Plan de SQA

Sistema VASPA

VASPA Team

Francisco G. Estrada

Revisor: Fabricio González



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

En el Plan de SQA se reflejan las evaluaciones a realizar, los estándares a aplicar, los productos a realizar, los procedimientos a seguir en 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.



Tabla de contenido

[Propósito 5](#_Toc525465951)

[Referencias 5](#_Toc525465952)

[Gestión 6](#_Toc525465953)

[Organización 6](#_Toc525465954)

[Actividades 6](#_Toc525465955)

[Ciclo de vida del software cubierto por el Plan 6](#_Toc525465956)

[Actividades de calidad a realizarse 6](#_Toc525465957)

[Relaciones entre las actividades de SQA y la planificación 8](#_Toc525465958)

[Responsables 8](#_Toc525465959)

[Documentación 9](#_Toc525465960)

[Propósito 9](#_Toc525465961)

[Documentación mínima requerida 9](#_Toc525465962)

[Especificación de requerimientos del software 9](#_Toc525465963)

[Descripción del diseño del software 11](#_Toc525465964)

[Plan de Verificación & Validación 12](#_Toc525465965)

[Documentación de usuario 12](#_Toc525465966)

[Plan de Gestión de configuración 13](#_Toc525465967)

[Propósito 13](#_Toc525465968)

[Resumen 13](#_Toc525465969)

[Organización, Responsabilidades 13](#_Toc525465970)

[Herramientas, Entorno, e Infraestructura 13](#_Toc525465971)

[Forma de trabajo 14](#_Toc525465972)

[Control de Cambios 14](#_Toc525465973)

[Reportes y Auditorias 14](#_Toc525465974)

[Otros documentos 15](#_Toc525465975)

[Estándares, prácticas, convenciones y métricas 15](#_Toc525465976)

[Objetivos 15](#_Toc525465977)

[Métricas de proceso 16](#_Toc525465978)

[Métricas de proyecto 16](#_Toc525465979)

[Métricas de producto 17](#_Toc525465980)

[Estándar de documentación 17](#_Toc525465981)

[Estándar de verificación y prácticas 18](#_Toc525465982)

[Otros Estándares 18](#_Toc525465983)

[Revisiones y auditorías 18](#_Toc525465984)

[Objetivo 18](#_Toc525465985)

[Requerimientos mínimos 18](#_Toc525465986)

[Revisión de requerimientos 19](#_Toc525465987)

[Revisión de diseño preliminar 19](#_Toc525465988)

[Revisión de diseño crítico 19](#_Toc525465989)

[Auditoría funcional 19](#_Toc525465990)

[Auditoría física 19](#_Toc525465991)

[Auditorías internas al proceso 19](#_Toc525465992)

[Revisiones de gestión 19](#_Toc525465993)

[Revisión del Plan de gestión de configuración 20](#_Toc525465994)

[Revisión Post Mortem 20](#_Toc525465995)

[Agenda 20](#_Toc525465996)

[Otras revisiones 20](#_Toc525465997)

[Revisión de documentación de usuario 20](#_Toc525465998)

[Verificación 20](#_Toc525465999)

[Reporte de problemas y acciones correctivas 20](#_Toc525466000)

[Herramientas, técnicas y metodologías 20](#_Toc525466001)

[Gestión de riesgos 21](#_Toc525466002)

[Anexos 22](#_Toc525466003)

[Formulario de Pedidos y Detección de Cambios 22](#_Toc525466004)

Plan de SQA

Propósito

El propósito de este plan es especificar las actividades que se realizarán para asegurar la calidad del software a construir. En él se detallan los productos que se van a revisar y los estándares, normas o métodos a aplicar; los métodos y procedimientos que se utilizarán para revisar que la elaboración de los productos se realice como lo establece el modelo de ciclo de vida del proyecto; y procedimientos para informar a los responsables de los productos los defectos encontrados y realizar un seguimiento de dichos defectos hasta su corrección.

El software que se desarrollará será una aplicación web y una posible aplicación móvil. En la aplicación web los docentes podrán cargar los programas de asignaturas por medio de formularios y la generación del mismo en PDF. El personal de Secretaria Académica podrá tener un registro de los profesores a los cuales se notificó la solicitud del programa, y poder ver el estado del programa durante el proceso de firmas de las autoridades correspondientes. Por último la comunidad universitaria en general podrá tener acceso a la visualización y descarga de los programas de asignaturas.

Con la aplicación se quiere lograr:

* Mejorar la calidad y eficiencia en el control y seguimiento de los programas de asignaturas de la UNPA - UARG.
* Gestionar de manera más eficiente los programas de asignaturas.
* Aminorar la carga de trabajo de los empleados administrativos de Secretaria Académica.

