



# AZ-400T00

## Designing and Implementing Microsoft DevOps solutions



### Información general.

Este curso proporciona el conocimiento y las habilidades para diseñar e implementar procesos y prácticas de DevOps. Los alumnos descubrirán cómo planificar DevOps, usar el control de origen, escalar Git para una empresa, consolidar artefactos, diseñar una estrategia de gestión de dependencias, administrar secretos, implementar una integración continua, implementar una estrategia de construcción de contenedores, diseñar una estrategia de lanzamiento, configurar un flujo de trabajo de gestión de lanzamiento, implementar un patrón de implementación y optimizar los mecanismos de realizar comentarios.

### Duración.

5 Días (KeD).

### Perfil del público.

Estudiantes de este curso están interesados en diseñar e implementar procesos de DevOps o en aprobar el examen de certificación de Microsoft Azure DevOps Solutions.

### Examen.

AZ-400: Designing and Implementing Microsoft DevOps Solutions.

### Temario.

#### **Ruta de aprendizaje: Desarrollo para DevOps empresarial.**

Esta ruta de aprendizaje examina las características clave de Git que las organizaciones deben planear al diseñar sus productos DevOps de tipo empresarial.

### Módulo 1: Introducción a DevOps.

Este módulo explora las áreas clave que las organizaciones deben aplicar para comenzar su recorrido de transformación de DevOps, cambiar la mentalidad del equipo y definir escalas de tiempo y objetivos.

- ¿Qué es DevOps?
- Explorar el recorrido de DevOps.
- Identificación de los equipos de transformación.
- Exploración de objetivos compartidos y definición de escalas de tiempo.

### Módulo 2: Planeación de Agile con proyectos de GitHub y Azure Boards.

En este módulo se presentan proyectos de GitHub, paneles de proyectos de GitHub y Azure Boards. Además, se analizan formas de vincular Azure Boards y GitHub, de configurar proyectos de GitHub y vistas de proyecto, y de administrar el trabajo con proyectos de GitHub.

- Introducción a los proyectos de GitHub y a los paneles de proyectos.
- Introducción a Azure Boards.
- Configuración de proyectos y equipos en Azure DevOps.
- Vinculación de GitHub a Azure Boards.
- Configuración de proyectos de GitHub.
- Administración del trabajo con los paneles de proyectos de GitHub.
- Personalización de vistas de proyectos.



- Colaboración mediante debates de equipo.
- Diseño e implementación de una estrategia para los ciclos de comentarios.
- Diseño e implementación de la rastreabilidad del origen, el error y la calidad.
- Planeación de Agile y administración de carteras con Azure Boards.

### **Módulo 3: Diseño e implementación de estrategias y flujos de trabajo de rama.**

En este módulo se exploran los tipos, conceptos y modelos de ramas de Git para el proceso de entrega continua. Ayuda a las empresas a definir su organización y estrategia de bifurcación.

- Exploración de los tipos de flujo de trabajo de rama.
- Exploración del flujo de trabajo de rama de características.
- Exploración del modelo de rama de Git para la entrega continua.
- Exploración del flujo de GitHub.
- Exploración del flujo de trabajo de bifurcación.
- Implementación de restricciones de combinación de ramas.
- Control de versiones con Git en Azure Repos.

### **Módulo 4: Colaboración con solicitudes de incorporación de cambios en Azure Repos.**

En este módulo se presentan solicitudes de incorporación de cambios para colaboración y revisiones de código mediante Azure DevOps y GitHub Mobile para aprobaciones de solicitudes de incorporaciones de cambios. Le ayuda a comprender cómo funcionan las solicitudes de incorporación de cambios y cómo configurarlas.

- Colaboración con solicitudes de incorporación de cambios.

### **Ejercicio: Colaboración de Azure Repos con solicitudes de incorporación de cambios.**

### **Módulo 5: Exploración de enlaces de Git.**

En este módulo se describen los enlaces de Git y su uso durante el proceso de desarrollo, su implementación y su comportamiento.

- Introducción a los enlaces de Git.
- Implementación de enlaces de Git.

### **Módulo 6: Planeamiento del fomento de un origen interno.**

En este módulo se explica cómo usar Git para fomentar orígenes internos en toda la organización, así como implementar bifurcaciones y sus flujos de trabajo.

