

Realizar las pruebas funcionales de software, planes de mantenimiento y soporte y

Documentación y procesos de migración

Juan Luis Becquet Martínez

Servicio Nacional de Aprendizaje

(2721441)Análisis y desarrollo de software

Ingrid Caterine Ramirez Aldana

27 de Octubre de 2024

Sumario

Introducción.....	3
Análisis de Modificaciones y Problemas.....	4
1. Problemas Iniciales.....	4
2.Desarrollo del Mantenimiento.....	5
Mantenimiento Correctivo.....	5
Mantenimiento Adaptativo.....	5
Mantenimiento Perfectivo.....	6
Mantenimiento Preventivo.....	7
3. Análisis de Resultados de Modificaciones.....	7
4. Problemas Persistentes y Recomendaciones.....	7
Aceptación y Revisión del Mantenimiento.....	9
1. Objetivos de Aceptación.....	9
2. Criterios de Aceptación.....	9
3. Proceso de Revisión.....	9
4. Aceptación Final.....	10
5. Recomendaciones Futuras.....	10
Migración del Software de Gestión de Notas.....	11
1. Objetivos de la Migración.....	11
2. Planificación de la Migración.....	11
3. Ejecución de la Migración.....	11
4. Validación de la Migración.....	12
5. Migración en Producción.....	12
6. Post-Migración y Soporte.....	12
Retiro del Software de Gestión de Notas.....	14
1. Objetivos del Retiro.....	14
2. Planificación del Retiro.....	14
3. Ejecución del Retiro.....	14
4. Post-Retiro y Evaluación.....	15
Lista de Chequeo para la Revisión de Modificaciones del Software de Gestión de Notas..	16
Conclusión.....	18
Bibliografía.....	19

Introducción

En el contexto actual de la educación, el uso de tecnologías de la información es fundamental para la gestión eficiente de procesos académicos. Los sistemas de gestión de notas desempeñan un papel crucial, ya que permiten a instituciones educativas registrar, evaluar y analizar el rendimiento de sus estudiantes de manera efectiva. Sin embargo, el funcionamiento óptimo de estas herramientas depende de un diseño robusto y un mantenimiento continuo. En este informe, se abordará el caso de un software de gestión de notas que presentaba problemas de funcionamiento relacionados con su base de datos y deficiencias en la calidad del código implementado.

En particular, la base de datos del sistema estaba experimentando lentitud en las consultas y errores en la recuperación de datos, lo que afectaba la experiencia del usuario y la integridad de la información. Además, el código del software no seguía las mejores prácticas de programación, resultando en una arquitectura difícil de mantener y susceptible a fallos. Ante esta situación, se hizo necesaria la implementación de un plan de mantenimiento alineado con la norma ISO/IEC 14764, que establece las directrices para el mantenimiento efectivo del software.

El objetivo principal de este informe es documentar el proceso de diagnóstico y las actividades realizadas para corregir los errores existentes, optimizar el rendimiento de la base de datos y mejorar la calidad del código. A través de un enfoque estructurado y sistemático, se busca no solo resolver los problemas actuales, sino también establecer un marco de referencia para el mantenimiento preventivo y el desarrollo futuro del sistema, garantizando su funcionalidad y adaptabilidad en un entorno educativo en constante cambio.

Análisis de Modificaciones y Problemas

1. Problemas Iniciales

Antes de implementar el proceso de mantenimiento y optimización, se identificaron varios problemas críticos en el software de gestión de notas:

- **Rendimiento de la Base de Datos:** Las consultas SQL eran lentas, lo que resultaba en tiempos de espera prolongados para los usuarios. En algunos casos, las consultas tardaban más de 10 segundos en completarse, lo que impactaba negativamente en la experiencia del usuario.
- **Errores en la Recuperación de Datos:** Se reportaron errores frecuentes al intentar recuperar datos, especialmente al generar informes de calificaciones. Esto generaba desconfianza en la precisión de los datos presentados.
- **Código No Optimizado:** El código del software estaba desorganizado, con redundancias y falta de comentarios, lo que dificultaba su mantenimiento y evolución. Se utilizaban prácticas de programación obsoletas, lo que hacía que el sistema fuera susceptible a errores.
- **Falta de Monitoreo:** No existían herramientas de monitoreo implementadas, lo que dificultaba la detección de problemas antes de que afectaran a los usuarios finales.

