

Plan de SQA

Gestión de calidad - 2025

Agustin Collareda y Hugo Frey





La Calidad del Software tiene como objetivo brindar la confianza de que el producto final logrará satisfacer los requisitos del cliente.



Tabla de contenido

Propósito	5
Referencias	5
Gestión	5
Organización	5
Actividades	5
Ciclo de vida del software cubierto por el Plan	5
Actividades de calidad a realizarse	5
Revisar cada producto	6
Revisar el ajuste al proceso	6
Realizar Revisión Técnica Formal (RTF)	7
Asegurar que las desviaciones son documentadas	7
Relaciones entre las actividades de SQA y la planificación	7
Responsables	7
Documentación	8
Propósito	8
Documentación mínima requerida	8
Planificación en Trello	8
Especificación de requerimientos del software	8
Descripción del diseño del software	9
Documentación de pruebas	10
Documentación de usuario	10
Plan de Gestión de configuración	10
Propósito	11
Resumen	11
Organización, Responsabilidades	11
Reportes y Auditorías	11
Otros documentos	11
Estándares, prácticas, convenciones y métricas	12
Objetivos	12
Métricas de proceso	12
Métricas de proyecto	13
Métricas de producto	13
Estándar de documentación	14
Estándar de verificación y prácticas	15
Estándares y pautas a seguir	15
Revisiones y auditorías	16
Objetivo	16
Requerimientos mínimos	16
Revisión de requerimientos	16
Revisión de diseño preliminar	16
Revisión de diseño crítico	16

Auditoría funcional	16
Auditoría física	17
Auditorías internas al proceso	17
Revisiones de gestión	17
Revisión del Plan de gestión de configuración	17
Revisión Post Mortem	17
Agenda	17
Otras revisiones	17
Revisión de documentación de usuario	17
Verificación	18
Reporte de problemas y acciones correctivas	18
Herramientas, técnicas y metodologías	18
Gestión de riesgos	18
Anexos	20
Formulario de Pedidos y Detección de Cambios	20

Plan de SQA

Propósito

El plan de aseguramiento de calidad es aplicable a todos los proyectos de desarrollo de software de los autores. Incluye los elementos del software que serán evaluados y su finalidad dentro del proyecto. Además, especifica qué etapas del ciclo de vida del software estarán cubiertas por el plan.

Por otro lado, se debe tener una noción fresca de la definición de la calidad del software. Esta tiene como objetivo brindar la confianza de que el producto final logrará satisfacer los requisitos del cliente.

Referencias

Documento plan de gestión de riesgos

Gestión

Organización

El grupo conformado para la cátedra de Gestión de Calidad es:

- Agustin Collareda.
- Hugo Frey.

Actividades

Ciclo de vida del software cubierto por el Plan

Este plan cubre las siguientes actividades del ciclo de vida:

- Gestión de proyecto.
- Relevamiento de requerimientos.
- Análisis de los requerimientos.
- Diseño del software.
- Implementación.
- Integración y pruebas.
- Mantenimiento.

Actividades de calidad a realizarse

[Las tareas a ser llevadas a cabo deberán reflejar las evaluaciones a realizar, los estándares a seguir, los productos a revisar, los procedimientos para la elaboración de los distintos productos]

y los procedimientos para informar de los defectos detectados a sus responsables y realizar el seguimiento de los mismos hasta su corrección.

Las actividades que se realizarán son:

Revisar cada producto

Revisar el ajuste al proceso

Realizar Revisión Técnica Formal (RTF)

Asegurar que las desviaciones son documentadas.]

Revisar cada producto

Los productos clave son:

- Especificación de requerimientos.
- Modelos representativos obtenidos a través del análisis.
- Diseño del producto software.
- Código fuente.
- Planes asociados a la calidad y la gestión.