- Exploración del fomento de un origen interno.
- Implementación del flujo de trabajo de bifurcación.
- Descripción del origen interno con bifurcaciones.

### **Módulo 7: Administración y configuración de repositorios.**

En este módulo se explora cómo trabajar con grandes repositorios, purgar los datos del repositorio, y administrar y automatizar las notas de la versión mediante GitHub.

- Trabajo con repositorios grandes.
- Exploración de un repositorio mono frente a repositorios múltiples.
- Implementación de un registro de cambios.
- Implementación de Scalar y entre repositorios con el uso compartido entre repositorios.
- Recuperación de datos específicos mediante comandos de Git.
- Purga de datos del repositorio.
- Administración de versiones con GitHub Repos.
- Automatización de las notas de la versión con GitHub.
- Creación de documentación de API.
- Implementación de la automatización de la documentación del historial de Git.
- Configuración de los permisos del repositorio de control de código fuente con GitHub.
- Configuración de etiquetas GitHub para organizar repositorios.

### **Módulo 8: Identificación de la deuda técnica.**

En este módulo se analizan la deuda técnica, las métricas de calidad y complejidad y los planes para revisiones de código y validaciones de su calidad eficaces.

- Análisis de la calidad del código.
- Análisis de las métricas de complejidad y calidad.
- Introducción a la deuda técnica.





- Medida y administración de la deuda técnica.
- Introducción a GitHub Advanced Security.
- Integración de otras herramientas de calidad del código.
- Planificación de revisiones de código eficaces.

### **Ruta de aprendizaje: Implementación de CI con Azure Pipelines y Acciones de GitHub.**

En esta ruta de aprendizaje se presenta el concepto de integración continua mediante Azure Pipelines y Acciones de GitHub y se proporcionan instrucciones para configurar esos servicios y compilar aplicaciones.

### **Módulo 9: Exploración de Azure Pipelines.**

Este módulo se presentan los conceptos de Azure Pipelines y se explican los términos y componentes clave de la herramienta, lo que le ayudará a decidir la estrategia y las responsabilidades respecto a las canalizaciones.

- Exploración del concepto de canalizaciones en DevOps.
- Describir Azure Pipelines.
- Descripción de los términos clave de Azure Pipelines.

### **Módulo 10: Administración de grupos y agentes de Azure Pipelines.**

En este módulo se exploran las diferencias entre los agentes que hospeda Microsoft y los agentes autohospedados, se detallan los tipos de trabajos y se presenta la configuración de los grupos de agentes. Comprenda las situaciones típicas en las que usar grupos de agentes y cómo administrar su seguridad.

- Elección entre agentes hospedados por Microsoft y agentes autohospedados.
- Exploración de tipos de trabajos.
- Introducción a los grupos de agentes.
- Exploración del grupo de agentes predefinido.
- Descripción de situaciones típicas en los grupos de agentes.
- Comunicación con Azure Pipelines.
- Comunicación para implementar en servidores de destino.
- Examen de otras consideraciones.
- Descripción de la seguridad de los grupos de agentes.
- Configurar grupos de agentes y descripción de los estilos de canalización.

### **Módulo 11: Descripción de canalizaciones y simultaneidad.**

En este módulo, se describen los trabajos paralelos y cómo calcular su uso. Además, se presenta Azure Pipelines para proyectos de código abierto y se exploran el diseñador visual y las canalizaciones de YAML.

- Información sobre los trabajos paralelos.
- Estimación de trabajos paralelos.
- Descripción de Azure Pipelines y proyectos de código abierto.
- Exploración de Azure Pipelines y el diseñador visual.
- Descripción de Azure Pipelines y YAML.
- Habilitación de la integración continua con Azure Pipelines.

### **Módulo 12: Diseño e implementación de una estrategia de canalización.**

En este módulo se describen las estrategias de canalización, su configuración, la implementación de compilaciones de varios agentes y los controles de código fuente compatibles con Azure Pipelines.

- Configuración de las demandas del agente.
- Exploración de multiconfiguración y multiagente.
- Integración de repositorios de GitHub y Azure Pipelines.
- Diseño e implementación de una estrategia de prueba completa.
- Implementación de la cobertura de código y visualización en la canalización.
- Implementar compilaciones de varios trabajos.
- Exploración de los tipos de control de código fuente compatibles con Azure Pipelines.