2.Desarrollo del Mantenimiento

El mantenimiento del software de gestión de notas se llevó a cabo siguiendo los cuatro tipos de mantenimiento definidos por la norma ISO/IEC 14764: correctivo, adaptativo, perfectivo y preventivo. A continuación, se detalla cada uno de estos tipos de mantenimiento y las actividades específicas realizadas.

Mantenimiento Correctivo

El mantenimiento correctivo se centró en la identificación y corrección de errores y fallos en el sistema. Las actividades incluyeron:

- **Análisis de Errores:** Se revisaron los registros de errores generados por el sistema y se analizaron los informes de los usuarios para identificar las principales áreas problemáticas. Esto incluyó consultas SQL que mostraban un rendimiento deficiente y errores en la recuperación de datos.
- **Corrección de Consultas SQL:** Se identificaron consultas que tardaban demasiado en ejecutarse. Las más críticas se optimizaron, mejorando su estructura y utilizando índices apropiados. Por ejemplo, se modificó la consulta que recuperaba las calificaciones de los estudiantes para que utilizara joins en lugar de subconsultas, reduciendo así el tiempo de ejecución.
- **Pruebas de Regresión:** Después de implementar las correcciones, se realizaron pruebas exhaustivas para asegurar que las modificaciones no introdujeran nuevos errores. Se crearon pruebas automatizadas para cubrir los escenarios más comunes, lo que facilitaría futuras actualizaciones.

Mantenimiento Adaptativo

El mantenimiento adaptativo se enfocó en ajustar el software a los cambios en el entorno y en los requisitos del usuario. Las actividades incluyeron:

- **Actualización de la Base de Datos:** Se realizó una actualización de la versión de MySQL utilizada por el sistema, migrando de la versión 5.6 a la 8.0. Esta actualización no solo mejoró la seguridad, sino que también permitió la utilización de nuevas funciones de rendimiento.
- **Ajuste de Configuraciones:** Se revisaron las configuraciones del servidor de base de datos para maximizar el rendimiento. Esto incluyó la optimización de la configuración de caché y el número de conexiones simultáneas permitidas, así como la revisión de las configuraciones de memoria para garantizar un uso eficiente de los recursos.

- **Adaptación a Nuevas Normativas:** Se llevaron a cabo ajustes en el software para cumplir con nuevas normativas educativas que requerían la implementación de funcionalidades adicionales, como el seguimiento del rendimiento académico en tiempo real. Se implementó un nuevo módulo que permite a los docentes ingresar notas de manera más eficiente y obtener reportes de rendimiento.

Mantenimiento Perfectivo

El mantenimiento perfectivo se centró en la mejora del rendimiento y la mantenibilidad del software. Las actividades incluyeron:

- **Refactorización del Código:** Se revisó y reestructuró el código existente para mejorar su legibilidad y eficiencia. Se aplicaron principios de programación como DRY (Don't Repeat Yourself) y KISS (Keep It Simple, Stupid). Esto incluyó la eliminación de funciones redundantes y la reestructuración de módulos para facilitar su comprensión y uso.
- **Optimización de Algoritmos:** Se mejoraron algunos algoritmos utilizados en la lógica del software. Por ejemplo, se optimizó el algoritmo para calcular el promedio de notas, que ahora utiliza técnicas de programación más eficientes, reduciendo el tiempo de procesamiento.
- **Implementación de Consultas Preparadas:** Se introdujeron consultas SQL preparadas para mejorar la seguridad y el rendimiento del sistema. Esto no solo ayuda a prevenir ataques de inyección SQL, sino que también optimiza la ejecución de consultas repetidas.

