

Proyecto - Blockode

Sistema constructor de ERPs

PLAN DE ASEGURAMIENTO DE CALIDAD

Producto de la materia:

Fundamentos de Ingeniería de Software

Facilitado por:

Laura Lucía Fernández Romero

Realizado por:

Angeles Arteaga Ulises Yasua
Ayala Sandoval Melina Dannaé
Macedo Hernández Andrea
Suarez Cruz Brayan Eduardo

Tabla de Contenido

1 - Objetivos.....	4
1.1 Objetivo General.....	4
1.2 Objetivos Específicos.....	4
1.2.1 Arquitectura modular y dinámica.....	4
1.2.2 Automatización del proceso de construcción.....	4
1.2.3 Personalización sin tocar el código núcleo.....	5
1.2.4 Escalabilidad y evolución constante.....	5
1.2.5 Seguridad, estándar y control.....	5
1.2.6 Acceso multi-industria.....	6
1.2.7 Analítica transversal.....	6
1.2.8 Modelo de negocio.....	6
2 - Introducción.....	7
2.1 - Objetivos de SQA.....	7
2.2 - El rol de SQA.....	7
2.3 - Responsabilidades de SQA.....	8
2.4 - Funciones de SQA.....	9
3 - Documentos relacionados.....	9
4 - Destinatarios.....	10
5 - Administración.....	11
5.1 - Organización.....	11
5.2 - Responsabilidades.....	11
5.3 - Análisis de riesgos.....	13
6 - Estándares, Prácticas, Convenciones y Mediciones.....	14
6.1 Estándares.....	14
6.1.1 Estándares de especificación y análisis.....	14
6.1.2 Estándares de arquitectura y diseño.....	14
6.1.3 Estándares de desarrollo y código fuente.....	15
6.2 Templates.....	16
6.2.1 Planificación de Proyectos.....	16
6.2.2 Control y Seguimiento de Proyectos.....	16
6.2.3 Administración de Requerimientos.....	16
6.2.4 Administración de la Subcontratación de Software.....	17
6.2.5 Administración de la Configuración de Software.....	17
6.2.6 Aseguramiento de Calidad de Software.....	17

6.3 Checklists.....	18
6.3.1 Planificación de Proyectos.....	18
6.3.2 Control y Seguimiento de Proyectos.....	18
6.3.3 Administración de Requerimientos.....	18
6.3.4 Administración de la Subcontratación de Software.....	18
6.3.5 Administración de la Configuración de Software.....	19
6.3.6 Aseguramiento de Calidad de Software.....	19
6.4 Procesos, Guías y Procedimientos.....	20
6.4.1 Planificación de Proyectos.....	20
6.4.2 Control y Seguimiento de Proyectos.....	20
6.4.3 Administración de Requerimientos.....	21
6.4.4 Administración de la Subcontratación de Software.....	21
6.4.5 Administración de la Configuración de Software.....	21
6.4.6 Aseguramiento de Calidad de Software.....	22
6.5 Mediciones.....	22
6.5.1 Mediciones del producto.....	22
6.5.2 Mediciones del Proceso de Desarrollo.....	23
6.5.3 Mediciones del Desempeño de SQA.....	24
7 - Tareas de SQA.....	25
7.1 - Tareas de infraestructura.....	25
7.2 - Productos bajo control de calidad.....	25
7.3 - Procesos bajo control de calidad.....	27
7.4 - Cronograma de actividades de SQA.....	28
8 - Soporte de SQA a clientes, proveedores y subcontratistas.....	29
9 - Registros y Reportes de SQA.....	30
9.1 - Registro de SQA.....	30
9.2 - Reporte de SQA.....	31
10 - Reportes de problemas y acciones correctivas.....	33
11 - Apéndices.....	35
11.1 - Glosario.....	35
11.2 - Historial de cambios.....	35

1 - Objetivos

1.1 Objetivo General

Desarrollar una plataforma tecnológica modular para la construcción de sistemas ERP mediante bloques configurables e integrables, que permita a cualquier empresa crear su propio ERP a la medida, sin depender de desarrollos extensos, costosos y personalizados desde cero.

1.2 Objetivos Específicos

1.2.1 Arquitectura modular y dinámica

- Diseñar una plataforma construida a partir de bloques funcionales que puedan activarse, desactivarse o reemplazarse sin afectar otros componentes.
- Implementar un modelo de integración basado en API y microservicios que permita interoperabilidad con sistemas externos.
- Permitir la creación de módulos desde plantillas predefinidas o mediante creación personalizada.

1.2.2 Automatización del proceso de construcción

- Desarrollar un constructor visual o asistido que permita armar el ERP sin programar.
- Crear un repositorio de componentes reutilizables (usuarios, permisos, inventarios, CRM, finanzas, HR, facturación, etc.).
- Permitir que los bloques puedan ser reconfigurados por reglas de negocio, sin intervención del equipo técnico.

1.2.3 Personalización sin tocar el código núcleo

- Crear un sistema de configuración empresarial basado en roles, industria y tipos de operación.
- Permitir que la lógica de negocio se defina mediante flujos, parámetros y validaciones, no por código.
- Adaptar el ERP generado a procesos simples o avanzados según la madurez digital de la empresa.