### **Módulo 13: Integración con Azure Pipelines.**

En este módulo, se describen la anatomía y la estructura de Azure Pipelines, las plantillas, los recursos de YAML y cómo usar repositorios múltiples en la canalización.

- Descripción de la anatomía y estructura de la canalización.
- Descripción de plantillas.
- Exploración de recursos de YAML.
- Uso de repositorios múltiples en la canalización.
- Migración de una canalización de clásica a YAML en Azure Pipelines.

## Módulo 14: Introducción a Acciones de GitHub.

En este módulo, ha aprendido qué son Acciones de GitHub, el flujo de acción y sus elementos. Comprenda qué son los eventos, explore trabajos y ejecutores, y descubra cómo leer la salida de la consola desde las acciones.

- ¿Qué son las acciones?
- Exploración del flujo de acciones.
- Descripción de los flujos de trabajo.
- Descripción de los elementos de sintaxis de flujo de trabajo estándar.
- Explorar eventos.
- Exploración de trabajos.
- Exploración de ejecutores.
- Examen de la versión y prueba de una acción.

## Módulo 15: Información sobre la integración continua con Acciones de GitHub.

En este módulo se detalla la integración continua mediante Acciones de GitHub y se describen las variables de entorno, los artefactos, los procedimientos recomendados y cómo proteger la canalización mediante variables y secretos cifrados.

- Descripción de la integración continua con acciones.
- Análisis de las variables de entorno.
- Uso compartido de los artefactos entre trabajos.
- Análisis de los distintivos de flujo de trabajo.
- Descripción de los procedimientos recomendados para crear acciones.
- Marcar versiones con etiquetas de Git.
- Creación de secretos cifrados.
- Uso de secretos en un flujo de trabajo.
- Implementación de Acciones de GitHub para CI/CD.

## Módulo 16: Diseño de una estrategia de creación de contenedores.

Este módulo le ayuda a planear una estrategia de creación de contenedores, explica los contenedores y su estructura, presenta Docker, microservicios, Azure Container Registry y los servicios relacionados.

- Examen de la estructura de los contenedores.
- Trabajo con contenedores de Docker.

- Descripción de los conceptos básicos de Dockerfile.
- Examen de archivos Dockerfile de varias fases.
- Examen de las consideraciones para varias compilaciones de fase.
- Exploración de los servicios relacionados con contenedores de Azure.
- Implementación de contenedores de Docker en aplicaciones web de Azure App Service.

## Ruta de aprendizaje: Diseño e implementación de una estrategia de versiones.

En esta ruta de aprendizaje se explican los conceptos de entrega continua y consideraciones sobre la estrategia de versiones para establecer fases de implementación y la cadencia de la entrega y la implementación, además de cómo configurar aprobaciones de versiones.

## Módulo 17: Creación de una canalización de versión.

En este módulo se describen las funcionalidades, compilación y tarea de versión de Azure Pipelines.

- Descripción de las funciones de canalización de versiones de Azure DevOps.
- Exploración de canalizaciones de versión.
- Exploración de orígenes de artefactos.
- Elección del origen de artefacto adecuado.
- Consideraciones para la implementación en fases.
- Exploración de las tareas de compilación y versión.
- Exploración de las tareas personalizadas de compilación y versión.
- Exploración de trabajos de versión.
- Descripción de la tarea de implementación de bases de datos.
- Configuración de canalizaciones como código con YAML.

## Módulo 18: Exploración de recomendaciones de versión.

En este módulo se exploran las recomendaciones de estrategia de versión críticas que las organizaciones deben tener en cuenta al diseñar implementaciones automatizadas y se explica cómo definir componentes de canalizaciones de versión y orígenes de artefactos, crear aprobaciones y configurar puertas de lanzamiento.





- Comprensión de la cadencia de entrega y tres tipos de desencadenadores.
- Exploración de las aprobaciones de versión.
- Exploración de puertas de lanzamiento.
- Uso de puertas de versión para proteger la calidad.
- Exploración de la estrategia y las recomendaciones de versión de GitOps.
- Control de implementaciones mediante puertas de versión.

## Módulo 19: Aprovisionamiento y prueba de entornos.

En este módulo se detalla el aprovisionamiento del entorno de destino, el proceso de creación de conexiones de servicio y la configuración de la infraestructura de prueba. Aprenda a configurar la automatización de pruebas funcionales y a ejecutar pruebas de disponibilidad.