Estos se deberán revisar y generar un informe el cual contenga las anomalías encontradas junto a las posibles correcciones que se podrían realizar y los artefactos que deberán ser revisados, es decir, los artefactos derivados de estos productos claves. A su vez, se debe verificar que se hayan realizado las correcciones de las anomalías encontradas.

Revisar el ajuste al proceso

[En esta actividad se revisan los productos que se definieron como claves para verificar el cumplimiento de las actividades definidas en el proceso. Con el fin de asegurar la calidad en el producto final del desarrollo, se deben llevar a cabo revisiones sobre los productos durante todo el ciclo de vida del software.

Se debe recoger la información necesaria de cada producto, buscando hacia atrás los productos previos que deberían haberse generado, para poder establecer los criterios de revisión y evaluar si el producto cumple con las especificaciones.

Esta información se obtiene de los siguientes documentos:

Plan del Proyecto

Plan de la iteración

Plan de Verificación

Antes de comenzar, se debe verificar en los informes de revisión previos que todas las desviaciones fueron corregidas, si no fuese así, las faltantes se incluyen para ser evaluadas.

Como salida se obtiene el Informe de revisión de ajuste al Proceso, este informe debe ser distribuido a los responsables de las actividades y se debe asegurar de que son consientes de desviaciones o discrepancias encontradas.]

Realizar Revisión Técnica Formal (RTF)

El objetivo de la RTF es descubrir errores en la función, la lógica ó la implementación de cualquier producto del software, verificar que satisface sus especificaciones, que se ajusta a los estándares establecidos, señalando las posibles desviaciones detectadas.

Es un proceso de revisión riguroso, su objetivo es llegar a detectar lo antes posible, los defectos o desviaciones en los productos que se van generando a lo largo del desarrollo.

Las reuniones que se llevan a cabo por estas RFT no deben durar más de dos horas y deberán ser planificadas con anterioridad, es decir, se deberá juntar todos los artefactos necesarios junto con una lista de chequeo la cual también deberá ser rellenada con anterioridad. En la reunión no se buscan soluciones sino informar las anomalías encontradas.

Asegurar que las desviaciones son documentadas

[Las desviaciones encontradas en las actividades y en los productos deben ser documentadas y manejadas de acuerdo a un procedimiento establecido.

Se debe chequear que los responsables de cada plan los modifiquen cada vez que sea necesario, basados en las desviaciones encontradas.]

Relaciones entre las actividades de SQA y la planificación

Actividad	Semana
[Actividad 1]	[Semana cuando se realiza]
[Actividad 2]	[Semana cuando se realiza]

Responsables

[Identificar los distintos responsables de cada actividad identificada.]

[Dichas actividades son: las revisiones, el análisis causal, el mantener una base de datos de los errores encontrados a lo largo del desarrollo e informes.

Para la puesta en marcha de estas actividades se deberá seguir el siguiente ciclo de prevención:

- *Ejecutar una tarea*
- *Realizar un control de revisiones, para decidir la aceptación o necesidad de corrección de dicha tarea.*
- *En caso de que en la revisión se presenten errores se realizara un análisis causal para determinar el motivo de estos. Se analiza un determinado error, se establece una hipótesis de su posible causa, se trata de deducir en qué momento se produjo y por qué. Luego se deberá realizar la corrección del mismo y tomar una acción correctiva con el fin de eliminar la causa del problema.*
- *El resultado del análisis causal es ingresado a una base de datos para mantener un registro y poder obtener métricas.*
- *Se comienza nuevamente el ciclo ejecutando la tarea.]*

Documentación

Propósito

Se deben establecer los criterios de calidad y los métodos de revisión utilizados para comprobar el cumplimiento de los mismos de cada uno de los documentos claves del desarrollo del producto de software.

Documentación mínima requerida

Planificación en Trello

El desarrollo de un sistema tendrá atrás una planificación que debe contener las tareas próximas que se deben realizar. Una tarea tiene asociado un título representativo con una posible descripción para más detalles. A su vez, cada tarea puede tener subtareas, siguiendo la filosofía de divide y vencerás, lo que permite al equipo poder visualizar el avance para tener mucha más motivación.