Mantenimiento Preventivo

El mantenimiento preventivo se llevó a cabo para evitar problemas futuros y garantizar la continuidad del servicio. Las actividades incluyeron:

- **Monitoreo Continuo:** Se implementaron herramientas de monitoreo como New Relic y Grafana para realizar un seguimiento del rendimiento del sistema en tiempo real. Estas herramientas permiten identificar problemas antes de que afecten a los usuarios finales y ofrecen estadísticas sobre el uso del sistema.
- **Establecimiento de Protocolos de Mantenimiento Regular:** Se desarrollaron protocolos para realizar revisiones periódicas del sistema. Esto incluye la revisión de logs de errores, pruebas de rendimiento y actualizaciones de seguridad cada tres meses.
- **Capacitación y Documentación:** Se elaboró documentación detallada sobre las modificaciones realizadas en el sistema y se proporcionó capacitación al personal técnico sobre cómo manejar el nuevo entorno de software, lo que ayudará a mantener la calidad y el rendimiento del sistema en el futuro.

3. Análisis de Resultados de Modificaciones

Las modificaciones realizadas resultaron en mejoras significativas en el funcionamiento del software:

- **Mejora del Rendimiento:** Después de las optimizaciones, el tiempo promedio de las consultas se redujo en un 75%, mejorando la experiencia del usuario y la confianza en el sistema.
- **Reducción de Errores:** Se redujeron drásticamente los errores en la recuperación de datos, con una tasa de fallos cercana al 2% en comparación con el 15% anterior.
- **Facilidad de Mantenimiento:** La refactorización del código permitió al equipo de desarrollo implementar nuevas funcionalidades más rápidamente y con menor riesgo de introducir errores.
- **Capacidad de Monitoreo:** La implementación de herramientas de monitoreo proporcionó una visibilidad constante del rendimiento del sistema, permitiendo la detección temprana de problemas antes de que impacten a los usuarios.

4. Problemas Persistentes y Recomendaciones

A pesar de las mejoras logradas, algunos problemas persistentes fueron identificados durante el proceso de evaluación:

- **Resistencia al Cambio por Parte de los Usuarios:** Algunos usuarios mostraron resistencia a adoptar las nuevas funcionalidades y cambios en el sistema. Es fundamental proporcionar capacitación continua y soporte técnico para facilitar esta transición.
- **Necesidad de Ajustes Continuos:** Aunque se implementaron mejoras significativas, el entorno educativo y los requisitos de los usuarios continúan evolucionando. Se recomienda establecer un proceso de revisión y mejora continua que permita adaptarse a los cambios y requerimientos futuros.
- **Escalabilidad del Sistema:** A medida que el número de usuarios crece, es necesario evaluar la escalabilidad del sistema para asegurar que puede manejar cargas adicionales sin perder rendimiento. Se sugiere realizar pruebas de carga y considerar arquitecturas escalables en el futuro.

- **Aceptación y Revisión del Mantenimiento**

- 1. **Objetivos de Aceptación**

La aceptación del mantenimiento realizado en el software de gestión de notas se centró en validar que las modificaciones realizadas cumplieran con los objetivos establecidos al inicio del proceso. Estos objetivos incluían:

- Mejorar el rendimiento de las consultas de la base de datos.
- Reducir la tasa de errores en la recuperación de datos.
- Optimizar la estructura y legibilidad del código.
- Implementar un sistema de monitoreo efectivo para la detección de problemas en tiempo real.

- 2. **Criterios de Aceptación**

Se establecieron criterios claros para la aceptación de las modificaciones:

- **Rendimiento:** Las consultas SQL optimizadas debían mostrar una reducción del 70% en los tiempos de respuesta en comparación con las versiones anteriores.
- **Tasa de Errores:** La tasa de errores en la recuperación de datos debía bajar por debajo del 5%.
- **Mantenibilidad del Código:** Se evaluó la legibilidad del código mediante herramientas de análisis estático, y se debía lograr un aumento en la puntuación de mantenibilidad en un 30%.
- **Monitoreo:** El sistema de monitoreo debía proporcionar alertas en tiempo real y reportes semanales sobre el rendimiento del sistema.

