**Plan de SQA**

****

**José Ángel Ruiz Escalante**

**Jesús Andrei Torres Landero**

**David Abraham Paredes Coob**

**Julián Alejandro Pérez Koo**

**SQA**

**M.I.T. Edwin Jesús León Bojórquez**

**Mayo 2022**

**ÍNDICE**

[Control de la documentación](#_heading=h.30j0zll) **5**

[Control de configuración](#_heading=h.1fob9te) 5

[Histórico de versiones](#_heading=h.3znysh7) 5

[Histórico de cambios](#_heading=h.tyjcwt) 5

[**Propósito**](#_heading=h.qgntdiy5xqcy) **7**

[Alcance](#_heading=h.4d34og8) 7

[Descripción del proyecto](#_heading=h.2s8eyo1) 7

[Elementos cubiertos por el Plan de Aseguramiento de la Calidad de Software](#_heading=h.11n5wez3zns2) 8

[**Documentos referenciados**](#_heading=h.3rdcrjn) **8**

[**Gestión**](#_heading=h.eiqll53m14wv) **10**

[Organización](#_heading=h.lnxbz9) 10

[Tareas de la Unidad de SQA](#_heading=h.35nkun2) 16

[Responsabilidades de la Unidad de SQA](#_heading=h.1ksv4uv) 20

[Recursos estimados para el aseguramiento de la calidad](#_heading=h.44sinio) 21

[**Documentación**](#_heading=h.z337ya) **21**

[Propósito](#_heading=h.3j2qqm3) 21

[Documentación mínima requerida](#_heading=h.1y810tw) 21

[Especificación de requerimientos del software](#_heading=h.4i7ojhp) 21

[Plan de Pruebas de Sistema](#_heading=h.2xcytpi) 21

[Manual de Usuario](#_heading=h.3t5veu52dnl8) 22

[Manual de Mantenimiento](#_heading=h.y6ambcre0v61) 22

[Plan de Pruebas de Integración](#_heading=h.mak1tk3fya2q) 22

[Descripción del diseño del software](#_heading=h.15up9zue87kn) 23

[Plan de Verificación y Validación](#_heading=h.1ci93xb) 23

[Plan de Gestión de configuración](#_heading=h.qsh70q) 23

[Plan de desarrollo de software](#_heading=h.1pxezwc) 23

[Manual de Operación](#_heading=h.js555nbbava) 24

[Análisis y Diseño](#_heading=h.a0dunon41qik) 24

[Registro de rastreo](#_heading=h.z2e8k91h3nco) 24

[Reportes de Verificación y Validación](#_heading=h.3whwml4) 24

[**Estándares, prácticas, convenciones y métricas**](#_heading=h.xn39x8ppfr81) **25**

[Propósito](#_heading=h.i90pe175ur60) 25

[Fase de requerimientos](#_heading=h.23ckvvd) 25

[Fase de Análisis y Diseño](#_heading=h.axfipedeg9o9) 25

[Fase de Construcción](#_heading=h.5kxcuq56q19l) 26

[Fase de Integración y Pruebas](#_heading=h.owpnfu9fjs9p) 27

[Métricas seleccionadas de productos y procesos de aseguramiento de la calidad del software](#_heading=h.2grqrue) 27

[**Revisiones y Auditorías de software**](#_heading=h.yme6is76t8b5) **28**

[Objetivo](#_heading=h.3fwokq0) 28

[Requerimientos mínimos](#_heading=h.1v1yuxt) 28

[Revisión de requerimientos](#_heading=h.4f1mdlm) 28

[Revisión del Plan de Pruebas de Sistema](#_heading=h.hen7xhrppgit) 28

[Revisión de Manual de Usuario](#_heading=h.dhy339l7jb4a) 29

[Revisión Análisis y Diseño](#_heading=h.7a3o2oftb24j) 29

[Revisión de Plan de Pruebas de Integración](#_heading=h.n8bf4hzfevs6) 29

[Revisión de Registro de Rastreo](#_heading=h.j5opdiieredm) 29

[Revisión de Manual de Operación](#_heading=h.h0e1aqiic1vq) 30

[Revisión de Manual de Usuario](#_heading=h.vzxgpz7s80t3) 30

[Revisión de Manual de Mantenimiento](#_heading=h.dsn03zt43ong) 30

[Revisión del Plan de gestión de configuración](#_heading=h.46r0co2) 30

[Revisiones de gestión](#_heading=h.cvdzxph7ccwx) 31

[Revisión de diseño preliminar](#_heading=h.2u6wntf) 31

[Revisión del Plan de Verificación y Validación](#_heading=h.3tbugp1) 31

[Auditoría funcional](#_heading=h.28h4qwu) 31

[Auditoría física](#_heading=h.nmf14n) 31

[Auditorías internas al proceso](#_heading=h.37m2jsg) 32

[Revisión Post-implementación](#_heading=h.2lwamvv) 32

[**Pruebas**](#_heading=h.6974xteyzw5i) **33**

[**Informes de problemas y acciones correctivas**](#_heading=h.tfn62r2c82lp) **34**