- Aprovisionamiento y configuración de entornos de destino.
- Configuración de la integración automatizada y automatización de pruebas funcionales.
- Descripción del desplazamiento a la izquierda.
- Configuración y ejecución de pruebas de disponibilidad.
- Exploración de Azure Load Testing.
- Configuración y ejecución de pruebas funcionales.

**Ejercicio: Configuración de conexiones de servicio.**

## Módulo 20: Administración y modularización de tareas y plantillas.

En este módulo se describe la creación de grupos de variables y tareas y el uso de variables de versión y variables de fase en la canalización.

- Examen de grupos de tareas.
- Exploración de variables en canalizaciones de versión.

**Ejercicio: Creación y administración de grupos de variables.**

## Módulo 21: Automatizar la inspección de estado.

En este módulo se describe cómo automatizar la inspección de eventos de mantenimiento, configurar notificaciones en Azure

DevOps y GitHub, configurar enlaces de servicio para supervisar canalizaciones, medir la calidad del proceso de versión y detallar las puertas de versión con fines de calidad. Va a examinar herramientas de administración de versiones y detalles sobre ellas.

- Automatizar la inspección de estado.
- Exploración de eventos y notificaciones.
- Exploración de enlaces de servicio.
- Configuración de las notificaciones de Azure DevOps.
- Configuración de las notificaciones de GitHub.
- Exploración de cómo medir la calidad del proceso de versión.
- Examen de las notas de la versión y la documentación.
- Examen de las consideraciones para elegir herramientas de administración de versiones.
- Exploración de herramientas comunes de administración de versiones.

## Ruta de aprendizaje: Creación de una implementación continua segura mediante Azure Pipelines.

En esta ruta de aprendizaje se explica cómo funcionan las puertas de lanzamiento automáticas, los secretos y la administración de secretos en una canalización. Aprenderá a implementar mecanismos de alerta, crear informes de calidad y recibir notificaciones mediante enlaces de servicio.

## Módulo 22: Introducción a los patrones de implementación.

En este módulo se presentan patrones de implementación y se explica la arquitectura de microservicios para ayudar a mejorar el ciclo de implementación y examinar patrones de implementación clásicos y modernos.

- Exploración de la arquitectura de microservicios.
- Análisis de los patrones de implementación clásicos.
- Descripción de los patrones de implementación modernos.

## Módulo 23: Implementación azul-verde y activación/desactivación de funcionalidad.

En este módulo se describe el proceso de implementación azul-verde y se presentan técnicas de activación/desactivación de funcionalidad para implementar en el proceso de desarrollo.



- ¿Qué es la implementación azul-verde?
- Exploración de las de ranuras de implementación.
- Introducción a la activación/desactivación de funcionalidad.
- Descripción del mantenimiento de activación/desactivación de funcionalidad.

## **Módulo 24: Implementación de versiones de valor controlado e inicio oscuro.**

En este módulo se describen las estrategias de implementación en torno a las versiones de valor controlado e inicio oscuro, y se realiza un examen de algunos Traffic Manager.

- Exploración de las versiones de valor controlado.
- Examen de Traffic Manager.
- Información sobre el inicio oscuro.

## **Módulo 25: Implementación de pruebas A/B y exposición progresiva.**

En este módulo se presentan los conceptos de implementación de pruebas A/B y exposición progresiva y se explora la integración continua y entrega continua con anillos de implementación (implementación basada en anillos).

- ¿Qué son las pruebas A/B?
- Exploración de la integración continua y entrega continua con anillos de implementación.

## **Módulo 26: Integración con sistemas de administración de identidades.**

En este módulo se describe la integración con GitHub y el inicio de sesión único (SSO) para la autenticación, las entidades de servicio y las identidades de servicio administradas.

- Integrar GitHub con el inicio de sesión único (SSO).
- Diseño e implementación de permisos y roles en GitHub.
- Diseño e implementación de permisos y grupos de seguridad en Azure DevOps.
- Exploración de las identidades de carga de trabajo.
- Implementación de identidades administradas.

## **Módulo 27: Administración de datos de configuración de aplicaciones.**

En este módulo se examinan formas de replantearse los datos de configuración de aplicaciones y el método de separación de intereses. Explore Azure App Configuration, detalle los pares clave-valor y la administración de funciones de App Configuration e integre Azure Key Vault con Azure Pipelines.

