Para diseñar un plan de mantenimiento y soporte de software basado en la norma ISO 14724, es importante entender los dos tipos de mantenimiento que se abordan: \*\*preventivo\*\* y \*\*correctivo\*\*. El mantenimiento preventivo tiene como objetivo anticipar posibles fallas antes de que ocurran, mientras que el correctivo se realiza después de que se detecta un problema o error.

A continuación, te guiaré paso a paso para elaborar un plan de mantenimiento y soporte de software. Además, te proporcionaré detalles clave para cumplir con cada apartado solicitado.

### 1. \*\*Portada\*\*

- \*\*Título del documento\*\*: Plan de Mantenimiento y Soporte del Software.

- \*\*Nombre del autor\*\*.

- \*\*Nombre del proyecto formativo\*\*.

- \*\*Fecha\*\*.

### 2. \*\*Introducción\*\*

En la introducción, proporciona una descripción general del propósito del documento. Explica brevemente qué es el mantenimiento de software, su importancia y la razón por la que se aborda tanto el mantenimiento preventivo como el correctivo. Aclara que el plan sigue el estándar ISO 14724.

### 3. \*\*Descripción del sistema\*\*

Describe el sistema o software para el cual se diseña el plan de mantenimiento. Debes incluir:

- \*\*Nombre del software\*\*.

- \*\*Descripción general\*\* de su funcionalidad.

- \*\*Arquitectura\*\*: por ejemplo, si es cliente-servidor, basado en la web, etc.

- \*\*Módulos principales\*\* y funcionalidades clave.

Esta sección debe ofrecer un panorama claro para que cualquier persona que lo lea pueda entender qué sistema está bajo mantenimiento.

### 4. \*\*Proceso de implementación\*\*

Describe cómo se llevará a cabo el mantenimiento preventivo y correctivo:

- \*\*Mantenimiento preventivo\*\*: especifica tareas programadas como:

- Verificación de logs y alertas.

- Actualizaciones de seguridad.

- Pruebas de rendimiento y de integridad de datos.

- Limpieza de datos obsoletos.

- \*\*Mantenimiento correctivo\*\*: define los pasos a seguir cuando se detecten errores, tales como:

- Clasificación de errores por gravedad.

- Diagnóstico del problema.

- Corrección del código o del sistema.

- Pruebas posteriores a la corrección.

\*\*Frecuencia\*\*: Determina la periodicidad del mantenimiento preventivo (semanal, mensual, trimestral, etc.).

### 5. \*\*Análisis de modificación y problemas\*\*

Esta sección debe detallar cómo se analizarán las modificaciones y problemas que surjan. Considera los siguientes puntos:

- \*\*Identificación de problemas\*\*: cómo se detectan los errores o fallos (informes de usuarios, herramientas de monitoreo).

- \*\*Análisis de impacto\*\*: evaluación del impacto de cada modificación o problema en el sistema. Por ejemplo, si afecta solo una funcionalidad o varias.

- \*\*Decisión de modificar\*\*: si el problema requiere una modificación, deberías definir el protocolo a seguir antes de realizar cambios (aprobación por equipo técnico, etc.).

### 6. \*\*Implementación de la modificación\*\*

Explica el proceso para implementar modificaciones cuando surjan errores o problemas que requieran cambios en el código o en la configuración del sistema:

- \*\*Fases de desarrollo\*\*: desarrollo, prueba, y despliegue de la modificación.

- \*\*Ambiente de pruebas\*\*: cómo se probarán los cambios antes de implementarlos en el ambiente de producción.

- \*\*Plan de comunicación\*\*: cómo y cuándo se notificará a los usuarios sobre los cambios o paradas del servicio.

### 7. \*\*Aceptación y revisión del mantenimiento\*\*

En esta sección, debes describir cómo se verificará que el mantenimiento se ha realizado correctamente:

- \*\*Pruebas de verificación\*\*: revisiones que confirmen que el software funciona correctamente tras el mantenimiento (correctivo o preventivo).

- \*\*Retroalimentación del usuario\*\*: encuestas o reportes que los usuarios puedan proporcionar para confirmar que todo está funcionando.

- \*\*Revisión del equipo técnico\*\*: evaluación interna sobre el proceso de mantenimiento realizado.

### 8. \*\*Migración\*\*

Si es necesario migrar el sistema a una nueva plataforma o tecnología, este apartado debe detallar el proceso de migración:

- \*\*Plan de migración\*\*: cómo se realizarán las copias de seguridad, transferencia de datos, y la implementación del sistema en una nueva infraestructura.

- \*\*Pruebas de migración\*\*: cómo se verificará que el sistema funciona adecuadamente después de la migración.

- \*\*Capacitación\*\*: formación que pueda requerirse para el personal o usuarios finales tras la migración.

### 9. \*\*Retiro\*\*

Explica qué pasará cuando el sistema llegue al final de su vida útil:

- \*\*Criterios de retiro\*\*: define los factores que determinarían cuándo retirar el sistema (obsolescencia tecnológica, falta de soporte, etc.).

- \*\*Plan de desactivación\*\*: cómo se cerrará el sistema y se gestionarán los datos históricos.

- \*\*Alternativas o sistemas sustitutos\*\*: si existe un sistema que reemplazará al actual, deberías explicarlo aquí.

### 10. \*\*Cronograma\*\*

El cronograma debe detallar las fechas o periodos específicos para llevar a cabo cada tipo de mantenimiento. Puedes utilizar un formato de tabla para mostrar:

- \*\*Tareas de mantenimiento preventivo\*\* (por ejemplo, revisiones mensuales de seguridad).

- \*\*Tareas de mantenimiento correctivo\*\* (incluir estimaciones de tiempos para respuesta a incidencias).

### 11. \*\*Fuentes consultadas\*\*

Incluye cualquier fuente bibliográfica o digital que hayas utilizado para diseñar el plan, como normativas ISO, libros o artículos de la web.

### Consejos adicionales:

- Utiliza un lenguaje claro y técnico, adecuado para un documento profesional.

- Asegúrate de que el diseño del documento cumpla con las normas de presentación de documentos escritos (portada, numeración de páginas, títulos adecuados, etc.).

Este documento estructurado te proporcionará una base sólida para gestionar el mantenimiento y soporte de tu software, cumpliendo con los requisitos de calidad descritos en la norma ISO 14724.