[Propósito](#_heading=h.cribysz77kpu) 34

[Objetivo](#_heading=h.yfm6u848sne9) 34

[Identificación de no conformidades](#_heading=h.tcol9dyuxfso) 34

[Presentación del reporte al líder de Unidad de SQA](#_heading=h.um24sjltt936) 34

[Implementación de Acciones correctivas](#_heading=h.4k668n3) 35

[Las acciones correctivas que se generaron en la reunión de revisiones o auditorías, se debe entregar al equipo de trabajo que fue o fueron afectados por el problema de esta manera podrán implementar las acciones correctivas.](#_heading=h.gm8by3chbymk) 35

[**Herramientas, técnicas y metodologías**](#_heading=h.6ty23lnvqz2m) **36**

[Herramientas](#_heading=h.jd5up1qmwuof) 36

[Técnicas](#_heading=h.ov10tdg2x438) 37

[Metodologías](#_heading=h.w4ja58pk8kk0) 37

[**Control de Código**](#_heading=h.y65g516gktix) **38**

[Propósito](#_heading=h.1egqt2p) 38

[Documentación afectada por el control de código](#_heading=h.1rvwp1q) 38

[Elemento de configuración](#_heading=h.sh138na49m4h) 38

[Mantenimiento y control de la gestión de la configuración](#_heading=h.2r0uhxc) 38

[Proceso de control de cambios](#_heading=h.moafaowg2ec8) 38

[Recuperación de los elementos de configuración](#_heading=h.jly5g9cya595) 39

[Identificaciones de versiones](#_heading=h.awhibz86ujzw) 40

[**Control de medios**](#_heading=h.i05kln26pgs0) **41**

[Propósito](#_heading=h.1664s55) 41

[Medio de almacenamiento](#_heading=h.3q5sasy) 41

[Procesos de copias de seguridad](#_heading=h.25b2l0r) 41

[Acceso no autorizado](#_heading=h.kgcv8k) 41

[**Recopilación, mantenimiento y retención de registros**](#_heading=h.2iq8gzs) **42**

[Procedimiento de control de la documentación](#_heading=h.xvir7l) 42

[Lista de documentos controlados](#_heading=h.1x0gk37) 42

[Preparación de documentos controlados](#_heading=h.4h042r0) 42

[**Entrenamiento**](#_heading=h.9nwi6ykkw95o) **44**

[**Glosario**](#_heading=h.d9tdljojo3hm) **46**

[**Anexo**](#_heading=h.rfyo0s89jmtw) **47**

[A1. Reporte de no conformidad](#_heading=h.6r331a4q56cr) 47

[A2. Solicitud de Cambio de Software](#_heading=h.sxeqcpce1xxw) 48

[A3. SCI Version Document](#_heading=h.c0wf7iddv3ov) 49

[A4. Software Configuration Release Documentation](#_heading=h.iv24bai246nm) 50

# Control de la documentación

## Control de configuración

|  |  |
| --- | --- |
| **Título** | Plan de SQA |
| **Referencia** | <https://github.com/JoseAngelRE/PlanDeSQA> |
| **Autor** | José Ángel Ruiz Escalante |
| **Fecha** | 09/05/2022 |

## Histórico de versiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versión** | **Fecha** | **Estado** | **Responsable** | **Nombre del archivo** |
| 1.0.0 | 09/05/2022 | A | José Ángel Ruiz Escalante | Plan\_de\_SQA |
| 2.0.0 | 18/05/2022 | A | José Ángel Ruiz Escalante | Plan\_de\_SQA |
| 3.0.0 | 29/05/2022 | A | José Ángel Ruiz Escalante | Plan\_de\_SQA |

Estado (B)orrador, (R)evisión, (A)probado

## Histórico de cambios

|  |  |  |
| --- | --- | --- |
| **Versión** | **Fecha** | **Cambios** |
| 1.0.0 | 09/05/2022 | Se realizaron los apartados del 1 al 7: propósito, documentos referenciados, administración, documentación, estándares, prácticas, convenciones y métricas, revisiones y auditorías, y pruebas. |
| 2.0.0 | 18/05/2022 | Se realizaron los apartados Informes de problemas y acciones correctivas, Herramientas, técnicas y metodologías, Control de código, Control de medios, Control de proveedores, Recopilación, mantenimiento y retención de registros, Entrenamiento y Gestión de riesgos. |
| 3.0.0 | 29/05/2022 | Se realizaron las correcciones en los apartados Propósito, Documentos referenciados, Gestión, Documentación, Estándares Prácticas Convenciones y métricas, Revisiones y Auditorías, Pruebas, Informes de problemas y acciones correctivas, Herramientas Técnicas y Metodologías, Control de código, Entrenamiento y se agregaron las plantillas en el Anexo. |