- Replantearse los datos de configuración de aplicaciones.
- Exploración de la separación de intereses.
- Descripción de los patrones del almacén de configuración externo.
- Implementación de archivos seguros de Azure DevOps.
- Introducción a Azure App Configuration.
- Examen de pares clave-valor.
- Examen de la administración de características de la configuración de aplicaciones.
- Integrar Azure Key Vault con Azure Pipelines.
- Administración de secretos, tokens y certificados.
- Examen del bucle interno y externo de DevOps.
- Integración de Azure Key Vault con Azure DevOps.
- Habilitación de marcas de características y configuración dinámica.

## **Ruta de aprendizaje: Administración de infraestructura como código mediante Azure y DSC.**

En esta ruta de aprendizaje se presenta el concepto de infraestructura como código (IaC) y cómo administrar un entorno de operaciones de la misma forma que se hace con aplicaciones u otros códigos para versiones generales.

## **Módulo 28: Exploración de la administración de configuración y la infraestructura como código.**

En este módulo se describen los conceptos de la infraestructura como código y la creación y configuración de la implementación del entorno. Además, entienda la configuración imperativa, declarativa e idempotente y cómo se aplica a su empresa.

- Exploración de la implementación del entorno.
- Examen de la configuración del entorno.
- Descripción de la configuración imperativa frente a la declarativa.
- Descripción de la configuración idempotente.





## Módulo 29: Crear recursos de Azure mediante las plantillas de Azure Resource Manager.

En este módulo se exploran las plantillas de Azure Resource Manager y sus componentes, y se detallan las dependencias y las plantillas modulares con secretos.

- ¿Por qué conviene utilizar las plantillas de Azure Resource Manager?
- Exploración de los componentes de la plantilla.
- Administración de dependencias.
- Modularización de plantillas.
- Administración de los secretos en las plantillas.

## Módulo 30: Creación de recursos de Azure con la CLI de Azure.

En este módulo se explica la CLI de Azure para crear recursos de Azure, ejecutar plantillas y detallar comandos de la CLI de Azure.

- ¿Qué es la CLI de Azure?
- Uso de la CLI de Azure.

## Módulo 31: Exploración de Azure Automation con DevOps.

En este módulo, se describe Azure Automation con Azure DevOps, mediante runbooks, webhooks y flujos de trabajo de PowerShell. Aprenderá a crear y administrar la automatización para el entorno.

- Creación de cuentas de Automation.
- ¿Qué es un runbook?
- Descripción de los recursos compartidos de automatización.
- Exploración de la galería de runbooks.
- Análisis de webhooks.
- Exploración de la integración del control de código fuente.
- Exploración de flujos de trabajo de PowerShell.
- Creación de un flujo de trabajo.
- Exploración de la administración híbrida.
- Análisis del punto de control y procesamiento paralelo.

## Módulo 32: Implementación de Desired State Configuration (DSC).

En este módulo se describe Desired State Configuration (DSC) y sus componentes para la implementación. Puede practicar cómo

importar, compilar y automatizar la creación del entorno, así como usar DSC para la automatización de Linux en Azure.

- Descripción sobre el desfase de configuración.
- Exploración de Desired State Configuration (DSC).
- Exploración de State Configuration (DSC) de Azure Automation.
- Examen del archivo de configuración de DSC.
- Implementación de DSC y la automatización de Linux en Azure.

## Módulo 33: Implementación de Bicep.

En este módulo se habla sobre Bicep y cómo se integra con diferentes herramientas, como la CLI de Azure y Visual Studio Code, para la configuración de la implementación del entorno.

- ¿Qué es Bicep? e Instalación de Bicep.
- Estructura y sintaxis de un archivo Bicep.
- Implementaciones mediante plantillas de Azure Bicep.

### Ejercicios:

- Creación de plantillas de Bicep.
- Implementación de un archivo de Bicep desde Azure Pipelines.
- Implementación de un archivo de Bicep a partir de flujos de trabajo de GitHub.

## Ruta de aprendizaje: Implementación de seguridad y comprobación del cumplimiento de las de bases de código.

En esta ruta de aprendizaje se explora una infraestructura y una estrategia de configuración y un conjunto de herramientas apropiado para un proceso de lanzamiento y una infraestructura de aplicación. Se explica la implementación de cumplimiento y seguridad en la infraestructura de su aplicación.