- 3. **Proceso de Revisión**

La revisión del mantenimiento se llevó a cabo mediante una serie de pasos estructurados:

- **Evaluación de Rendimiento:** Se realizaron pruebas de rendimiento antes y después de las modificaciones. Las consultas SQL se ejecutaron en un entorno de prueba y se midieron los tiempos de respuesta y la carga del servidor. Los resultados mostraron una reducción del 75% en los tiempos de respuesta, superando el objetivo establecido.
- **Pruebas de Recuperación de Datos:** Se llevaron a cabo pruebas exhaustivas para verificar la recuperación de datos. Se simulaban diversas situaciones para evaluar la tasa de errores. Los resultados mostraron una tasa de errores inferior al 2%, cumpliendo con los criterios de aceptación.

- **Revisión del Código:** Se aplicaron herramientas de análisis estático de código para evaluar la legibilidad y mantenibilidad. Se obtuvo una puntuación de mantenibilidad que reflejó una mejora del 35%, cumpliendo con el objetivo de aumento del 30%.
- **Evaluación del Sistema de Monitoreo:** Se revisaron las configuraciones de las herramientas de monitoreo implementadas, asegurando que proporcionaban alertas precisas y relevantes. Se generaron reportes semanales que mostraron tendencias de rendimiento estables y sin incidentes críticos.

4. Aceptación Final

Con base en los resultados de la revisión, se presentó un informe a las partes interesadas que incluía:

- Resumen de las mejoras realizadas.
- Comparación de los resultados antes y después del mantenimiento.
- Evaluación de los objetivos alcanzados.

Las partes interesadas revisaron el informe y se llegó a un consenso sobre la aceptación del mantenimiento. Se acordó que el sistema cumplía con los objetivos establecidos y estaba listo para ser utilizado en producción.

5. Recomendaciones Futuras

Aunque se logró una aceptación positiva del mantenimiento, se identificaron algunas recomendaciones para asegurar la continuidad de la calidad del software:

- **Mantenimiento Continuo:** Se sugiere establecer un plan de mantenimiento regular que incluya revisiones semestrales del rendimiento y la calidad del código.
- **Capacitación Adicional:** Proporcionar capacitación continua a los usuarios para asegurar una adopción efectiva de las nuevas funcionalidades y cambios en el sistema.
- **Monitoreo Proactivo:** Continuar utilizando las herramientas de monitoreo implementadas y ajustar configuraciones según sea necesario para adaptarse a la evolución del uso del software.

Migración del Software de Gestión de Notas

1. Objetivos de la Migración

La migración del software de gestión de notas tenía como objetivos principales:

- Actualizar el sistema de gestión de bases de datos de MySQL 5.6 a MySQL 8.0 para beneficiarse de mejoras en rendimiento y seguridad.
- Asegurar que todos los datos existentes fueran trasladados sin pérdida de información.
- Mantener la integridad de la base de datos y la funcionalidad del software durante y después del proceso de migración.

2. Planificación de la Migración

La planificación de la migración se llevó a cabo mediante los siguientes pasos:

- **Evaluación del Entorno Actual:** Se realizó un análisis completo del entorno actual del software, incluyendo la versión de la base de datos, estructuras de tablas, y procedimientos almacenados. Se generaron informes sobre el uso de la base de datos y se identificaron potenciales problemas de compatibilidad.
- **Elaboración de un Plan de Migración:** Se creó un plan que incluía un cronograma, tareas específicas, responsables, y recursos necesarios. Se decidió realizar la migración en un entorno de desarrollo primero para validar los cambios antes de implementarlos en producción.

3. Ejecución de la Migración

La migración se llevó a cabo siguiendo estos pasos:

- **Respaldo de Datos:** Se realizó un respaldo completo de la base de datos existente, asegurando que todos los datos estuvieran protegidos. Este paso incluyó tanto los datos de las tablas como los procedimientos almacenados y configuraciones de la base de datos.
- **Instalación de MySQL 8.0:** Se instaló MySQL 8.0 en un entorno de desarrollo, configurando todos los parámetros necesarios para asegurar un rendimiento óptimo y la compatibilidad con el software existente.
- **Migración de Datos:**
 - Se utilizó la herramienta de migración proporcionada por MySQL para transferir los datos desde la versión anterior a la nueva. Durante este proceso, se monitoreó el progreso y se registraron cualquier error o advertencia.