1.2.4 Escalabilidad y evolución constante

- Facilitar que el ERP construido pueda crecer con nuevos módulos sin reconstruir el sistema.
- Establecer un mecanismo de actualización de bloques sin detener la operación del cliente.
- Permitir agregar módulos propios o de terceros (marketplace de bloques).

1.2.5 Seguridad, estándar y control

- Incorporar gestión de permisos, auditoría, trazabilidad y respaldo dentro de cada ERP generado.
- Asegurar la protección de datos mediante estándares como ISO 27001 o equivalentes.
- Garantizar compatibilidad legal (facturación, impuestos, etc.) por país o región.

1.2.6 Acceso multi-industria

- Diseñar la plataforma para ser aplicable a cualquier giro: manufactura, servicios, educación, salud, retail, construcción, etc.
- Permitir que empresas consultoras, desarrolladores o startups puedan construir soluciones sobre la plataforma (modelo licencia/partners).
- Crear plantillas de ERP prearmadas por sector para acelerar la adopción.

1.2.7 Analítica transversal

- Proveer un motor de reportes y BI que se alimente de los bloques activados.
- Permitir dashboards personalizados según procesos y áreas.
- Integrar analítica predictiva en futuros bloques mediante IA.

1.2.8 Modelo de negocio

- Establecer un sistema basado en suscripción, pago por bloque, o marketplace de módulos.
- Crear esquemas de licenciamiento por usuario, por transacción o por empresa.

2 - Introducción

El SQA aporta valor al desarrollo de software al enfocarse en prevenir errores antes de que lleguen al usuario, lo que reduce costos, retrabajos y riesgos operativos. Su enfoque basado en procesos estandarizados mejora la eficiencia interna, facilita la comunicación entre equipos y asegura continuidad en los proyectos. Además, fortalece la confianza de los clientes al garantizar productos más confiables, seguros y estables. Finalmente, impulsa la mejora continua gracias a métricas y retroalimentación constante, permitiendo que la organización evolucione y mantenga su competitividad tecnológica.

2.1 - Objetivos de SQA

El principal objetivo del SQA es asegurar que el software cumpla con los requisitos y estándares de calidad, de modo que funcione de forma confiable, segura y eficiente. Su enfoque está en prevenir problemas desde las primeras etapas del desarrollo.

También busca estandarizar los procesos para que el trabajo sea consistente, claro y fácil de mantener, facilitando la colaboración entre equipos y reduciendo retrabajos.

Además, pretende aumentar la satisfacción y confianza del cliente entregando productos estables y acordes a sus expectativas, mientras promueve la mejora continua mediante el análisis de resultados y la optimización constante.

2.2 - El rol de SQA

El rol del SQA consiste en asegurar que la plataforma generadora de ERP por bloques cumpla con estándares de calidad a nivel técnico y funcional, verificando que cada módulo, bloque y proceso que integre el sistema sea confiable, seguro y consistente. El SQA revisará que las reglas y plantillas usadas para construir los ERP no generen errores, incompatibilidades o fallas al combinar módulos.

Además, implica supervisar y documentar los procesos de desarrollo para garantizar que la creación, modificación e integración de bloques se realice bajo prácticas estandarizadas. Esto permitirá que el sistema sea escalable, mantenible y fácil de actualizar sin comprometer la calidad.

El SQA actuará como un evaluador continuo, realizando pruebas, análisis de riesgos y mejoras sobre la plataforma, asegurando que lo que el sistema genera —los ERP de cada cliente— cumplan con los niveles de calidad esperados, evitando fallos antes de llegar al usuario final y fortaleciendo la confianza en el producto.

2.3 - Responsabilidades de SQA

- Definir y documentar los procesos y estándares de calidad para el desarrollo de la plataforma.
- Supervisar que la creación e integración de bloques se realice según los lineamientos establecidos.
- Realizar revisiones técnicas, auditorías de código y análisis de riesgos de forma continua.
- Ejecutar y coordinar pruebas unitarias, integrales y de compatibilidad entre bloques.
- Identificar, documentar, reportar y dar seguimiento a defectos detectados.
- Validar requisitos de seguridad, rendimiento y usabilidad en la plataforma y en los ERP generados.
- Establecer métricas para medir la calidad del proceso y del producto.
- Promover la mejora continua ajustando procesos, herramientas y prácticas de desarrollo.
- Garantizar que cada ERP generado cumpla con los niveles de calidad esperados por el cliente.