Todo esto se hará al estilo ágil con una pizarra en un tablero con la herramienta Trello. Cada miembro es responsable de hacer las tareas y completarlas en un tiempo establecido.

Especificación de requerimientos del software

El documento de especificación de requerimientos deberá describir, de forma clara y precisa, cada uno de los requerimientos esenciales del software además de las interfaces externas.

El cliente deberá obtener como resultado del proyecto una especificación adecuada a sus necesidades en el área de alcance del proyecto, de acuerdo al compromiso inicial del trabajo y a los cambios que este haya sufrido a lo largo del proyecto, que cubra aquellos aspectos que se haya acordado detallar con el cliente.

La especificación debe:

- Ser completa:
 - Externa, respecto al alcance acordado.
 - Internamente, no deben existir elementos sin especificar.
- Ser consistente, no puede haber elementos contradictorios.
- Ser no ambigua, todo término referido al área de aplicación debe estar definido en un glosario.
- Ser verificable, debe ser posible verificar siguiendo un método definido, si el producto final cumple o no con cada requerimiento.
- Estar acompañada de un detalle de los procedimientos adecuados para verificar si el producto cumple o no con los requerimientos.
- Incluir requerimientos de calidad del producto a construir.

Los requerimientos de calidad del producto a construir son considerados dentro de atributos específicos del software que tienen incidencia sobre la 'calidad en el uso'.

Los atributos de calidad que se buscan son:

- Funcionalidad
- Adecuación a las necesidades
- Seguridad de los datos
- Confiabilidad
- Madurez
- Tolerancia a faltas
- Usabilidad
- Comprensible
- Aprendible
- Operable
- Eficiencia
- Mantenibilidad
- Modificable
- Estable, no se producen efectos inesperados luego de modificaciones
- Verificable

Cada uno de estos atributos debe cumplir con las normas y regulaciones aplicables a cada uno.

Descripción del diseño del software

El documento de diseño especifica como el software será construido para satisfacer los requerimientos.

Deberá describir los componentes y subcomponentes del diseño del software, incluyendo interfaces internas. Este documento deberá ser elaborado primero como Preliminar y luego será gradualmente extendido hasta llegar a obtener el Detallado.

El cliente deberá obtener como resultado del proyecto el diseño de un producto de software que cubra aquellos aspectos que se haya acordado con el cliente incorporar al diseño, en función de la importancia que estos presenten y de sus conexiones lógicas.

Algunos de los criterios para este documento son:

- Todo elemento del diseño debe contribuir a algún requerimiento.
- La implementación de todo requerimiento a incorporar debe estar contemplada en por lo menos un elemento del diseño.
- Ser consistente con la calidad del producto.

Documentación de pruebas

Esta documentación debe contener anexada las pruebas unitarias y ejecutadas las pruebas de validación con el fin de que se haga un proceso de pruebas correcto y efectivo para garantizar la confiabilidad del software.

En este caso, se cuenta con una planilla de excel en el cual se van ingresando las pruebas de validación, esta debe contar con un identificador para cada prueba, el responsable de la misma, una descripción de los hechos observados, una serie de pasos para verificar su uso y el estado.

A su vez, se rellenará un documento llamado informe de verificación y validación el cual garantiza que los errores descubiertos no se pierdan y puedan ser rastreados para poder corregirlos.

Documentación de usuario

La documentación de usuario debe especificar y describir los datos y entradas de control requeridos, así como la secuencia de entradas, opciones, limitaciones de programa y otros ítems necesarios para la ejecución exitosa del software.

Todos los errores deben ser identificados y las acciones correctivas descritas.

Como resultado del proyecto el cliente obtendrá un manual de usuario y un manual de instalación.

Plan de Gestión de configuración

Propósito

Este plan tiene como fin controlar la entrega y el cambio de los elementos a través del ciclo de vida del sistema, y almacenar el estado de los elementos y de las peticiones de cambio.

