativos y procedimientos para almacenamiento de resultados de pruebas.
Actualización de versión y retorno automatizado a la versión estable anterior en casos de fallo.
Verificación de la monitorización del sistema desplegado.

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 laborales, accesibilidad universal, igualdad de género y protección medioambiental. Se considerará con carácter orientativo como espacios de uso:

- Instalación de 2 m² por alumno o alumna.

Perfil profesional del formador o formadora:

1. Dominio de los conocimientos y las técnicas relacionados con la definición de flujos de procesos ("pipelines") de despliegue continuo de contenedores, que se acreditará simultáneamente mediante las dos formas siguientes:
- Formación académica de nivel 2 (Marco Español de Cualificaciones para la Educación Superior) 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 5

Mantenimiento del sistema de contenedores desplegado

Nivel: *3*
Código: *MF2747_3*
Asociado a la UC: *UC2747_3 - Mantener el sistema de contenedores desplegado*
Duración (horas): *120*
Estado: *Tramitación BOE*

Capacidades y criterios de evaluación

C1: Aplicar técnicas de integración de la conectividad a través de la red de datos entre el contenedor y los sistemas de monitorización y alarmas asociados, asegurando el flujo de información, en condiciones de seguridad para permitir su interconexión.

CE1.1 Reconocer sistemas de monitorización de contenedores desplegados, identificando tipos de indicadores, eventos, alarmas y reglas de tratamiento.

CE1.2 Describir procedimientos de verificación de la integración del contenedor con el sistema de monitorización, explicando los pasos para probar las comunicaciones y comprobar la integridad de los datos enviados y recibidos y que dichos datos son almacenados en el sistema remoto.

CE1.3 Explicar el proceso de configuración de umbrales de los contadores y las cadenas de texto requeridas en los eventos, describiendo cómo especificar los valores recogidos en la documentación para generar diferentes tipos de indicadores.

CE1.4 Identificar reglas de agregado y correlación de contadores, describiendo cómo se configuran, aplicando la parametrización que figura en la documentación del proyecto, para crear nuevos indicadores.

CE1.5 Describir el procedimiento de integración de eventos generados en un contenedor en el sistema de gestión de alarmas comprobando su activación y/o recuperación, probando las comunicaciones y verificando que se almacenan en el sistema remoto para su gestión.

CE1.6 Clasificar eventos recibidos en un sistema de gestión de alarmas, agrupándolos usando parámetros tales como fecha de creación, origen, criticidad, entre otros, para la posterior notificación y tratamiento de la alarma.

CE1.7 Explicar el proceso de auditoría de las comunicaciones entre un contenedor y los sistemas de monitorización y alarmas, describiendo los pasos para verificar que sólo los protocolos y puertos requeridos para dicha comunicación están habilitados.

CE1.8 Relacionar reglas de protección y seguridad entre el contenedor y el entorno de monitorización y alarmas, explicando el proceso de configuración y comprobación de que no es posible otra comunicación distinta en dicho canal de comunicaciones.

CE1.9 En un supuesto práctico de aplicación de técnicas de integración de la conectividad a través de la red de datos entre el contenedor y los sistemas de monitorización y alarmas asociados, asegurando el flujo de información, en condiciones de seguridad para permitir su interconexión:

- Verificar la integración de un contenedor con el sistema de monitorización, probando las comunicaciones y comprobando que los datos enviados y recibidos con correctos y que dichos datos son almacenados en el sistema remoto.
- Configurar los umbrales de los contadores y las cadenas de texto requeridas en los eventos, especificando valores recogidos en la documentación para generar diferentes tipos de indicadores.
- Configurar reglas de agregado y correlación de contadores, aplicando una parametrización, para crear nuevos indicadores.
- Integrar eventos generados en el contenedor en el sistema de gestión de alarmas comprobando su activación y/o recuperación, probando las comunicaciones y verificando que se almacenan en el sistema remoto para su gestión.
- Categorizar los eventos recibidos en el sistema de gestión de alarmas, agrupándolos usando parámetros tales como fecha de creación, origen, criticidad, entre otros, para la posterior notificación y tratamiento de la alarma.
- Auditar las comunicaciones entre el contenedor y los sistemas de monitorización y alarmas se auditan, verificando que sólo los protocolos y puertos requeridos para dicha comunicación están habilitados.
- Configurar unas reglas de protección y seguridad entre el contenedor y el entorno de monitorización y alarmas, comprobando que no es posible otra comunicación distinta en dicho canal de comunicaciones.

C2: Aplicar técnicas de validación de un sistema desplegado, comprobando su estado, indicadores y el rendimiento esperado, para monitorizar su funcionalidad y calidad de servicio.