2.4 - Funciones de SQA

- planificación: definir políticas
- análisis: validar requisitos
- diseño: revisar arquitectura
- codificación: revisar código y estándares
- pruebas: Asegurar calidad funcional y técnica
- implementación: Control de entrega
- mantenimiento: Garantizar continuidad

3 - Documentos relacionados

- Calidad de software
- PSP y TSP
- Plan SQA
- Estándar de SQAP
- Especificación de requisitos de software
- Modelo Canvas

4 - Destinatarios

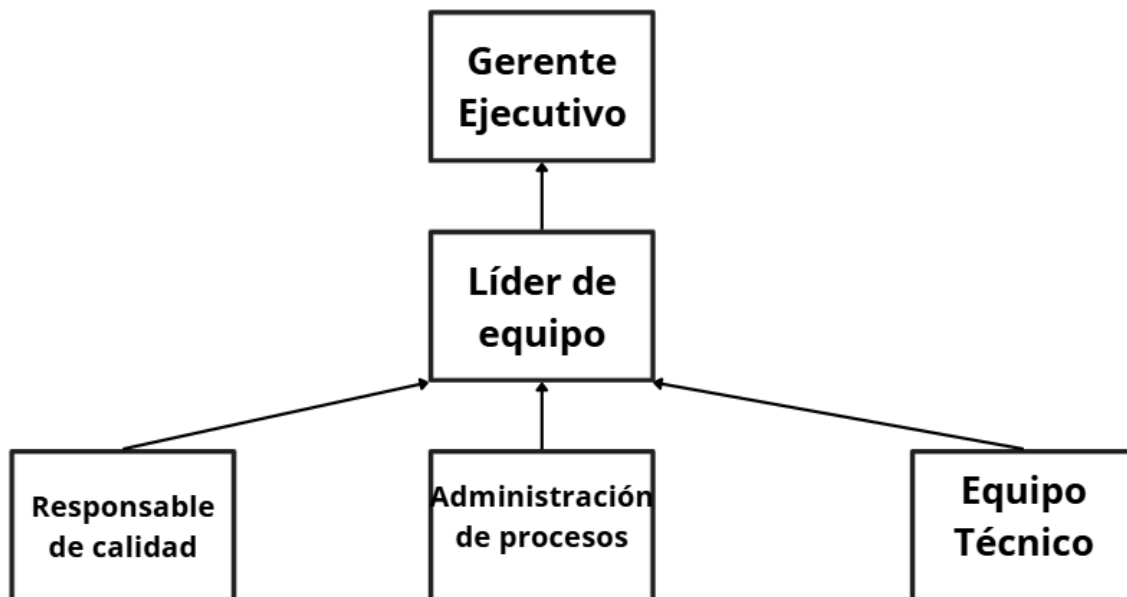
El presente documento está destinado a las siguientes personas:

- Gerente ejecutivo
- Lider de proyecto
- Responsable de calidad
- Miembro de calidad

5 - Administración

Esta sección del Plan de SQA describe aspectos relacionados con el management del equipo de SQA del proyecto. Se describen la organización del equipo de SQA, los roles, responsabilidades y tareas, el cronograma de actividades y los riesgos que pueden amenazar los objetivos de este plan.

5.1 - Organización



5.2 - Responsabilidades

ROL	RESPONSABILIDADES
Gerente Ejecutivo	<ul style="list-style-type: none">- Define presupuesto, alcance y lineamientos globales del proyecto.- Aprueba políticas de calidad y entrega final.- Autoriza cambios críticos.

Proyecto - Blockcode: Sistema constructor de ERPs

Plan de aseguramiento de calidad

	<ul style="list-style-type: none">- Evalúa riesgos estratégicos.
Líder de Equipo (Project Lead / SQA Lead)	<ul style="list-style-type: none">- Coordina al equipo y asegura que se cumplan los estándares de calidad.- Administra riesgos, métricas e informes.- Solicita aprobaciones al gerente ejecutivo.- Recibe resultados, problemas y cambios del equipo técnico, procesos y calidad.- Es el puente de comunicación entre áreas.
Administración de Procesos	<ul style="list-style-type: none">- Documenta, actualiza y controla los procesos SQA.- Define flujos de trabajo, plantillas y procedimientos.- Audita el cumplimiento del proceso.- Propone mejoras de procesos.
Equipo Técnico (Desarrolladores / QA ejecutor / Infraestructura)	<ul style="list-style-type: none">- Ejecuta pruebas, desarrolla, corrige defectos.- Registra incidentes y evidencia.- Implementa cambios técnicos aprobados.- Da retroalimentación técnica sobre riesgos o limitaciones.
Responsable de Calidad	<ul style="list-style-type: none">- Verifica cumplimiento de estándares.- Revisa entregables, pruebas y métricas.- Informa desviaciones o incumplimientos.- Recomienda aceptación o rechazo de entregas.

5.3 - Análisis de riesgos

El Aseguramiento de Calidad enfrenta riesgos que pueden comprometer la confiabilidad del sistema, el cumplimiento de los estándares o la satisfacción del cliente. Uno de los riesgos más frecuentes es la definición incompleta o ambigua de requisitos, que provoca productos que no cumplen expectativas o generan retrabajos costosos. Además, la ausencia de control en los procesos puede producir inconsistencias en la documentación, malas prácticas de codificación y fallas repetitivas.

Otro riesgo importante es la falta de comunicación entre equipos, lo cual conduce a pruebas desalineadas, correcciones tardías o decisiones mal fundamentadas. También existe riesgo en la dependencia excesiva de herramientas o recursos limitados, que puede afectar tiempos de entrega y la cobertura de pruebas. La falta de capacitación del personal o rotación de integrantes disminuye la calidad de ejecución y de análisis de defectos.