- Se revisaron las configuraciones de las tablas y se ajustaron según las nuevas características y mejoras de MySQL 8.0, como la implementación de nuevos tipos de índices y opciones de almacenamiento.
- **Pruebas de Integridad:** Una vez completada la migración, se llevaron a cabo pruebas exhaustivas para validar que todos los datos se habían trasladado correctamente y que no se había producido ninguna pérdida de información. Esto incluyó la comparación de registros entre la base de datos antigua y la nueva.

4. Validación de la Migración

Después de la migración, se realizó una serie de validaciones:

- **Pruebas Funcionales:** Se ejecutaron pruebas funcionales en el software de gestión de notas para asegurarse de que todas las funcionalidades operaran correctamente con la nueva versión de la base de datos. Esto incluyó la creación, modificación y eliminación de registros, así como la generación de informes.
- **Monitoreo del Rendimiento:** Se utilizaron herramientas de monitoreo para evaluar el rendimiento del sistema después de la migración. Se observó una mejora en los tiempos de respuesta de las consultas, validando que las optimizaciones realizadas eran efectivas.

5. Migración en Producción

Una vez que se validó que el entorno de desarrollo funcionaba correctamente con la nueva versión de la base de datos, se programó la migración en el entorno de producción:

- **Planificación de Ventanas de Mantenimiento:** Se seleccionó un período de baja actividad para realizar la migración en producción, minimizando el impacto en los usuarios.
- **Ejecución de la Migración en Producción:** Se repitieron todos los pasos de migración en el entorno de producción, asegurando la integridad de los datos y la continuidad del servicio.

6. Post-Migración y Soporte

Después de la migración, se llevaron a cabo las siguientes actividades:

- **Capacitación del Personal:** Se proporcionó capacitación al equipo técnico y a los usuarios finales sobre las nuevas funcionalidades y cambios en la base de datos.
- **Establecimiento de Soporte Técnico:** Se configuró un período de soporte técnico post-migración para atender cualquier problema o inquietud que pudiera surgir tras el cambio.

- **Revisión y Documentación:** Se actualizó la documentación del sistema para reflejar la nueva configuración de la base de datos y los cambios en la arquitectura del software.

Retiro del Software de Gestión de Notas

1. Objetivos del Retiro

El proceso de retiro del software de gestión de notas tenía como objetivos principales:

- Desactivar de manera segura la versión anterior del software y la base de datos, minimizando el riesgo de pérdida de datos.
- Asegurar que los usuarios tuvieran acceso continuo al nuevo sistema sin interrupciones.
- Documentar el proceso de retiro para futuras referencias.

2. Planificación del Retiro

La planificación del retiro se llevó a cabo mediante los siguientes pasos:

- **Evaluación del Sistema Existente:** Se realizó un análisis del software y la base de datos actuales para identificar todos los componentes que debían ser desactivados.
- **Desarrollo de un Plan de Retiro:** Se elaboró un plan que incluía un cronograma para el retiro, las tareas necesarias y los responsables de cada actividad.

3. Ejecución del Retiro

El retiro se llevó a cabo siguiendo estos pasos:

- **Notificación a los Usuarios:** Se informó a todos los usuarios sobre el proceso de retiro, incluyendo fechas y posibles interrupciones en el servicio. Se les proporcionó información sobre las nuevas funcionalidades disponibles en el sistema actualizado.
- **Desactivación del Software Antiguo:** Se desactivó el acceso a la versión anterior del software, asegurando que los usuarios solo pudieran acceder a la nueva versión. Esto se hizo de forma controlada, garantizando que no hubiera interrupciones inesperadas en el servicio.
- **Eliminación de la Base de Datos Antigua:**
 - Una vez confirmado que todos los datos se habían migrado correctamente a la nueva base de datos, se procedió a eliminar la base de datos anterior. Esto incluyó la eliminación de las tablas, índices y procedimientos almacenados relacionados con la versión anterior.
 - Se llevaron a cabo respaldos adicionales de seguridad antes de la eliminación definitiva, para garantizar que no hubiera pérdida de información crítica.

- **Desinstalación del Software:** Se desinstaló la versión antigua del software de los servidores, liberando recursos y asegurando que no hubiera conflictos con la nueva instalación.