Resumen

La Gestión de Configuración identifica los elementos de un proyecto de desarrollo de software (especificaciones, requisitos, arquitecturas, código, planes, etc.) proporcionando el control de los elementos identificados y la generación de informes de estado de la configuración, consiguiendo, al mismo tiempo, claridad de gestión, al asignar responsabilidades al personal encargado de las tareas de control a lo largo del ciclo de vida del producto.

Organización, Responsabilidades

Al tratarse de un grupo pequeño de desarrollo, la administración de gestión de versiones será un trabajo realizado de forma cooperativa. Los cambios a realizar serán solicitados y tratados mediante comunicación directa entre los integrantes del equipo de desarrollo, ya sea por Whatsapp, Discord u otras herramientas de mensajería utilizadas.

Reportes y Auditorias

Se realizará las siguientes auditorias:

Auditoria Funcional: *Cuyo objetivo es comprobar que se han completado todas las pruebas necesarias para el / los ECS auditados, y que, teniendo en cuenta los resultados de los tests, se puede afirmar que el / los ECS satisfacen los requisitos que se impusieron sobre él.*

Revisión formal de certificación: *Cuyo objetivo es certificar que el / los ECS se comportan correctamente en su entorno operativo.*

Otros documentos

[Esta sección puede contener otros documentos que se identifiquen de incidencia en la calidad del producto a desarrollar, por ejemplo:

- *Plan de desarrollo*
- *Plan de proyecto*
- *Manual de estándares y procedimientos.]*

Estándares, prácticas, convenciones y métricas

Objetivos

Existen dos objetivos importantes que se persiguen dentro del programa de métricas:

Documentar las metas a la hora de establecer un programa de métricas. Esto tiene sentido a la hora de decidir exactamente qué debe lograrse antes de gastar recursos estableciendo un programa de este tipo.

Identificar la información (la métrica) necesaria para lograr estas metas y establecer el marco de referencia de donde puede ser obtenida.

El cometido de los ocho pasos es crear un proceso a través del cual un programa corriente de métrica puede ser utilizado como una herramienta estratégica de gestión.

Métricas de proceso

Se recopilan de todos los proyectos y durante un largo periodo de tiempo

Caracterizados por:

- *Control y ejecución del proyecto.*
- *Medición de tiempos de las fases.*

Para este proyecto se trabajará con las siguientes métricas del proceso:

- *Costo de remoción de defectos*
- *Cantidad de código rehusado*
- *Distribución de esfuerzo por fase*
- *Efectividad para remover defectos entre fases*
- *Soporte de herramientas para procesos propuestos*

Métricas de proyecto

- *Permiten evaluar el estado del proyecto.*
- *Permiten seguir la pista de los riesgos.*

Para este proyecto se trabajará con las siguientes métricas del proyecto:

- *Cantidad de puntos de función liberados por unidad de tiempo*
- *Costo del desarrollo*
- *Costo del soporte*
- *Horas trabajadas*
- *Tiempo (calendario) transcurrido*
- *Distribución del esfuerzo por fase*
- *Cambios sobre requerimientos durante el desarrollo*
- *Cambio sobre requerimientos en operación*
- *Origen de los cambios sobre requerimientos*
- *Cronograma Vs Estimado*
- *Costo sobre valor agregado*
- *Porcentaje de requerimientos implementados por unidad de tiempo*
-

Métricas de producto

- *Se centran en las características del software y no en cómo fue producido.*
- *También son productos los artefactos, documentos, modelos, y componentes que conforman el software.*
- *Se miden cosas como el tamaño, la calidad, la totalidad, la volatilidad, y el esfuerzo*

Para este proyecto se trabajará con las siguientes métricas del producto:

- *Puntos de Caso de Uso*
- *Puntos de función*
- *Complejidad de diseño (acoplamiento)*
- *Complejidad de código*
- *Métodos por clase*
- *Profundidad y ancho de jerarquías*
- *Cantidad de objetos y cantidad de relaciones de colaboración diferentes*
- *Volatilidad de componentes*
- *Complejidad de despliegue*
- *Densidad de defectos*
- *Tipo y origen de defectos*
- *Cantidad de problemas reportados*
- *Tiempo transcurrido entre fallas*
- *Tiempo esperado para la siguiente falla*
- *Tiempo requerido para reparar*
- *SLOC*
- *Facilidad de aprendizaje de uso*
-

Estándar de documentación

Como estándares de documentación se definirán dos documentos:

- *Estándar de documentación técnica*
- *Estándar de documentación de usuario.*

La documentación técnica del producto debe:

- *Ser adecuada para que un grupo independiente del de desarrollo pueda encarar el mantenimiento del producto.*
- *Incluir fuentes, Modelos de Casos de Uso, Objetos de diseño.*

Para la escritura de documentos se han definido plantillas para ser utilizadas en la elaboración de entregables.

En estas plantillas se definen:

- *Encabezado y pie de página.*
- *Fuente y tamaño de fuente para estilo normal.*
- *Fuente y tamaño de fuente para los títulos a utilizar.*
- *Datos mínimos que se deben incluir: fecha, versión y responsables.*

Estándar de verificación y prácticas

[Se utilizan las prácticas definidas en el Plan de Verificación y Validación.

Como estándar se utiliza el documento de: Std 1012-1986 IEEE Standard for Software Verification and Validation Plans.]

Estándares y pautas a seguir

Los autores se comprometen a seguir los siguientes estándares y pautas pactados:

- Se deberá seguir y aprender un estilo de codificación por cada lenguaje que se utilice, asegurando coherencia y legibilidad en el código.
- Se deberá documentar el código de manera clara y estructurada, utilizando herramientas y formatos estándar según el lenguaje de programación empleado.
- Los comentarios de código deben tener un formato asociado, incluyendo descripciones concisas de funciones, clases y métodos, así como etiquetas específicas para remarcar aspectos importantes.
- Realizar revisiones periódicas para verificar el cumplimiento de estándares, identificar errores y mejorar la calidad del producto software.
- Implementar pruebas para validar la funcionalidad del código y garantizar su fiabilidad.
- Utilizar control de versiones para gestionar cambios con el fin de que exista una recuperación ante una falla grave.
- Reutilizar código mediante el uso de patrones de diseño, bibliotecas y buenas prácticas de programación.

- Asegurar el cumplimiento de normas de seguridad y protección de datos en todas las etapas del desarrollo.
- Capacitarse en nuevas herramientas, lenguajes de programación y buenas prácticas de programación.
- Seguir un enfoque de mejora continua.

Revisiones y auditorías

Objetivo

Definir tipos de revisiones, auditorías técnicas y auditorías de gestión que se realizarán y especificar cómo serán llevadas a cabo.

Requerimientos mínimos

[Se especifican las revisiones y auditorías que deben realizarse como mínimo, así como la agenda para la realización de las mismas.]

Revisión de requerimientos

[Esta revisión se realiza para asegurar que se ha cumplido con los requerimientos especificados por el Cliente.]

Revisión de diseño preliminar

[Esta revisión se realiza para asegurar la consistencia y suficiencia técnica del diseño preliminar del software.]

Revisión de diseño crítico

[Esta revisión se realiza para asegurar la consistencia del diseño detallado con la especificación de requerimientos.]

Auditoría funcional

[Esta auditoría se realiza previa a la liberación del software, para verificar que todos los requerimientos especificados en el documento de requerimientos fueron cumplidos.]

Auditoría física

[Esta revisión se realiza para verificar que el software y la documentación son consistentes y están aptos para la liberación.]

Auditorías internas al proceso

[Estas auditorías sirven para verificar la consistencia: del código versus el documento de diseño, especificaciones de interfase, implementaciones de diseño versus requerimientos funcionales, requerimientos funcionales versus descripciones de testeó.]