Finalmente, un riesgo crítico es no dar seguimiento a métricas y auditorías, pues sin datos es imposible medir el desempeño del proyecto, identificar causas raíz o mejorar procesos. Si SQA no monitorea, comunica y actúa sobre dichos indicadores, la calidad queda solo como criterio subjetivo y no como un valor medible y gestionado.

6 - Estándares, Prácticas, Convenciones y Mediciones

El propósito de esta sección es definir los estándares, prácticas, convenciones y mediciones utilizadas para lograr los objetivos definidos en este Plan.

6.1 Estándares

Para garantizar la calidad, consistencia y mantenibilidad del sistema a lo largo de su ciclo de vida, el proyecto adoptará un conjunto formal de estándares técnicos y metodológicos. Estos estándares regulan la forma en que se documentan los requerimientos, cómo se representan los modelos de datos y procesos, y cómo se implementan los componentes del sistema, asegurando que el ERP y sus módulos integrables cumplan con buenas prácticas internacionales. Cada estándar referido en esta sección corresponde a un documento independiente, consultable por los equipos de desarrollo, QA, arquitectura y documentación técnica.

6.1.1 Estándares de especificación y análisis

- **Estándar de Especificación de Requerimientos (IEEE 830 / ISO/IEC/IEEE 29148):** Todo el documento de requisitos —tanto funcionales como no funcionales— se redactará conforme a los criterios de claridad, verificabilidad, trazabilidad y consistencia definidos en IEEE 830. Esto establece un marco estructurado para documentar el comportamiento esperado del ERP, sus restricciones y sus interfaces API.

6.1.2 Estándares de arquitectura y diseño

- **Estándar de Diseño Detallado UML y principios SOLID:** El diseño de los módulos se documentará mediante diagramas de clases, interfaces, componentes y secuencias. Se aplicarán principios de diseño orientado a objetos (SOLID) para garantizar modularidad, escalabilidad y mantenibilidad.

- **Estándar de Nomenclatura para Componentes y APIs:** Las APIs deben seguir convenciones RESTful o gRPC (según corresponda), con reglas de nomenclatura consistentes para rutas, parámetros, modelos y eventos.

Ejemplos:

- rutas REST en formato kebab-case (`/sales-orders/create`)
- modelos en PascalCase (`CustomerOrderRequest`)
- variables y campos en camelCase (`orderId`)

6.1.3 Estándares de desarrollo y código fuente

Para asegurar cohesión entre los módulos del ERP y las capas de integración:

- **Estándar de Estilo de Programación:**
 - Uso de guías de estilo oficiales según el lenguaje elegido (por ejemplo, *Google Java Style Guide*, *Airbnb JavaScript Style Guide* o *PEP8*, según los módulos).
 - Convenciones uniformes de nombres de variables, clases, funciones, archivos y directorios.

- **Estándar de Control de Versiones (Git Flow):**

El repositorio seguirá la estructura:

- ramas principales: `main`, `develop`
- ramas auxiliares: `feature/*`, `hotfix/*`, `release/*`

Esto garantiza orden y trazabilidad en la evolución del proyecto.

- **Estándar de Documentación en Código (Docstrings / Javadoc / Swagger):**

Todo componente expuesto vía API deberá estar documentado automáticamente mediante Swagger/OpenAPI.

6.2 Templates

Esta sección define los formatos y plantillas oficiales que el equipo utilizará durante las etapas de planificación, ejecución, documentación, aseguramiento de calidad y gestión de proveedores. Las plantillas garantizan uniformidad, trazabilidad y claridad en la elaboración de documentos fundamentales para el desarrollo del sistema ERP.

6.2.1 Planificación de Proyectos

- Template del Plan de Proyecto
- Template de Cronograma del Proyecto (Gantt)
- Template para Estimación de Puntos de Función o Métricas Alternas (PF / COSMIC / Historia de Usuario)
- Template de Matriz RACI

6.2.2 Control y Seguimiento de Proyectos

- Template de Informe de Avance del Proyecto
- Template de Registro de Riesgos y Acciones Mitigadoras
- Template de Registro de Lecciones Aprendidas

6.2.3 Administración de Requerimientos

- Template de Especificación de Requerimientos de Software (SRS)
- Template de Matriz de Trazabilidad de Requerimientos
- Template de Registro de Cambios de Requerimientos

6.2.4 Administración de la Subcontratación de Software

- Template de Plan de Adquisición
- Template de Solicitud de Propuesta (RFP)
- Template de Evaluación Técnica y Económica
- Template de Encuesta a Proveedor / Evaluación de Satisfacción
- Template de Acuerdo de Nivel de Servicio (SLA) para APIs
- Template de Contrato de Integración de API

6.2.5 Administración de la Configuración de Software

- Template del Plan de Gestión de Configuraciones (SCM Plan)
- Template de Registro de Configuración / Baselines
- Template de Solicitud de Cambio (CR – Change Request)

6.2.6 Aseguramiento de Calidad de Software

- Template de Checklist de Calidad
- Template de Registro SQA
- Template de Reporte de Actividades SQA
- Template del Plan de Pruebas (Test Plan) según IEEE 829
- Template de Casos de Prueba (Test Case Specification)
- Template de Reporte de Fallos (Bug Report)

6.3 Checklists

A continuación se listan los checklists oficiales que serán utilizados durante las revisiones formales del proyecto. Cada checklist corresponde a un tipo de documento, actividad o estándar definido dentro del proceso de desarrollo.

