

Plan de gestión de la calidad



Fecha: 18/02/2025

Grupo 8: Infanitem

Ideando un proyecto

Álvaro Jiménez Osuna
Ángela López Oliva
Antonio Jiménez Ortega
Daniel del Castillo Piñero
David Fuentelsaz
Rodríguez
David Vargas Muñiz

Enrique García Abadía
Felipe Solís Agudo
Javier Santos Martín
Javier Ulecia García
José García de Tejada
Delgado

Jose Maria Morgado
Prudencio
Josué Rodríguez López
Lucía Noya Cano
Luis Giraldo Santiago
Miguel Galán Lerate
Paula Luna Navarr



PROYECTO	MiDeporte - Sistema de gestión de alquileres de instalaciones deportivas	CÓDIGO DE PROYECTO	G8	FECHA DE ELABORACIÓN	18/02/2025
----------	--------------------------------------------------------------------------	--------------------	----	----------------------	------------

1. NORMAS Y PROCEDIMIENTOS A APLICAR

Normas:

1. ISO/IEC 25010:2011 - Calidad del Producto Software: Este estándar proporciona un marco para evaluar y garantizar la calidad del software a través de varias características clave. Al aplicar esta norma, aseguras que el software tiene las cualidades necesarias para satisfacer las necesidades del usuario y mantener la robustez técnica. Sus principales características son:

- **Funcionalidad:** El software debe cumplir con los requisitos y expectativas funcionales, asegurando que todas las funcionalidades solicitadas estén correctamente implementadas.
- **Fiabilidad:** Garantizar que el software puede mantener su nivel de rendimiento bajo diferentes condiciones. Deben realizarse pruebas de estabilidad y recuperación ante fallos.
- **Usabilidad:** El software debe ser intuitivo y fácil de usar, con una interfaz clara y accesible para los usuarios. Implementar pruebas de usabilidad para medir la experiencia del usuario.
- **Eficiencia de rendimiento:** El sistema debe optimizar el uso de recursos como el tiempo de procesamiento y la memoria. Esto incluye pruebas de rendimiento bajo diferentes condiciones de carga.
- **Compatibilidad:** El software debe ser interoperable y funcionar correctamente en diferentes entornos.
- **Seguridad:** Establecer mecanismos para proteger la información, evitar accesos no autorizados y cumplir con las normativas de protección de datos (como GDPR o Ley de Protección de Datos).
- **Mantenibilidad:** Diseñar un software fácil de mantener y actualizar, con código limpio y bien documentado.
- **Portabilidad:** El sistema debe ser trasladable a diferentes entornos tecnológicos sin grandes esfuerzos.

2. ISO/IEC 12207:2017 - Ciclo de Vida del Software: Este estándar establece los procesos que deben seguirse durante todo el ciclo de vida del software, desde la planificación inicial hasta el mantenimiento posterior al despliegue. Su objetivo es garantizar que todas las actividades necesarias se realicen de manera ordenada y eficiente. Procesos recomendados:

- **Proceso de desarrollo:** Definir procedimientos para la especificación, diseño, codificación, pruebas y validación del software.
- **Gestión de proyectos:** Establecer prácticas para la planificación, monitorización, evaluación de riesgos y control del progreso del proyecto.
- **Mantenimiento:** Garantizar que los procedimientos de mantenimiento y actualización del software sean eficientes y gestionen correctamente el ciclo de vida del producto.
- **Proceso de adquisición y suministro:** Controlar que las relaciones con los proveedores y los clientes sigan un enfoque estandarizado para asegurar la calidad y cumplimiento de los requisitos.
- **Proceso de aseguramiento de la calidad:** Asegurarse de que todas las actividades cumplan con los requisitos establecidos mediante revisiones y auditorías internas.

4. ISO 9001 - Gestión de Calidad en el Desarrollo de Software La aplicación de la norma ISO 9001 en el proyecto garantiza un enfoque basado en la mejora continua y en la satisfacción del cliente. Se implementa mediante:



- **Estandarización de procesos** para asegurar consistencia y eficiencia.
- **Control de calidad mediante auditorías y revisiones periódicas.**
- **Gestión de riesgos** para prevenir problemas en el desarrollo.
- **Optimización de procesos y mejora continua** basada en métricas y feedback

3. ISO/IEC 27001:2022 - Seguridad de la Información: Este estándar se aplica para garantizar que el sistema software cumple con las normativas de seguridad en cuanto al manejo de la información sensible y de los datos de los usuarios. Se aplica especialmente si el software gestiona datos confidenciales, como información personal o financiera.