CE2.1 Clasificar métricas de validación de un sistema desplegado, identificando objetivos, incluyendo tanto indicadores de capacidad como de rendimiento y calidad.

CE2.2 Identificar métricas proporcionadas por una aplicación, explicando cada contador y su referencia asociada.

CE2.3 Describir el proceso de implementación de indicadores, explicando las fórmulas y cálculos sobre los contadores proporcionados por el sistema, especificando los umbrales para cada tipo de indicador, documentando los valores máximos, mínimo y recomendado por el sistema.

CE2.4 Reconocer posibles alarmas generadas por un sistema, identificando origen, posible fallo e impacto en servicio derivado de cada una, indicando los pasos a seguir para el análisis y resolución de las mismas.

CE2.5 Clasificar pruebas de validación del sistema, tales como pruebas funcionales, de aseguramiento de calidad del servicio, rendimiento y seguridad de la aplicación y de estrés, explicando cómo verificar que los parámetros están dentro de los umbrales marcados por unas especificaciones.

CE2.6 En un supuesto práctico de aplicación de técnicas de validación de un sistema desplegado, comprobando su estado, indicadores y el rendimiento esperado, para monitorizar su funcionalidad y calidad de servicio:
- Documentar métricas proporcionadas por una aplicación, explicando cada contador y su referencia asociada, verificando que se incluyen tanto indicadores de capacidad como de rendimiento y calidad.
- Implementar indicadores, realizando las fórmulas y cálculos sobre los contadores proporcionados por el sistema, especificando los umbrales para cada tipo de indicador, documentando los valores máximos, mínimo y recomendado por el sistema.

- Documentar las posibles alarmas generadas por el sistema, indicando el origen, posible fallo e impacto en servicio derivado de cada alarma, adjuntando las guías para su manejo, indicando los pasos a seguir para el análisis y resolución de las mismas.
- Ejecutar pruebas del aplicativo o componente, incluyendo pruebas funcionales, de aseguramiento de calidad del servicio, rendimiento y seguridad de la aplicación y de estrés, verificando que todos los parámetros están dentro de unos umbrales marcados.
- Documentar las pruebas, almacenando los resultados y evidencias de ejecución de cada caso, junto con los indicadores y archivos de registro.
- Monitorizar los indicadores y su evolución usando el sistema de monitorización, comprobando que el rendimiento de la aplicación no se excede de los umbrales marcados y sus indicadores funcionan acorde con la carga de trabajo requerida a la aplicación.

C3: Aplicar técnicas de extracción de datos e información, ejecutándolas manualmente o previa programación del proceso en su caso, para el control y toma de decisiones.

CE3.1 Clasificar archivos y mecanismos de registro de una aplicación, describiendo sus objetivos, para aplicar en la detección de errores y fallos, tales como "log" de ejecución, accesos, volcados de memoria ("crashdumps") y "log" de errores, entre otros.

CE3.2 Describir la estructura y contenidos de los archivos de registro de una aplicación, para su uso en la comprobación de errores provocados por fallos de software, salida inesperada de la aplicación o reinicio.

CE3.3 Explicar el proceso de monitorización de accesos a un sistema, describiendo cómo detectar intentos con contraseña equivocada y alertando de conexiones por fuerza bruta y cómo bloquear los no permitidos para proteger la integridad de la aplicación.

CE3.4 Describir herramientas de monitorización del rendimiento de un equipo, explicando su uso en la comprobación, evaluación y comparación de indicadores tales como uso de CPU, ocupación de memoria, acceso a disco y otros.

CE3.5 En un supuesto práctico de aplicación de técnicas de extracción de datos e información, ejecutándolas manualmente o previa programación del proceso en su caso, para el control y toma de decisiones:

- Comprobar que no existen errores internos derivados de algún posible fallo "software", salida inesperada de la aplicación o reinicio de una aplicación, observando sus archivos de registro.
- Monitorizar accesos al sistema, detectando intentos con contraseña equivocada y alertando de conexiones por fuerza bruta, bloqueando los no permitidos para proteger la integridad de la aplicación.
- Monitorizar la aparición de fallos o reinicios de la aplicación tales como volcados de memoria ("crashdumps"), registros de error ("log"), entre otros, recogiendo los datos generados, anotando su frecuencia y contenido, para inspeccionar la aplicación y corregir el origen del fallo.
- Comprobar el rendimiento del equipo, evaluando y comparando indicadores tales como uso de CPU, ocupación de memoria, acceso a disco y otros, para comprobar que se cumplen unas especificaciones.