6.3.1 Planificación de Proyectos

- Checklist del Plan de Proyecto
- Checklist del Cronograma
- Checklist de Matriz RACI
- Checklist de Estimación de Esfuerzo y Métricas

6.3.2 Control y Seguimiento de Proyectos

- Checklist del Informe de Avance del Proyecto
- Checklist de Gestión de Riesgos
- Checklist de Lecciones Aprendidas

6.3.3 Administración de Requerimientos

- Checklist de Especificación de Requerimientos de Software (SRS)
- Checklist de Trazabilidad de Requerimientos
- Checklist de Gestión de Cambios de Requerimientos

6.3.4 Administración de la Subcontratación de Software

- Checklist de Estrategia de Contratación
- Checklist de Solicitud de Propuesta (RFP)
- Checklist de Evaluación del Proveedor
- Checklist de Evaluación Técnica y Económica
- Checklist de Monitoreo de Proveedores

- Checklist de Aceptación de Producto
- Checklist de Prueba de Aceptación
- Checklist de SLA para Integraciones vía API

6.3.5 Administración de la Configuración de Software

- Checklist del Plan de SCM
- Checklist de Baselines y Control de Versiones
- Checklist de Solicitudes de Cambio (CR)

6.3.6 Aseguramiento de Calidad de Software

- Checklist del Plan de SQA
- Checklist de Auditoría de Calidad
- Checklist de Casos de Prueba y Datos de Prueba
- Checklist del Reporte de Fallos
- Checklist de Pruebas de Integración con APIs

6.4 Procesos, Guías y Procedimientos

A continuación se listan los procesos, guías y procedimientos oficiales que serán utilizados durante el proyecto. Cada documento orienta la ejecución de actividades clave y será verificado por el área de SQA para asegurar su cumplimiento.

6.4.1 Planificación de Proyectos

- Proceso de Planificación de Proyectos
 - Guía de Uso del Plan de Proyecto y Estructura de Entregables
 - Guía de Elaboración de Cronogramas (MS Project / Gantt Online)
- Descripción del Método de Cálculo de Puntos de Función
 - Guía de Estimación de Esfuerzo mediante Puntos de Función (PF)
 - Guía de Estimación de Esfuerzo con COCOMO II
 - Guía del Método de Putnam para curvas de esfuerzo-tiempo
- Procedimiento de Identificación y Priorización de Riesgos del Proyecto

6.4.2 Control y Seguimiento de Proyectos

- Proceso de Control y Seguimiento de Proyectos
 - Guía para Elaboración del Informe de Avance
 - Guía de Gestión y Seguimiento de Riesgos
 - Guía de Gestión de Incidentes y Problemas del Proyecto
- Manual de Procedimientos de la Oficina de Proyectos (PMO)
 - Guía de Registro y Gestión de Lecciones Aprendidas
 - Guía de Uso de Paneles/KPIs de Seguimiento (Dashboards)

6.4.3 Administración de Requerimientos

- Proceso de Administración de Requerimientos
 - Guía de Especificación de Requerimientos (IEEE 830 adaptado)
- Procedimiento de Control de Cambios de Requerimientos
 - Guía de Gestión de Trazabilidad de Requerimientos
- Procedimiento para Validación y Aprobación de Requerimientos con Stakeholders
 - Guía para Elaboración de Casos de Uso y Escenarios UML

6.4.4 Administración de la Subcontratación de Software

- Proceso de Administración de Subcontratación de Software
 - Guía de Revisión de Solicitudes de Propuesta (RFP)
- Procedimiento de Evaluación Técnica y Económica de Proveedores
- Procedimiento de Integración y Validación de Componentes Externos
 - Guía de Gestión de SLA y Contratos con Terceros

6.4.5 Administración de la Configuración de Software

- Proceso de Administración de la Configuración de Software (SCM)
 - Guía de Uso del Sistema de Control de Versiones (Git)
 - Guía de Gestión de Branching, Releases y Hotfixes
- Procedimiento de Generación de Baselines
 - Guía de Administración de Ambientes (Desarrollo, QA, Producción)
- Procedimiento de Backup y Recuperación de Configuración

6.4.6 Aseguramiento de Calidad de Software

- Proceso de SQA
 - Guía de Elaboración de Planes de Prueba
 - Guía de Diseño de Casos de Prueba
 - Guía de Pruebas de Integración para APIs y Microservicios (*añadido*)
- Procedimiento de Registro de No Conformidades
- Procedimiento de Auditorías Internas de Calidad
 - Guía de Uso de Reportes de Actividades de SQA

6.5 Mediciones

Las mediciones definidas para este proyecto permitirán evaluar de manera continua la calidad del producto, la eficiencia del proceso de desarrollo y el desempeño de las actividades de aseguramiento de calidad. Estas mediciones serán registradas en los informes periódicos del proyecto y su análisis servirá como base para la toma de decisiones, la identificación de desviaciones y la implementación de acciones correctivas.