## Módulo 34: Introducción a Secure DevOps.

En este módulo se presentan conceptos de DevSecOps, ataques por inyección de código SQL, modelado de amenazas y seguridad para la integración continua.

- Descripción del ataque por inyección de código SQL.
- Información sobre DevSecOps.
- Exploración de la canalización de Secure DevOps.
- Exploración de los puntos de validación clave.
- Exploración de la validación de seguridad continua.

- Descripción del modelado de amenazas.
- Exploración de CodeQL en GitHub.

### **Módulo 35: Implementación de software de código abierto.**

En este módulo se exploran el software de código abierto y los problemas corporativos con los componentes de software. Además, se explican las licencias de código abierto comunes, las implicaciones de licencia y las clasificaciones.

- Exploración de cómo se ha creado el software.
- ¿Qué es el software de código abierto?
- Exploración de problemas corporativos con componentes de software de código abierto
- Introducción a las licencias de código abierto.
- Exploración de licencias de código abierto comunes.
- Examen de las implicaciones y clasificaciones de las licencias.

### **Módulo 36: Análisis de composición de software.**

En este módulo se explica cómo inspeccionar y validar las bases de código para el cumplimiento, la integración con herramientas de seguridad y la integración con Azure Pipelines.

- Inspección y validación de las bases de código para el cumplimiento.
- Exploración del análisis de composición de software (SCA).
- Implementación de las actualizaciones de seguridad y las alertas de Dependabot de GitHub.
- Integración de comprobaciones de análisis de composición de software en canalizaciones.
- Análisis de las herramientas para evaluar la tasa de licencia y la seguridad de los paquetes.
- Automatización del examen de contenedores, incluyendo las imágenes de contenedor.
- Interpretar las alertas de las herramientas de examen.

### **Módulo 37: Supervisión y gobernanza de la seguridad.**

En este módulo se describe la supervisión y gobernanza de la seguridad con Microsoft Defender for Cloud y sus escenarios de uso, directivas de Azure, Microsoft Defender for Identity y prácticas de seguridad relacionadas con las herramientas.

- Implementación de la seguridad de la canalización.
- Exploración de Microsoft Defender for Cloud.
- Examen de escenarios de uso de Microsoft Defender for Cloud.
- Explorar Azure Policy.
- Información sobre las directivas.
- Exploración de iniciativas.
- Exploración de los bloqueos de recursos:
- Información sobre Microsoft Defender for Identity.
- Integración de GitHub Advanced Security con Microsoft Defender for Cloud.
- Configuración de GitHub Advanced Security para GitHub y Azure DevOps.

### **Ruta de aprendizaje: Diseño e implementación de una estrategia de administración de dependencias.**

En esta ruta de aprendizaje se examina la administración de dependencias en el desarrollo de software, cómo identificarlas en el código base y cómo empaquetar y administrar dependencias en fuentes de paquetes.

### **Módulo 38: Exploración de las dependencias de paquetes.**

En este módulo se exploran conceptos relacionados con la administración de dependencias y se ayuda a identificar las dependencias de un proyecto. Aprende cómo descomponer el sistema, identificar dependencias y dividir los paquetes en componentes.

- ¿Qué es la administración de dependencias?
- Descripción de los elementos de una estrategia de administración de dependencias.
- Identificación de las dependencias.
- Información sobre la división en componentes de origen y paquete.
- Descomposición del sistema.
- Búsqueda de dependencias en el código base.

### **Módulo 39: Descripción de la administración de paquetes.**

En este módulo se describen las fuentes de paquetes, los orígenes de paquetes públicos comunes y la creación y publicación de paquetes.







- Exploración de paquetes.
- Descripción de las fuentes de paquetes.
- Exploración de los administradores de la fuente de paquetes.
- Exploración de orígenes de paquetes públicos comunes.
- Exploración de orígenes de paquetes autohospedados y basados en SaaS.
- Consumir paquetes.
- Introducción a Azure Artifacts.
- Publicar paquetes.
- Administración de paquetes con Azure Artifacts.

## **Módulo 40: Migración, consolidación y protección de artefactos.**

En este módulo se detalla la migración, la consolidación y la configuración de paquetes para proteger el acceso a fuentes de paquetes y repositorios de artefactos.

- Identificación de los repositorios de artefactos existentes.
- Migración e integración de repositorios de artefactos.
- Protección del acceso a las fuentes de paquetes.
- Examen de los roles.
- Examen de los permisos.
- Examen de la autenticación.