C4: Aplicar procedimientos de "backup", programándolos y ejecutándolos para garantizar la integridad y disponibilidad de la aplicación.

CE4.1 Diferenciar los tipos de copia de seguridad disponibles, valorando las situaciones en que es aplicable cada opción.

CE4.2 Explicar procedimientos de "backup" de una aplicación, indicando cómo programarlos y configurarlos para que sean ejecutados periódicamente, describiendo los pasos de verificación de su ejecución para que se asegure la recuperación del sistema en caso de fallo.

CE4.3 En un supuesto práctico de aplicación de procedimientos de "backup", programándolos y ejecutándolos para garantizar la integridad y disponibilidad de la aplicación:
- Programar los "backup" de una aplicación, configurándolos para que sean ejecutados periódicamente, verificando su ejecución para que se asegure la recuperación del sistema en caso de fallo.
- Exportar los ficheros de "backup" producidos por la aplicación a un medio externo, comprobando que son almacenados de acuerdo a las políticas de almacenamiento, rotación y limpieza establecidas en las especificaciones del proyecto.
- Asegurar que la última copia de seguridad puede ser restaurada, instanciando dicha copia en una plataforma de pruebas y verificando el despliegue del contenedor.

C5: Aplicar procedimientos de instalación de actualizaciones de nuevas versiones de "software", comprobando la existencia de parches de mantenimiento y de corrección de posibles vulnerabilidades para garantizar la seguridad y funcionalidad.

CE5.1 Identificar repositorios del "software" en uso, reconociendo los mecanismos de aviso de aparición de nuevas versiones que solucionen problemas encontrados.

CE5.2 Clasificar herramientas de escaneado de código, binarios y librerías "software" para la detección de vulnerabilidades, explicando los pasos para aplicar soluciones en función de los defectos detectados.

CE5.3 Localizar foros de programadores de componentes que informen de vulnerabilidades y otros defectos, describiendo la aplicación de soluciones en su caso.

CE5.4 Identificar mecanismos de comprobación de la integridad de la aplicación y la disponibilidad de la información aplicables en actualizaciones, diferenciando herramientas y su uso.

CE5.5 En un supuesto práctico de aplicación de procedimientos de instalación de actualizaciones de nuevas versiones de "software", comprobando la existencia de parches de mantenimiento y de corrección de posibles vulnerabilidades para garantizar la seguridad y funcionalidad:
- Mantener el "software" de base de una aplicación, revisando en el repositorio la aparición de nuevas versiones que solucionen problemas encontrados durante las pruebas o reportados durante la vida de la aplicación e instalándolas en su caso.
- Examinar vulnerabilidades sobre los componentes usados internamente por un contenedor o una aplicación, a partir de escaneados de código, binarios y librerías o buscando información en foros del programador del componente, estudiando su posible afectación y consecuencias e implementando posibles soluciones recomendadas si existieran.
- Realizar una actualización de los paquetes de "software" programada de forma periódica, descargando de un repositorio propio o externo, instalando los cambios y verificando que no se produce pérdida de los datos previamente almacenados para asegurar la integridad de la aplicación y la disponibilidad de la información.
- Resolver en su caso los problemas que aparezcan durante el proceso de actualización "software", analizando el problema e identificando su naturaleza.
- Comprobar el funcionamiento de la aplicación una vez actualizada o corregida y la integridad de los datos, ejecutando pruebas para verificar su funcionamiento.
- Documentar la actualización de las versiones "software", actualizando un repositorio de incidencias, incluyendo información tal como el nombre y versión de la aplicación actualizada, información acerca de las incidencias generadas e incompatibilidades detectadas, incremento o decremento de rendimiento, garantizando la trazabilidad de los procesos, de cara a facilitar su seguimiento.

C6: Aplicar procedimientos de terminación de un contenedor, eliminando programas y datos relacionados en condiciones de seguridad, para reutilizar el almacenamiento.

CE6.1 Reconocer herramientas de visualización y control de recursos usados por el contenedor en la infraestructura, tales como CPU, memoria y espacio de almacenamiento en disco, entre otros, describiendo su acceso y uso para identificar que han quedado liberados tras la terminación del contenedor.

CE6.2 Clasificar herramientas de borrado de datos mediante sobreescritura en el espacio de disco usado previamente u otra técnica similar, explicando su uso para asegurar que no es posible su restauración mediante técnicas de recuperación de datos.