6.5.1 Mediciones del producto

Las mediciones aplicadas al producto permiten evaluar la calidad técnica del ERP y sus componentes, tanto internos como integraciones vía API. Entre las mediciones consideradas para este proyecto se encuentran:

- **Número total de defectos detectados**, clasificados por severidad y módulo.
- **Densidad de defectos** (defectos por cada 1,000 líneas de código o por punto de función).
- **Cantidad de líneas de código nuevas y modificadas**, incluyendo componentes reusados.

- **Porcentaje de código cubierto por pruebas automáticas** (unitarias, integración y API testing).
- **Tasa de fallos en entornos preproductivos**, específicamente en integraciones con sistemas externos.
- **Tiempo promedio de respuesta del sistema** bajo condiciones normales y de carga.
- **Porcentaje de transacciones que cumplen con los requisitos de rendimiento** establecidos (p. ej., "95% de las transacciones procesadas en < 1s").
- **Porcentaje de disponibilidad del sistema** medido en los entornos del proyecto.

Estas mediciones se utilizarán para determinar la madurez del producto, controlar su estabilidad y garantizar el cumplimiento de los requerimientos no funcionales definidos.

6.5.2 Mediciones del Proceso de Desarrollo

Las mediciones del proceso permiten identificar si el desarrollo avanza conforme al plan establecido y si el proceso en sí está produciendo resultados de forma eficiente.

Las mediciones consideradas incluyen:

- **Duración prevista vs. duración real** de cada fase del ciclo de vida del proyecto.
- **Esfuerzo estimado vs. esfuerzo real** reportado en horas-hombre por fase.
- **Costo estimado vs. costo real** del proyecto.
- **Número de cambios en requerimientos** y su impacto en cronograma y esfuerzo.
- **Índice de retrabajo**, medido como horas empleadas en corregir defectos o rediseñar módulos.
- **Velocidad del equipo** (en puntos de función o equivalente) por iteración.
- **Cumplimiento del cronograma** respecto a los hitos definidos.

- **Tiempo promedio de resolución de incidencias del proceso** (bloqueos, dependencias, retrasos).

Estas mediciones permiten evaluar la efectividad del proceso, identificar cuellos de botella y mejorar la planificación continua del proyecto.

6.5.3 Mediciones del Desempeño de SQA

Las mediciones de SQA permiten evaluar si las actividades de aseguramiento de calidad se ejecutan conforme al Plan de SQA (SQAP) y si están contribuyendo a mejorar la calidad del producto y del proceso. Las mediciones incluidas son:

- **Cumplimiento de hitos de SQA** comparados con lo planificado en el SQAP.
- **Número de revisiones realizadas** (documentación, código, API contracts, pruebas) vs. las programadas.
- **Cantidad de desviaciones observadas por SQA**, clasificadas por severidad y tipo.
- **Trabajo completado vs. trabajo planificado** para actividades de SQA.
- **Esfuerzo real vs. esfuerzo planificado** para el equipo de SQA.
- **Presupuesto gastado en SQA** comparado con lo asignado en el SQAP.
- **Porcentaje de no conformidades identificadas y resueltas** dentro del plazo acordado.
- **Nivel de cumplimiento de estándares y procedimientos** definidos por el proyecto.

Estas mediciones aseguran una supervisión continua de la calidad y permiten evaluar el impacto de SQA sobre la mejora del proceso y del producto final.

7 - Tareas de SQA

7.1 - Tareas de infraestructura

Las tareas a llevar a cabo por el equipo de SQA para construir y mantener su infraestructura, esencial para la ejecución de este Plan, incluyen:

- **Capacitación y Entrenamiento:** Obtención y capacitación del personal de SQA en metodologías ágiles, pruebas de microservicios y API, y en el uso de herramientas específicas para la plataforma Blokcode (constructores visuales, repositorios de componentes).
- **Recopilación y/o Construcción de Activos de Proceso:** Recopilación, revisión y/o creación de los estándares, *templates*, procesos, guías, procedimientos y *checklists* definidos en la Sección 6.
- **Definición e Implantación de Métricas:** Definición e implantación de las métricas de SQA y del producto (Sección 6.5), incluyendo la configuración de herramientas para su recolección y análisis.
- **Capacitación al Equipo de Proyecto:** Entrenamiento al Equipo Técnico y Líderes sobre el rol de SQA, los estándares de codificación (Git Flow) y el procedimiento de Reporte de Fallos y No Conformidades (Sección 10).
- **Actualizaciones del SQAP:** Realización de revisiones y actualizaciones previstas al Plan de SQA en hitos clave del proyecto, especialmente después de fases de integración mayor de bloques funcionales o cambios en la arquitectura de microservicios.

7.2 - Productos bajo control de calidad

En esta sección se listan los productos de *software* (documentos, código, modelos) que serán revisados por el equipo de SQA, el *Activo de Proceso* que se utilizará como base

Proyecto - Blockcode: Sistema constructor de ERPs

Plan de aseguramiento de calidad

de revisión (estándares, *templates* o *checklists*), el proceso de revisión, y el rol de SQA asignado.