Áreas clave de ISO/IEC 27001:

- **Gestión de riesgos de seguridad:** Evaluar y mitigar los riesgos relacionados con el acceso no autorizado, pérdida o robo de datos.
- **Políticas de acceso:** Asegurar que solo los usuarios autorizados pueden acceder a la información crítica, implementando autenticación robusta y control de accesos.
- **Cifrado de datos:** Asegurar que los datos sensibles, tanto en tránsito como en reposo, estén cifrados según las mejores prácticas.

Procedimientos a aplicar

1. Proceso de Revisión de Código

- **Código limpio y documentado:** Aplicar procedimientos de revisiones de código para garantizar que esté bien estructurado, sea fácil de entender y cumpla con los estándares de codificación establecidos.
- **Análisis estático:** Utilizar herramientas de análisis estático para detectar defectos, problemas de rendimiento y posibles vulnerabilidades en el código antes de la integración.

2. Control de Versiones

- **Uso de sistemas de control de versiones (como Git):** Asegurar que cada cambio en el software esté correctamente versionado para poder rastrear la evolución del sistema y gestionar posibles retrocesos en caso de errores.
- **Branching:** Implementar una estrategia de branching clara para separar las fases de desarrollo, pruebas y producción.

3. Procedimientos de Testing Automatizado

- **Pruebas unitarias:** Garantizar que cada componente del software es probado individualmente, y que cumple con los requisitos funcionales antes de ser integrado en el sistema completo.
- **Pruebas de integración:** Validar que los distintos módulos del software funcionan correctamente cuando se integran entre sí.
- **Pruebas de regresión:** Automatizar pruebas que verifiquen que nuevas modificaciones no rompan las funcionalidades existentes.

4. Gestión de la Configuración y Control de Cambios

- **Sistema de seguimiento de cambios:** Aplicar procedimientos de solicitud de cambios para garantizar que cada modificación sea justificada, aprobada y evaluada.



- **Gestión de configuraciones:** Establecer procedimientos claros para gestionar y registrar las versiones del software y los entornos de desarrollo, asegurando consistencia y trazabilidad.

5. Auditorías Internas de Calidad

- **Auditorías periódicas:** Realizar auditorías internas para verificar el cumplimiento con las normas ISO y los procedimientos establecidos, tanto en términos de desarrollo como en la gestión de la seguridad y calidad del producto.

OBJETIVOS DE CALIDAD		
ENTREGABLE	MÉTRICA	VALOR OBJETIVO
Documento de requisitos	<ul style="list-style-type: none">● Flujo de requisitos	<ul style="list-style-type: none">● Flujo de requisitos: implementación del 100% de los requisitos
Planificación	<ul style="list-style-type: none">● Desviación de la planificación	<ul style="list-style-type: none">● Desviación de la planificación: Valores menores al 10% en actividades no críticas y menores o iguales que 0% en actividades críticas.
Backend	<ul style="list-style-type: none">● Porcentaje de casos de prueba aceptados● Porcentaje de defectos reparados	<ul style="list-style-type: none">● Porcentaje de casos de prueba aceptados: al menos 80%.● Porcentaje de defectos reparados: al menos 90%.
Frontend	<ul style="list-style-type: none">● Porcentaje de casos de prueba aceptados● Porcentaje de defectos reparados	<ul style="list-style-type: none">● Porcentaje de casos de prueba aceptados: al menos 80%.● Porcentaje de defectos reparados: al menos 90%.
Manual de usuario	<ul style="list-style-type: none">● Nivel de satisfacción del usuario mediante encuestas	<ul style="list-style-type: none">● Porcentaje de usuarios que califiquen el manual como “útil” o “muy útil”: al menos un 85%.
Pruebas	<ul style="list-style-type: none">● Cobertura de las pruebas	<ul style="list-style-type: none">● Cobertura de las pruebas: al menos 90%.
Fiabilidad	Tasa de fallos	Menos de 1% de fallos durante la operación
Usabilidad	Evaluación de usuarios	Puntuación mínima de 4/5 en encuestas de usabilidad.
Compatibilidad	Interoperabilidad	Compatible con los 5 principales navegadores y plataformas.
Mantenibilidad	Complejidad del código	