Referencias

[ANSI/IEEE Std 730.1-1989, IEEE Standard for Software Quality Assurance]

Gestión

En esta sección, se detalla el rol que ocupará cada uno de los integrantes del VASPA Team.

Organización

El equipo de trabajo está estructurado de la siguiente forma:

|  |  |  |  |
| --- | --- | --- | --- |
| **Persona**  **Rol** | **Fabricio González** | **Nicolás Sartini** | **Francisco Estrada** |
| **Administrador de la Configuración** | **X** |  |  |
| **Analista** |  | **X** |  |
| **Arquitecto** |  |  | **X** |
| **Diseñador** |  | **X** |  |
| **Documentador** |  | **X** |  |
| **Gerente de Calidad** |  |  | **X** |
| **Ingeniero de Pruebas** | **X** |  |  |
| **Líder de Proyecto** | **X** |  |  |
| **Programador** |  |  | **X** |
| **Rol General** | **X** | **X** | **X** |

Actividades

Ciclo de vida del software cubierto por el Plan

El presente Plan comprende las etapas de Inicio, Elaboración, Construcción y Transición del RUP.

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

En esta actividad se revisan los productos que se definieron como claves para verificar en el Plan de calidad.

Se debe verificar que no queden correcciones sin resolver en los informes de revisión previos, si se encuentra alguna no resuelta, debe ser incluida en esta revisión. Se revisan los productos contra los estándares, utilizando la checklist definida para el producto.

Se debe identificar, documentar y seguir la pista a las desviaciones encontradas y verificar que se hayan realizado las correcciones.

Como salida se obtiene el Informe de revisión de SQA, este informe debe ser distribuido a los responsables del producto y se debe asegurar de que son conscientes de desviaciones o discrepancias encontradas.

Revisar el ajuste al proceso

[En esta actividad se revisan los productos que de 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 conscientes 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 posibles defectos o desviaciones en los productos que se van generando a lo largo del desarrollo. Por esta característica se adopta esta práctica para productos que son de especial importancia.

En la reunión participan el responsable de SQA e integrantes del equipo de desarrollo.

Se debe convocar a la reunión formalmente a los involucrados, informar del material que ellos deben preparar por adelantado, llevar una lista de preguntas y dudas que surgen del estudio del producto a ser revisado.

La duración de la reunión no debe ser mayor a dos horas.

Como salida se obtiene el Informe de RTF.

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.

Documentación

Propósito

Identificación de la documentación relativa a desarrollo, Verificación & Validación, uso y mantenimiento del software.

Establecer como los documentos van a ser revisados para chequear consistencia: se confirman criterio e identificación de las revisiones.

Documentación mínima requerida

Esta busca asegurar que la implementación logrará satisfacer los requerimientos.

La documentación mínima es la requerida para asegurar que la implementación lograra satisfacer los requerimientos.

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’.

Funcionalidad

* Adecuación a las necesidades
* Precisión de los resultados
* Interoperabilidad
* Seguridad de los datos

Confiabilidad

* Madurez
* Tolerancia a faltas
* Recuperabilidad

Usabilidad

* Comprensible
* Aprendible
* Operable
* Atractivo

Eficiencia

* Comportamiento respecto al tiempo (Ver si aplica)
* Utilización de recursos

Mantenibilidad

* Analizable
* Modificable
* Estable, no se producen efectos inesperados luego de modificaciones
* Verificable

Portabilidad

* Adaptable (Ver si aplica)
* Instalable
* Co-existencia
* Reemplazante (Ver si aplica)

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.

El diseño debe:

* Corresponder a los requerimientos a incorporar:
  + 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

Plan de Verificación & Validación

El Plan de V & V deberá identificar y describir los métodos a ser utilizados en:

* La verificación de que:
  + Los requerimientos descritos en el documento de requerimientos han sido aprobados por una autoridad apropiada.
  + Los requerimientos descritos en el documento de requerimientos son implementados en el diseño expresado en el documento de diseño.
  + El diseño expresado en el documento de diseño esta implementado en código.
* Validar que el código, cuando es ejecutado, se adecua a los requerimientos expresados en el documento de requerimientos.

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 una documentación para el usuario de acuerdo a los requerimientos específicos del proyecto.

Plan de Gestión de configuración

La Gestión de Configuraciones permite controlar el sistema como producto global a lo largo de su creación, obtener informes sobre el estado de desarrollo en que se encuentra y reducir el número de errores durante el mismo, lo que se traduce en un aumento de calidad del proceso de desarrollo y de mejora de la productividad en la organización.

