Plan de Gestión de Calidad del Software (SQA)

Proyecto: Sistema de Gestión de Proyectos de Titulación e Investigación (SGPTI)
Fecha: 28/11/2025

1. Introducción y Propósito

El presente Plan de Gestión de Calidad (SQA) establece el marco de trabajo para garantizar que el **SGPTI** satisfaga las necesidades explícitas e implícitas de los interesados. Este documento no solo define métricas, sino que establece una cultura de calidad preventiva para evitar los altos costos asociados al reproceso y la corrección tardía de defectos.

1.1 Objetivos de Calidad

- **Conformidad:** Asegurar que los entregables cumplan con los Requisitos Funcionales (RF) y No Funcionales (RNF) aprobados.
- **Prevención:** Detectar desviaciones en etapas tempranas mediante revisiones estáticas y pruebas continuas.
- **Satisfacción:** Garantizar una experiencia de usuario fluida y libre de errores críticos para el Comité Académico y los Estudiantes.

2. Estructura Organizativa y Responsabilidades

La calidad es responsabilidad de todo el equipo, pero se definen roles específicos para asegurar la segregación de funciones y la objetividad en las pruebas.

Rol	**Miembro(s)**	**Responsabilidades Detalladas de Calidad**
Gerente de Proyecto / QA Lead	**Bruno Parra**	• Definición de la estrategia de pruebas. • Auditoría de cumplimiento de procesos. • Decisión final de "Go/No-Go" para paso a producción. • Gestión de riesgos de calidad.

Equipo de Desarrollo y Pruebas Unitarias	**Diego** **Emiliano** **Hugo** **Axel**	• **Hugo y Axel:** Enfoque en pruebas unitarias (Jest/JUnit) y revisión de pares (Peer Reviews). • **Diego y Emiliano:** Corrección de defectos (Bug fixing) y refactorización de código. • Mantenimiento de la deuda técnica bajo control.
Analista de Calidad / Tester de Sistema	**Aquiles**	• Diseño y ejecución de casos de prueba integrales. • Pruebas de regresión tras cada entrega. • Validación de criterios de aceptación de Historias de Usuario.

3. Estándares de Calidad (Modelo ISO/IEC 25010)

Se adopta el modelo de calidad de producto de software **ISO/IEC 25010** para categorizar y evaluar los atributos del sistema de manera exhaustiva.

3.1 Adecuación Funcional

- **Completitud:** El sistema debe cubrir el 100% de los flujos críticos: Registro, Postulación, Asignación de Revisor, y Dictamen.
- **Exactitud:** Los datos guardados (nombres, fechas, estados) deben tener precisión exacta frente a la entrada del usuario. Validaciones estrictas en el *Backend*.

3.2 Eficiencia de Desempeño

- **Comportamiento temporal:** Tiempo de respuesta ≤ 3 segundos para el 95% de las transacciones (Percentil 95).
- **Utilización de recursos:** El uso de CPU del servidor no debe exceder el 70% bajo una carga de 300 usuarios concurrentes.

3.3 Seguridad (Ciberseguridad)

- **Confidencialidad:** Cifrado de datos en reposo y en tránsito (TLS 1.2+).
- **Autenticación:** Mecanismos robustos de hashing (ej. bcrypt) para contraseñas. Protección contra ataques comunes (OWASP Top 10: SQL Injection, XSS).

3.4 Compatibilidad y Portabilidad

- **Navegadores:** El sistema debe renderizarse y funcionar correctamente en las últimas versiones estables de **Chrome, Firefox, Edge y Safari**.
- **Dispositivos:** Diseño *Responsive* adaptable a resoluciones de escritorio (1920x1080) y tablets/móviles (min 375px ancho).

3.5 Fiabilidad

- **Disponibilidad:** Se requiere un tiempo de actividad (Uptime) mensual ≥ 99.5% durante el horario laboral académico.
- **Recuperabilidad:** Capacidad de restaurar el servicio en < 4 horas ante una caída crítica.

3.6 Mantenibilidad

- **Modularidad:** Arquitectura basada en componentes desacoplados para facilitar cambios futuros sin efectos colaterales.
- **Analizabilidad:** Logs de errores detallados que permitan trazar el origen de un fallo en menos de 30 minutos.

4. Métricas de Calidad e Indicadores (KPIs)

Las métricas se recopilarán quincenalmente para evaluar la salud del proyecto.