		Complejidad del código por debajo de 10 en todas las funciones clave.
Seguridad	Incidentes de seguridad	0 incidentes de seguridad críticos reportados.
Portabilidad	Esfuerzo de migración	Migración exitosa a 3 plataformas sin cambios mayores en el código.
Rendimiento	Tiempo de respuesta	Tiempo de respuesta inferior a 2 segundos en el 90% de las solicitudes.
Usabilidad	Evaluación de usuarios	Puntuación mínima de 4/5 en encuestas de usabilidad.

ACTIVIDADES DE CONTROL DE CALIDAD DE LOS ENTREGABLES

NOMBRE PAQUETE DE TRABAJO	ACTIVIDAD	RESPONSABLE
IUP	Validar requisitos.	Director del proyecto
IUP	Validar prototipos de usuario (mockups)	Equipo de calidad
IUP	Análisis de los riesgos asumidos	Equipo de calidad
S1/S2/S3	Ejecución y análisis de pruebas del sistema (unitarias, integración, rendimiento, carga y usabilidad)	Equipo de calidad
S1/S2/S3	Revisión de historias de usuario y criterios de aceptación	Equipo de calidad
S1/S2/S3	Análisis de diseño y arquitectura	Equipo de calidad
S1/S2/S3	Documentación del código	Equipo de desarrollo
S1/S2/S3	Manual de usuario	Equipo de desarrollo
S1/S2/S3	Verificación de requisitos	Director del proyecto

LISTAS DE CONTROL		
#	DURACIÓN	DESCRIPCIÓN
1	1 hora	Verificar que el código fuente siga las normas de codificación establecidas para el lenguaje de programación.
2	2 horas	Comprobar que el manual de usuario esté completo y claro.
3	3 horas	Comprobar que la interfaz de usuario cumpla con los requisitos de usabilidad y que contenga todas las funciones necesarias.
4	3 horas	Verificar que las pruebas unitarias cubran todas las funciones críticas del sistema.

PLAN DE MEJORA DE LOS PROCESOS DE GESTIÓN



ACTIVIDAD #	NOMBRE DE LA ACTIVIDAD	METODOLOGÍA O PROCEDIMIENTO	RESPONSABLE
1	Mejorar la precisión de las estimaciones	Análisis de la precisión de las estimaciones realizadas para poder aplicar mejoras en las próximas estimaciones.	Equipo de calidad
2	Automatización de pruebas funcionales	Implementar pruebas automáticas para las funcionalidades críticas	Equipo de calidad
3	Mejora en la experiencia de usuario	Realizar encuestas de satisfacción y análisis de comportamiento de usuarios	Equipo de calidad
4	Implementación de pruebas de seguridad periódicas	Realizar auditorías de seguridad para detectar vulnerabilidades	Equipo de calidad
5	Mejora en la documentación técnica y manuales	Actualizar y estandarizar la documentación técnica y los manuales de usuario	Equipo de calidad
6	Gestión eficiente de incidencias	Implementar un sistema de gestión de incidencias más avanzado para priorización y tracking	Equipo de calidad

REUNIONES A REALIZAR		
#	DURACIÓN	DESCRIPCIÓN
1	1 hora	Reuniones semanales para seguimiento del desarrollo y pruebas. Incluye revisión de las actividades de control de calidad.
2	1 hora	Reuniones de control de calidad para revisar métricas y objetivos alcanzados. Evaluación de resultados de listas de control y plan de mejora.
3	1 hora	Reunión semanal del equipo QA para revisar los resultados de las pruebas de autenticación y gestión de usuarios.
4	1 hora	Reunión para la revisión de manuales, centrada en la claridad y comprensibilidad de los manuales para el usuario final.
5	1 hora	Reunión de evaluación post-implementación para revisar la experiencia del usuario tras la evaluación de los usuarios pilotos y realizar ajustes necesarios.

INFORMES A ELABORAR		
#	DURACIÓN	DESCRIPCIÓN
1	Semanal	Actas de las reuniones semanales que incluyan los temas abordados sobre el desarrollo y las pruebas y las medidas a tomar