La gestión de configuración facilita además el mantenimiento del sistema, aportando información precisa para valorar el impacto de los cambios solicitados y reduciendo el tiempo de implementación de un cambio, tanto evolutivo como correctivo.

Propósito

Controlar la entrega y el cambio de los elementos a través del ciclo de vida del sistema. Almacenar el estado de los elementos y de las peticiones de cambio.

Resumen

La Gestión de Configuración, en resumen, 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

El administrador de la configuración será Fabricio González. Las solicitudes de cambio se harán por correo electrónico.

Herramientas, Entorno, e Infraestructura

[Se utilizara la herramienta de Gestión de Configuraciones (CGS) Subversion y TortoiseSVN. Este maneja ficheros y directorios a lo largo del ciclo de vida del proyecto. Los ficheros se almacenan en un repositorio central, recordando todos los cambios que se hayan realizado, permitiendo a los integrantes del grupo poder recuperar versiones anteriormente guardadas, examinar la historia de cuando y como fueron modificados los datos, quien hizo los mismos y así poder coordinar el trabajo.

Siendo la misma especialmente útil para los documentos revisados frecuentemente, como el código fuente, la documentación, etc., como así también llevar un balance histórico de las diferentes versiones del sistema.]

Forma de trabajo

[Durante el proceso de gestión de configuración se utilizará la herramienta Subversion para el control de versiones del producto. Cuando algún miembro haga una modificación en el proyecto, deberá acceder al servidor donde está alojada esta aplicación para almacenar la parte modificada en él, teniendo el resto del equipo de desarrollo la última versión actualizada en dicho servidor. Esta gestión de acceso al servidor para la actualización se hará mediante la herramienta Tortoise para los documentos y el plugin Subclise para el código fuente.]

Control de Cambios

No se ha establecido documentación formal para la gestión de cada uno de los cambios.

Las herramientas utilizadas para el control de versiones dan muchísima información sobre cada cambio realizado. Además, permiten hacer búsquedas y filtrados dinámicos que no se podrían realizar teniendo esta información en documentos.

Estándares, prácticas, convenciones y métricas

[Esta sección deberá cumplir con las siguientes funciones:

Identificar los estándares, prácticas, convenciones y métricas que serán aplicadas.

Indicar como será monitoreado y asegurado el cumplimiento con estos ítems

El IEEE “Standard Glosary of Software Engering Terms” define como métrica: “una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado”.]

[Las métricas son una herramienta poderosa y fundamental para el trabajo en SQA. Su aporte fundamental son las medidas preventivas que pueden surgir a raíz de su estudio. Sin duda aportan conclusiones que muchas veces no se aprecian a simple vista y que ayudan a mejorar la eficiencia del grupo de trabajo y la calidad de los productos. Aportan un caudal de información para hacer controles estadísticos de la calidad. Además cabe resaltar que nunca debe dejarse de buscar nuevas métricas de acuerdo a las nuevas variaciones y tendencias de las estadísticas.]

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
* Volativilidad 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.

Revisiones y auditorías

Objetivo

Definición de las revisiones y auditorías técnicas y de gestión que se realizarán.

Especificación de cómo serán llevadas a cabo dichas revisiones y auditorías.

Requerimientos mínimos

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

Cómo mínimo deberán revisarse todas las entregas semanales, basado en los estándares definidos anteriormente. Estas revisiones serán realizadas por el Responsable de SQA y/o el Asistente de SQA.

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 testeo.

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.]

A continuación se muestra una agenda, tentativa, de las revisiones a realizar. Algunas fechas pueden ser modificadas conforme avanza el proyecto.

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.]

La verificación se hará conforme a lo expresado en el documento: Plan de Verificación y Validación.

Reporte de problemas y acciones correctivas

La manera en que el equipo de desarrollo trabajará es sencilla.

1. Desarrollador programa funcionalidad
2. Tester prueba dicha funcionalidad y elabora un documento formal sobre esta prueba.
3. Tester elabora un documento con el resumen de los errores encontrados.
4. El desarrollador corrige los errores detectados.
5. El tester realiza pruebas de regresión.
6. Se corrigen por última vez lo errores en caso de que se haya detectado alguno en las pruebas de regresión

Herramientas, técnicas y metodologías

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

Se utilizará la técnica de Check List definida en la página del MUM del curso.

Gestión de riesgos

La Gestión de Riesgos tendrá su propia documentación, ubicada en el directorio “Gestión de Riesgos” en el repositorio.

Se contará con un Plan de Gestión de riesgos, anexos con riesgos detectados en las distintas iteraciones del desarrollo y un documento de seguimiento de riesgos.