Producto (Documento)	Activos de Proceso a utilizar	Proceso a seguir – Participantes	Rol de SQA asignado
Especificación de Requerimientos	Estándar de Especificación de Requerimientos (IEEE 830), Checklist de Requerimientos y Trazabilidad	Revisión/Inspección	Miembro de Calidad (MQA/QC)
Especificación de Arquitectura y Diseño	Estándar de Arquitectura y Diseño, Checklist de Diseño de Arquitectura	Revisión/Auditoría	Miembro de Calidad (MQA/QC)
Código Fuente de Módulos/Bloques	Estándar de Estilo de Programación, Checklist de Código Fuente	Inspección de Código/Auditoría	Miembro de Calidad (MQA/QC)
Plan de Pruebas (Test Plan)	Template del Plan de Pruebas (IEEE 829)	Revisión y Aprobación	Responsable de Calidad (RQA/QC)
Casos de Prueba	Template de Casos de Prueba, Checklist de Casos de Prueba	Revisión	Miembro de Calidad (MQA/QC)

Plan de aseguramiento de calidad

Registro de Configuración / Baselines	Checklist de Baselines y Control de Versiones	Auditoría de Configuración	Miembro de Calidad (MQA/QC)
Reporte de Fallos (Bug Report)	Template de Reporte de Fallos (Bug Report)	Seguimiento de No Conformidades	Miembro de Calidad (MQA/QC)

7.3 - Procesos bajo control de calidad

Se indican los procesos críticos que serán auditados por SQA para verificar su cumplimiento y asegurar la aplicación de los estándares definidos en la Sección 6.4.

Proceso	Activos de Proceso a utilizar	Actividad de SQA	Rol de SQA asignado
Administración de Requerimientos	Proceso de Administración de Requerimientos, Procedimiento de Control de Cambios de Requerimientos	Auditoría para verificar la trazabilidad, gestión de cambios y validación con <i>stakeholders</i>	Miembro de Calidad (MQA/QC)
Administración de la Configuración (SCM)	Proceso de SCM, Checklist del Plan de SCM	Auditoría del cumplimiento de las políticas de control de versiones (<i>Git</i>)	Miembro de Calidad (MQA/QC)

Plan de aseguramiento de calidad

		<i>Flow) y baselines</i>	
Aseguramiento de Calidad de Software (SQA)	Proceso de SQA, Procedimiento de Auditorías Internas de Calidad	Auditoría periódica al cumplimiento del propio Plan de SQA	Responsable de Calidad (RQA/QC)
Reportes de Problemas y Acciones Correctivas	Procedimientos de la Sección 10	Monitoreo y seguimiento de la resolución de no conformidades	Miembro de Calidad (MQA/QC)
Administración de Subcontratación	Proceso de Administración de Subcontratación de Software	Auditoría de la evaluación técnica y el cumplimiento de SLA para APIs y servicios externos	Miembro de Calidad (MQA/QC)

7.4 - Cronograma de actividades de SQA

El cronograma de SQA es el plan a seguir para todas las revisiones y auditorías de SQA planificadas, incluyendo otros hitos específicos como la finalización de estándares, la configuración de la herramienta de mediciones o las tareas de infraestructura de SQA.

El Cronograma de Actividades de SQA se encuentra detallado y gestionado en el documento **Cronograma del Proyecto**. Las actividades de SQA se identifican en dicho cronograma con el prefijo "QA" (ej.: "QA: Revisión Plan de SCM") e incluyen:

- Hitos de infraestructura (p. ej., "Implantación de Métricas 6.5.3").
- Fechas de las revisiones de documentos (p. ej., "Revisión Plan de Pruebas").
- Frecuencia de las auditorías de proceso (p. ej., "Auditoría Mensual de SCM").
- Hitos de *Testing* (p. ej., "Supervisión de Pruebas de Integración con API" y "Testigo de Pruebas de Aceptación (UAT)").

8 - Soporte de SQA a clientes, proveedores y subcontractistas

Para el proyecto Blokode ERP se establecen mecanismos de aseguramiento de calidad orientados a garantizar que todos los productos y servicios generados por clientes, proveedores y subcontractistas cumplan con los estándares definidos para el proyecto.

En el caso de los clientes, SQA participará en la revisión y validación de los requerimientos iniciales, verificando su claridad, completitud y trazabilidad. Se coordinarán sesiones de revisión conjunta para asegurar que las personalizaciones solicitadas estén correctamente documentadas y alineadas con los lineamientos funcionales del sistema. Asimismo, durante la fase de pruebas de aceptación del usuario (UAT), SQA actuará como testigo de las verificaciones para confirmar el cumplimiento de los criterios definidos.

Respecto a proveedores tecnológicos (como servicios de hosting, autenticación, pasarelas de pago o APIs externas), SQA revisará la documentación técnica, certificaciones y garantías ofrecidas. También realizará auditorías basadas en checklists de calidad, enfocadas en la disponibilidad, seguridad, compatibilidad y estabilidad de los servicios integrados a Blokode. Todas las integraciones serán sometidas a pruebas específicas de interoperabilidad y rendimiento.