Revisiones de gestión

[Estas revisiones se realizan periódicamente para asegurar la ejecución de todas las actividades identificadas en este Plan. Deben realizarse por una persona ajena al grupo de trabajo.]

Revisión del Plan de gestión de configuración

[Esta revisión se realiza para asegurar la consistencia y completitud de los métodos especificados en el Plan de gestión de configuración.]

Revisión Post Mortem

[Esta revisión se realiza al concluir el proyecto para especificar las actividades de desarrollo implementadas durante el proyecto y para proveer recomendaciones.]

Agenda

[En esta sección se deberá especificar la agenda para las revisiones y auditorías detalladas anteriormente.]

Otras revisiones

Revisión de documentación de usuario

[Se revisa la completitud, claridad y aplicación de uso.]

Verificación

[Se debe identificar todas las verificaciones que no fueron identificadas en el Plan de V & V para el software y debe especificar los métodos a ser usados.]

Reporte de problemas y acciones correctivas

[Esta sección debe incluir: Descripción de las prácticas y procedimientos que se seguirán para el reporte, seguimiento, y resolución de los problemas surgidos en el desarrollo de software; especificar los responsables comprometidos con la implementación de estas acciones correctivas.]

Herramientas, técnicas y metodologías

[Se deben identificar herramientas de software, técnicas, y metodologías de soporte para las actividades de aseguramiento de calidad.]

Gestión de riesgos

Un riesgo es un evento adverso que puede o no ocurrir el cual tiene asociado un impacto al proyecto. Este impacto a su vez se puede definir como qué tan grave fueron las pérdidas.

La gestión de riesgo puede seguir una estrategia proactiva o reactiva. En este caso se optó por la estrategia proactiva ya que las ventajas son superiores a las estrategias reactivas.

La gestión de riesgos se componen de las siguientes actividades:

- **Identificación:** En esta se obtiene la lista de riesgos preliminares, se busca todos los riesgos posibles que puedan ocurrir. Usualmente se tiene una lista de preguntas para determinar qué riesgos se tiene o por la experiencia del equipo también se conoce algunos riesgos.
- **Análisis:** En esta etapa se obtiene una lista de riesgos prioritarios. Para este caso, se analiza sobre tres ejes los riesgos, el impacto en una escala de despreciable, tolerable, serio y catastrófico. La probabilidad que se mide en porcentaje del 1 al 100 y la magnitud es medida en semanas.
- **Planificación:** En esta etapa se obtienen los planes de prevención, minimización y contingencia para los riesgos prioritarios. En este caso se seleccionan los riesgos con mayor magnitud, probabilidad e impacto y se realizan los correspondientes planes los cuales tienen asociados tareas concretas. Estos planes tienen como objetivo disminuir la probabilidad, el impacto y si ocurre como se debería manejar.
- **Monitoreo:** Esta etapa se va realizando periódicamente, al inicio de cada iteración, y tiene como fin revisar si se introdujo un nuevo riesgo, si se modificó el impacto,

probabilidad o magnitud de los riesgos que fueron encontrados y si se debería volver a planificar los riesgos ya gestionados.

Se anexa el “*plan de Gestión de Riesgos*” que explica en más detalle estas etapas y cómo se realizará durante el proyecto.

Anexos

Formulario de Pedidos y Detección de Cambios

Formulario de Pedidos y Detección de Cambios	
<i>Fecha de Petición:</i>	
<i>Nombre y Versión del Elemento</i>	
<i>Nombre del Solicitante:</i>	
<i>Necesidad de Cambio:</i>	
<i>Descripción del cambio pedido:</i>	
<i>Prioridad:</i>	
<i>Estado:</i>	
<i>Fecha del cambio:</i>	
<i>Identificador de la nueva versión:</i>	
<i>Que fue afectado por este cambio</i>	