1. **Densidad de Defectos Residuales:** < 0.5 defectos de severidad media/baja por KLOC (mil líneas de código) en producción.
2. **Eficiencia de Detección de Defectos (DDE):**
   - Fórmula: $(Defectos Detectados en QA / (Defectos QA + Defectos Cliente)) * 100$
   - Meta: $> 90\%$ (El cliente debe encontrar menos del 10% de los errores totales).
3. **Cobertura de Código (Code Coverage):** $> 80\%$ de cobertura en pruebas unitarias para módulos críticos de negocio.
4. **Índice de Estabilidad de Requisitos:** Cantidad de cambios en requisitos aprobados / Total de requisitos iniciales. Meta: $< 10\%$.

5. Estrategia de Control y Aseguramiento (QC/QA)

5.1 Control de Calidad (QC) - Enfoque en el Producto

Actividades técnicas para verificar que el entregable es correcto.

- **Pruebas Unitarias:** Ejecutadas por **Hugo y Axel** localmente antes de cada commit.
- **Pruebas de Integración:** Verificar la comunicación entre el Frontend y la API, y la persistencia en Base de Datos.
- **Pruebas de Regresión:** Automatización de flujos clave (Login, Carga de Archivo) para asegurar

que nuevos cambios no rompan funcionalidades previas.
- **Revisiones de Código (Pull Requests):** Ningún código entra a la rama main sin la aprobación de al menos un par (ej. Diego revisa a Axel). Se verifican estándares de nomenclatura y optimización.

5.2 Aseguramiento de Calidad (QA) - Enfoque en el Proceso

Actividades gerenciales para asegurar que se siguen los procedimientos correctos.

- **Auditorías de Configuración:** Revisión mensual del repositorio para asegurar que no se suban credenciales o archivos binarios innecesarios.
- **Retrospectivas de Calidad:** Reunión al finalizar cada fase para analizar qué falló (Lecciones Aprendidas) y ajustar el proceso.

6. Herramientas de Gestión de Calidad

Se utilizarán las **7 Herramientas Básicas de Calidad** para el análisis de problemas:

1. **Diagrama de Ishikawa (Causa-Efecto):** Para investigar la causa raíz de defectos críticos recurrentes (ej. ¿Por qué falla la carga de archivos?).
2. **Diagrama de Pareto:** Para identificar el "20% de los módulos que causan el 80% de los errores" y priorizar su refactorización.
3. **Hojas de Verificación (Checklists):** Listas de comprobación para liberaciones (Deployment Checklist) para evitar errores humanos manuales.
4. **Gráficos de Control:** Monitoreo del rendimiento del servidor (tiempos de respuesta) para detectar anomalías estadísticas.

7. Gestión de Defectos e Incidentes

El ciclo de vida de un defecto será gestionado rigurosamente en la herramienta de tickets del proyecto.

7.1 Niveles de Severidad

- **S1 - Crítico (Bloqueante):** Impide la operación fundamental (ej. Sistema caído, pérdida de datos). *Tiempo de resolución: Inmediato (< 24h).*
- **S2 - Alto:** Funcionalidad principal afectada, sin *workaround*. *Tiempo de resolución: < 48h.*
- **S3 - Medio:** Funcionalidad parcial afectada o con solución alternativa manual.
- **S4 - Bajo:** Cosmético, ortografía, alineación.

7.2 Flujo de Resolución

1. **Identificación:** Aquiles reporta el hallazgo con pasos para reproducir, capturas y logs.
2. **Triaje:** Bruno valida y asigna prioridad.
3. **Corrección:** Asignado a **Diego, Emiliano, Hugo o Axel** según carga de trabajo.
4. **Verificación:** El desarrollador marca como "Resuelto".
5. **Cierre:** Aquiles re-ejecuta la prueba. Si pasa, se cierra. Si falla, se reabre con nota de devolución.

8. Gestión de Riesgos de Calidad

Riesgo	Probabilidad	Impacto	Estrategia de Mitigación
Pruebas insuficientes por presión de tiempo	Alta	Alto	Priorizar pruebas basadas en riesgo (módulos críticos primero). Automatizar pruebas de regresión.
Ambigüedad en los requisitos	Media	Alto	Realizar revisiones conjuntas de los requisitos con el cliente antes de comenzar el desarrollo (Shift-Left Testing).
Entorno de pruebas diferente a producción	Media	Medio	Utilizar contenedores (Docker) para garantizar paridad entre entornos de desarrollo, pruebas y producción.

9. Criterios de Aceptación y Cierre

Para declarar una fase o el proyecto como "Terminado", se debe cumplir:

1. 100% de los casos de prueba de aceptación ejecutados.
2. Cero defectos de severidad S1 o S2 abiertos.
3. Aprobación formal (firma de acta) del Comité Académico.
4. Manuales de usuario y documentación técnica entregados y validados.