4. Post-Retiro y Evaluación

Después del retiro, se realizaron las siguientes actividades:

- **Monitoreo del Nuevo Sistema:** Se implementaron herramientas de monitoreo para asegurar que el nuevo sistema estuviera funcionando de manera óptima. Se observaron métricas de rendimiento y se atendieron rápidamente cualquier problema reportado por los usuarios.
- **Recopilación de Feedback:** Se solicitó a los usuarios su opinión sobre el nuevo sistema y se evaluó la experiencia de uso. Se realizaron ajustes menores según los comentarios recibidos para mejorar la satisfacción del usuario.
- **Documentación del Proceso:** Se elaboró un informe detallado sobre el proceso de retiro, incluyendo los pasos seguidos, cualquier problema encontrado y las soluciones aplicadas. Este documento servirá como referencia para futuros retiros de software.

Lista de Chequeo para la Revisión de Modificaciones del Software de Gestión de Notas

1. Revisión del Rendimiento de la Base de Datos

- ¿Se han realizado pruebas de rendimiento antes y después de las modificaciones?
- ¿Las consultas SQL optimizadas cumplen con el tiempo de respuesta establecido?
- ¿Se han implementado índices adecuados para mejorar el rendimiento?
- ¿Se han comparado los resultados de rendimiento con los informes anteriores?

2. Validación de la Recuperación de Datos

- ¿Se han realizado pruebas exhaustivas para validar la recuperación de datos?
- ¿La tasa de errores en la recuperación de datos es inferior al 5%?
- ¿Se han revisado los registros para asegurar que no hay datos faltantes?
- ¿Se han documentado los procedimientos para la generación de informes?

3. Evaluación del Código y la Mantenibilidad

- ¿Se han realizado revisiones de código para evaluar la legibilidad y la organización?
- ¿El código cumple con las pautas de programación establecidas?
- ¿Se han añadido comentarios y documentación donde sea necesario?
- ¿Se ha utilizado alguna herramienta de análisis estático para medir la mantenibilidad?

4. Verificación del Sistema de Monitoreo

- ¿Se han configurado herramientas de monitoreo para rastrear el rendimiento del sistema?
- ¿Se generan alertas en tiempo real para eventos críticos?
- ¿Se han realizado revisiones periódicas de los reportes de rendimiento?
- ¿Se han ajustado las configuraciones del sistema de monitoreo según sea necesario?

5. Capacitación y Adopción del Usuario

- ¿Se ha proporcionado capacitación adecuada a los usuarios sobre las nuevas funcionalidades?
- ¿Se ha recopilado feedback de los usuarios sobre la experiencia con el nuevo sistema?
- ¿Se han implementado ajustes basados en el feedback recibido?
- ¿Los usuarios tienen acceso a recursos de soporte técnico?

6. Documentación del Proceso de Revisión

- ¿Se ha documentado el proceso de revisión y los hallazgos de manera detallada?
- ¿Se han creado informes sobre las pruebas de rendimiento y la validación de datos?
- ¿Se ha actualizado la documentación técnica para reflejar las modificaciones realizadas?
- ¿Se han archivado todos los registros relevantes para futuras referencias?

Conclusión

En conclusión, el proceso de mantenimiento y optimización implementado permitió resolver problemas críticos de funcionamiento y rendimiento en el sistema de gestión de notas escolares. Las actividades realizadas, en consonancia con los lineamientos de la norma ISO/IEC 14764, no solo lograron mejorar la eficiencia del sistema en términos de velocidad y uso de recursos, sino que también aumentaron la facilidad de mantenimiento futuro gracias a la limpieza y documentación del código. Este enfoque sistemático y estructurado en el mantenimiento de software demostró ser fundamental para asegurar la calidad y la escalabilidad del sistema, proporcionando una base sólida para futuros desarrollos y adaptaciones según las necesidades de la institución educativa. Se recomienda continuar con la implementación de prácticas de monitoreo y optimización preventiva para garantizar la sostenibilidad del sistema a largo plazo.

Bibliografia

ISO/IEC 14764. (2006). Software engineering—Maintenance.

New Relic. (2024). Performance monitoring tools. <https://newrelic.com/>

Grafana Labs. (2024). Grafana documentation. <https://grafana.com/docs/>