CE6.3 En un supuesto práctico de aplicación de procedimientos de terminación de un contenedor, eliminando programas y datos relacionados en condiciones de seguridad, para reutilizar el almacenamiento:

- Terminar definitivamente una aplicación "software" desplegada en contenedor, verificando que todos los recursos usados por el contenedor en la infraestructura han quedado liberados, tales como CPU, memoria y espacio de almacenamiento en disco, entre otros.
- Eliminar copias de seguridad ("backup") y archivos de registro almacenados en servidores externos, sobreescribiendo el espacio de disco usado previamente o utilizando alguna técnica similar, asegurando que no es posible su restauración mediante técnicas de recuperación de datos.
- Borrar datos en bases de datos internas de la aplicación que contengan información confidencial o sensible, destruyendo su contenido acorde con los procedimientos que garantizan la protección de datos personales.
- Desconfigurar la conectividad e integración del contenedor con un entorno de monitorización y alarmas, eliminando la conectividad a través de la red de datos y todas las referencias introducidas en los sistemas de monitorización y alarmas relativas a la aplicación "software" del contenedor.

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

C1 respecto a CE1.9; C2 respecto a CE2.6; C3 respecto a CE3.5; C4 respecto a CE4.3; C5 respecto a CE5.5 y C6 respecto a CE6.3.

Otras Capacidades:

Comunicarse eficazmente con las personas adecuadas en cada momento, respetando los canales establecidos en la organización.
Adaptarse a la organización, a sus cambios organizativos y tecnológicos, así como a situaciones o contextos nuevos.
Mantener una actitud asertiva, empática y conciliadora con las personas demostrando cordialidad y amabilidad en el trato.
Mostrar iniciativa en la búsqueda de soluciones y en la resolución de problemas.
Aplicar de forma efectiva el principio de igualdad de trato y no discriminación en las condiciones de trabajo entre mujeres y hombres.

Contenidos

1 Integración de contenedores y sistemas de monitorización

Sistemas de monitorización de contenedores. Tipos de indicadores, eventos, alarmas y reglas de tratamiento.
Verificación de la integración de contenedores y sistemas de monitorización. Pruebas.
Umbrales de los contadores y cadenas de texto en eventos. Configuración.
Reglas de agregado y correlación de contadores.
Eventos recibidos de sistemas de gestión de alarmas. Parámetros (fecha de creación, origen, criticidad, entre otros). Notificación y tratamiento de alarmas.
Auditoría de comunicaciones entre contenedores y sistemas de monitorización y alarmas. Análisis de protocolos y puertos.
Protección y seguridad del entorno de monitorización y alarmas.

2 Monitorización de sistemas desplegados

Métricas de validación. Objetivos. Indicadores de capacidad, rendimiento y calidad.
Implementación de indicadores. Fórmulas y cálculos sobre los contadores proporcionados. Umbrales para cada tipo de indicador.
Gestión de alarmas generadas. Identificación de origen del fallo e impacto en servicio. Análisis y resolución de los fallos.
Pruebas de validación del sistema. Pruebas funcionales, de aseguramiento de calidad del servicio, de rendimiento, de seguridad de la aplicación y de estrés.

3 Extracción de datos e información para el control del sistema desplegado

Archivos y mecanismos de registro de una aplicación para detección de fallos. "Log" de ejecución, accesos, volcados de memoria ("crashdumps") y "log" de errores, entre otros. Estructura y contenidos.
Monitorización de accesos al sistema. Detección de intentos con contraseña equivocada. Detección de conexiones por fuerza bruta. Bloqueo de accesos no permitidos.
Monitorización del rendimiento de un equipo. Herramientas. Indicadores (uso de CPU, ocupación de memoria, acceso a disco y otros).

4 Mantenimiento de sistemas de contenedores

Copias de seguridad. Tipos. Programación periódica. Verificación de la copia.
Actualizaciones del "software" en uso. Repositorios. Detección de nuevas versiones.
Vulnerabilidades. Foros de programadores de componentes. Escaneado de código, binarios y librerías "software". Herramientas. Aplicación de soluciones.
Mecanismos de comprobación de la integridad y disponibilidad de aplicaciones e información: herramientas.
Terminación de contenedores. Herramientas de borrado de datos mediante sobreescritura en disco similares. Liberación de recursos.

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 laborales, accesibilidad universal, igualdad de género y protección medioambiental. Se considerará con carácter orientativo como espacios de uso:

- Instalación de 2 m² por alumno o alumna.

Perfil profesional del formador o formadora:

1. Dominio de los conocimientos y las técnicas relacionados con el mantenimiento del sistema de contenedores desplegado, que se acreditará simultáneamente mediante las dos formas siguientes:
- Formación académica de nivel 2 (Marco Español de Cualificaciones para la Educación Superior) o de otras de superior nivel relacionadas con el campo profesional.
- Experiencia profesional de un mínimo de 3 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.