En el caso de subcontractistas que contribuyan con módulos adicionales al ERP, SQA validará el cumplimiento de los estándares de codificación, documentación técnica, testing y trazabilidad establecidos para el proyecto. Toda entrega deberá incluir el código fuente correspondiente, manual técnico, evidencia de pruebas y bitácoras de incidencias resueltas. Se realizarán auditorías periódicas para verificar la adopción del proceso de desarrollo definido y la calidad de los productos entregados.

9 - Registros y Reportes de SQA

El equipo de SQA generará, mantendrá y resguardará todos los registros e informes relacionados con las actividades de calidad del proyecto Blokode ERP. Estos documentos permitirán evidenciar el estado real de los procesos, productos y actividades de verificación, así como las desviaciones observadas a lo largo del ciclo de vida del proyecto.

9.1 - Registro de SQA

Propósito:

Registrar eventos relevantes asociados a actividades de aseguramiento de calidad, incluyendo validaciones completadas o desviaciones detectadas en productos o procesos.

Roles:

- **Gerencia Superior:** destinatario del registro.
- **Grupo de QA:** originador del informe.
- **Copias:** Líder de Proyecto y responsables técnicos correspondientes.

Contenido:

El registro incluirá la fecha del evento, descripción clara del hallazgo, evidencia asociada (como checklists o documentos revisados), participantes involucrados, clasificación de la conformidad o desviación, y recomendaciones o acciones sugeridas para su tratamiento.

Seguimiento:

Cuando se trate de una desviación, SQA dará seguimiento a las acciones correctivas definidas, verificará su implementación y asegurará que exista evidencia de cierre. Todos los registros se almacenarán en el repositorio oficial del proyecto para consulta y auditorías futuras.

Registro de SQA	
Fecha	
Para	
Copia	
De	
Descripción	
Conformidad	
Observaciones	
Seguimiento	
Necesario	
Esfuerzo	
Insumido	
Tiempo	
Insumido	

9.2 - Reporte de SQA

Propósito:

Proveer un resumen periódico de las actividades de calidad realizadas, los productos revisados, el estado de los procesos y los puntos pendientes de resolución. Este reporte también incluirá las métricas de calidad establecidas en el proyecto.

Roles:

- **Gerente de TI:** destinatario del reporte.
- **Grupo de QA:** responsable de elaborarlo.
- **Copias:** Jefes de Área, Gerencia Superior y Líder del Proyecto.

Contenido del reporte:

El reporte incluirá las actividades de QA realizadas durante el periodo, el estado general de los procesos del área, observaciones y riesgos identificados, resultados de auditorías y revisiones, número de no conformidades registradas, métricas planificadas vs. reales (tiempo, esfuerzo y revisiones realizadas), así como evidencia documental de las validaciones efectuadas.

Reporte de Actividades de SQA		
Fecha		
Area		
Actividades de QA del área durante este mes		
Estado de los procesos del área		
Observaciones sobre el área		
Mediciones de QA	Planificado	Real
Tiempo Insumido		
Esfuerzo Insumido		
Hitos logrados		
Revisiones y Auditorías realizadas		
Pruebas de Aceptación realizadas		
Número de Informes de no-conformidad por mes		
Firma _____		
Aclaración		

10 - Reportes de problemas y acciones correctivas

El procedimiento para reportar, monitorear y resolver problemas detectados en los productos o procesos del proyecto Blokcode ERP se basa en un enfoque sistemático de registro, evaluación y control.

Registro del problema:

Toda desviación, defecto o incumplimiento identificado será documentado a través de un Registro de SQA, detallando severidad, origen, descripción y evidencia.

Clasificación:

El problema será clasificado según su severidad (crítica, alta, media o baja), dependiendo del impacto potencial en la funcionalidad o estabilidad del ERP.

Asignación:

El responsable técnico o funcional recibirá el registro junto con el plazo para su resolución.

Acción correctiva:

El responsable implementará la corrección necesaria y documentará la acción aplicada.

Verificación:

SQA revisará que la acción correctiva haya sido implementada correctamente mediante pruebas o revisión documental.

Cierre:

El hallazgo será marcado como cerrado únicamente cuando SQA confirme su resolución total.

Escalamiento:

Si un problema no puede ser resuelto en el nivel operativo del proyecto, será elevado a la Gerencia Superior para su tratamiento y seguimiento, garantizando la resolución según los lineamientos del proceso de SQA.

11 - Apéndices

11.1 - Glosario

ERP (Enterprise Resource Planning):

Sistema de planificación de recursos empresariales que integra diversos módulos de gestión para automatizar y centralizar operaciones.

SQA (Software Quality Assurance):

Conjunto de actividades destinadas a asegurar la calidad del software y de los procesos utilizados para desarrollarlo.

SCM (Software Configuration Management):

Proceso encargado del control de versiones, cambios y entregables técnicos del proyecto.

UAT (User Acceptance Testing):

Pruebas realizadas por usuarios finales para validar que el sistema cumple con los requisitos establecidos.

Módulo:

Componente funcional del ERP que puede activarse o desactivarse según las necesidades del cliente dentro de la plataforma Blokode.

11.2 - Historial de cambios

Revisión: 1.0

Autor: Dannae Ayala

Fecha: 12/07/2025

Descripción: Primera versión creada.