## **Módulo 41: Implementar una estrategia de control de versiones.**

En este módulo se explican las estrategias de control de versiones para el empaquetado, los procedimientos recomendados del control de versiones y la promoción de paquetes.

- Información sobre el control de versiones de artefactos.
- Exploración del control de versiones semántico.
- Análisis de las vistas de versión.
- Promover paquetes.
- Exploración de los procedimientos recomendados del control de versiones.

## **Módulo 42: Introducción a los paquetes de GitHub.**

En este módulo se presentan los paquetes de GitHub. Se exploran las formas de controlar los permisos y la visibilidad de los paquetes, y cómo publicarlos, instalarlos, eliminarlos y restaurarlos mediante GitHub.

- Publicar paquetes.
- Instalación de un paquete.
- Eliminación y restauración de paquetes.
- Exploración del control de acceso y la visibilidad de paquetes.

## **Ruta de aprendizaje: Implementación de comentarios continuos.**

En esta ruta de aprendizaje se presenta el concepto de comentarios continuos y se describe cómo implementarlo en un ciclo de DevOps.

## **Módulo 43: Implementación de herramientas para realizar un seguimiento del uso y el flujo.**

En este módulo se presentan las prácticas y herramientas de comentarios continuos para realizar un seguimiento del uso y el flujo, como Azure Logs Analytics, Lenguaje de consulta Kusto (KQL) y Application Insights.

- Descripción del bucle interno.
- Introducción a la supervisión continua.
- Exploración de Azure Monitor y Log Analytics.
- Análisis del Lenguaje de consulta Kusto (KQL).
- Exploración de Application Insights.
- Implementación de Application Insights.
- Diseño e implementación de métricas y consultas.
- Supervisión del rendimiento de aplicaciones con Application Insights.

## **Módulo 44: Desarrollo de paneles de supervisión y estado.**

En este módulo se explican los pasos para desarrollar la supervisión con paneles de Azure, trabajar con el Diseñador de vistas y Azure Monitor y crear libros de Azure Monitor. Asimismo, explore las herramientas que admiten la supervisión con Power BI.

- Configuración de la supervisión en GitHub.
- Exploración de paneles de Azure.
- Exploración de libros de Azure Monitor.
- Exploración de Power BI.
- Compilar su propia aplicación personalizada.
- Supervisión del estado de la canalización, incluida la tasa de errores, la duración y las pruebas poco confiables.



- Optimización de una canalización para el costo, tiempo, rendimiento y fiabilidad.
- Optimización de la simultaneidad de canalizaciones para el rendimiento y el costo.
- Reducción de las alertas sin sentido y que no son procesables.
- Examen de una retrospectiva sin culpables.
- Desarrollo de una cultura justa.

## **Módulo 45: Compartir conocimientos en los equipos.**

En este módulo se describe cómo compartir conocimientos en los equipos, wikis de Azure DevOps y la integración con Azure Boards.

- Compartir los conocimientos adquiridos en los equipos de desarrollo.
- Introducción a wikis de proyecto de Azure DevOps.
- Integración de GitHub y Azure DevOps con Microsoft Teams.
- Uso compartido del conocimiento de equipo con las wikis de proyectos de Azure.

## **Módulo 46: Diseñar procesos para automatizar el análisis de aplicaciones.**

Este módulo ayuda a diseñar procesos para Application Insights, y explora las herramientas y tecnologías de telemetría y supervisión.

- Exploración de respuestas rápidas y búsqueda aumentada.
- Integrar la telemetría.
- Examen de las herramientas y las tecnologías de supervisión.
- Explorar el Conector de Administración de servicios de TI.

## **Módulo 47: Administración de alertas, retrospectivas sin culpables y una referencia cultural justa.**

En este módulo se examinan las alertas y las retrospectivas sin culpables y se crea una referencia cultural justa. Ayuda a mejorar el rendimiento de las aplicaciones y a reducir las alertas sin sentido y que no se pueden procesar, y explica la degradación del tiempo de respuesta del servidor.

- Examen de cuándo obtener una notificación.
- Exploración de la manera de corregirlo.
- Exploración de las notificaciones de detección inteligente.
- Mejorar el rendimiento.
- Descripción de la degradación del tiempo de respuesta del servidor.