# 

# Propósito

El propósito de este plan es definir la organización para el aseguramiento de la calidad del proyecto para la Prevención del Suicidio en Yucatán (P.S.Y.), las tareas y responsabilidades de SQA; proporcionar documentos de referencia y directrices para realizar las actividades de SQA; proporcionar los estándares, prácticas y convenciones utilizados para llevar a cabo las actividades de SQA; además, de proporcionar las herramientas, técnicas y metodologías para respaldar las actividades de SQA y los informes de SQA.

## Alcance

El plan de SQA está dirigido al líder del proyecto, el equipo de desarrollo y al grupo de SQA, responsable de la elaboración, actualización y monitoreo del plan. La meta del plan de aseguramiento de calidad es verificar que todo el software y la documentación que será entregada cumpla con todos los requerimientos técnicos.

Las actividades de SQA deben ser ejecutadas durante el ciclo de vida del software definido para la aplicación. El ciclo de vida comprende:

* Planificación
* Especificación de requerimientos
* Análisis
* Diseño
* Implementación
* Mantenimiento

## Descripción del proyecto

Se debe construir un sistema que promueva información sobre las enfermedades que pueden ser causas del suicidio, llevando a cabo una de las estrategias para la prevención del suicidio según el estudio realizado por la CODHEY (Comisión de los Derechos Humanos del Estado de Yucatán).

Según este estudio, aunque la depresión tiene posibilidades de tratamiento, no siempre es diagnosticada, lo que obstaculiza los esfuerzos para manejar las consecuencias económicas, sociales y de salud que tiene la depresión.

Al construir esta aplicación web se busca que las personas tengan conocimiento en medida de lo posible de la mayoría de las enfermedades que pueden ocasionar la depresión, así como los comportamientos que puede tener una persona que tiene depresión, con el fin de poder detectar cuando algún familiar cercano o alguna persona de su entorno está pasando por un mal momento y poder ayudar a tiempo.

## Elementos cubiertos por el Plan de Aseguramiento de la Calidad de Software

* Documento de Especificación de Requerimientos
* Plan de Pruebas del Sistema
* Manual de Usuario
* Manual de Mantenimiento
* Manual de Operación
* Análisis y Diseño
* Registro de rastreo
* Código fuente
* Plan de Pruebas de Integración

# 

# Documentos referenciados

ANSI/IEEE Std 729-1983, IEEE Standard Glossary of Software Engineering Terminology.

ANSI/IEEE Std 829-1983, IEEE Standard for Software Test Documentation.

IEEE Std 730-2002, IEEE Standard for Software Quality Assurance Plans.

IEEE Std 830-1998, IEEE Standard Recommended Practice for Software Requirements Specifications.

IEEE Std 1012-2016, IEEE Standard for System, Software, and Hardware Verification and Validation.

IEEE Std 1016-2009, IEEE Standard for Information Technology Systems Design, Software Design Descriptions.

IEEE Std 828-2012, IEEE Standard for Configuration Management in Systems and Software Engineering.

IEEE Std 1063-2001, IEEE Standard for Software User Documentation.

IEEE Std 1008-1987, IEEE Standard for Software Unit Testing.

IEEE Std 29119-2-2013, IEEE Standard for Software and systems engineering - Software testing.

IEEE Std 1028-2002, IEEE Standard for Software Reviews.

IEEE Std 1061-1992, IEEE Standard for a Software Quality Metrics Methodology.

IEEE Std 15939-2007, IEEE Standard for Systems and software engineering - Measurement Process.

IEEE Std 1045-1992, IEEE Standard for Software Productivity Metrics.

IEEE Std 26512-2018, IEEE Standard for Systems and software engineering requirements for acquirers and supplier of information for users.

IEEE Std 1540-2001, IEEE Standard for Software Lifecycle Processes - Risk Management.

IEEE Std 610.12 1990, IEEE Standard for Glossary of Software Engineering Terminology.

IEEE Std 9000-3-2008, IEEE Standard Adoption of ISO/IEC 90003:2014, Software Engineering- Guidelines for the Application of ISO 90001:2008 to Computer Software

ISO Std 9001-2015, ISO Quality Management Systems.

IEEE Std 1028-2002, IEEE Standard for Software Reviews

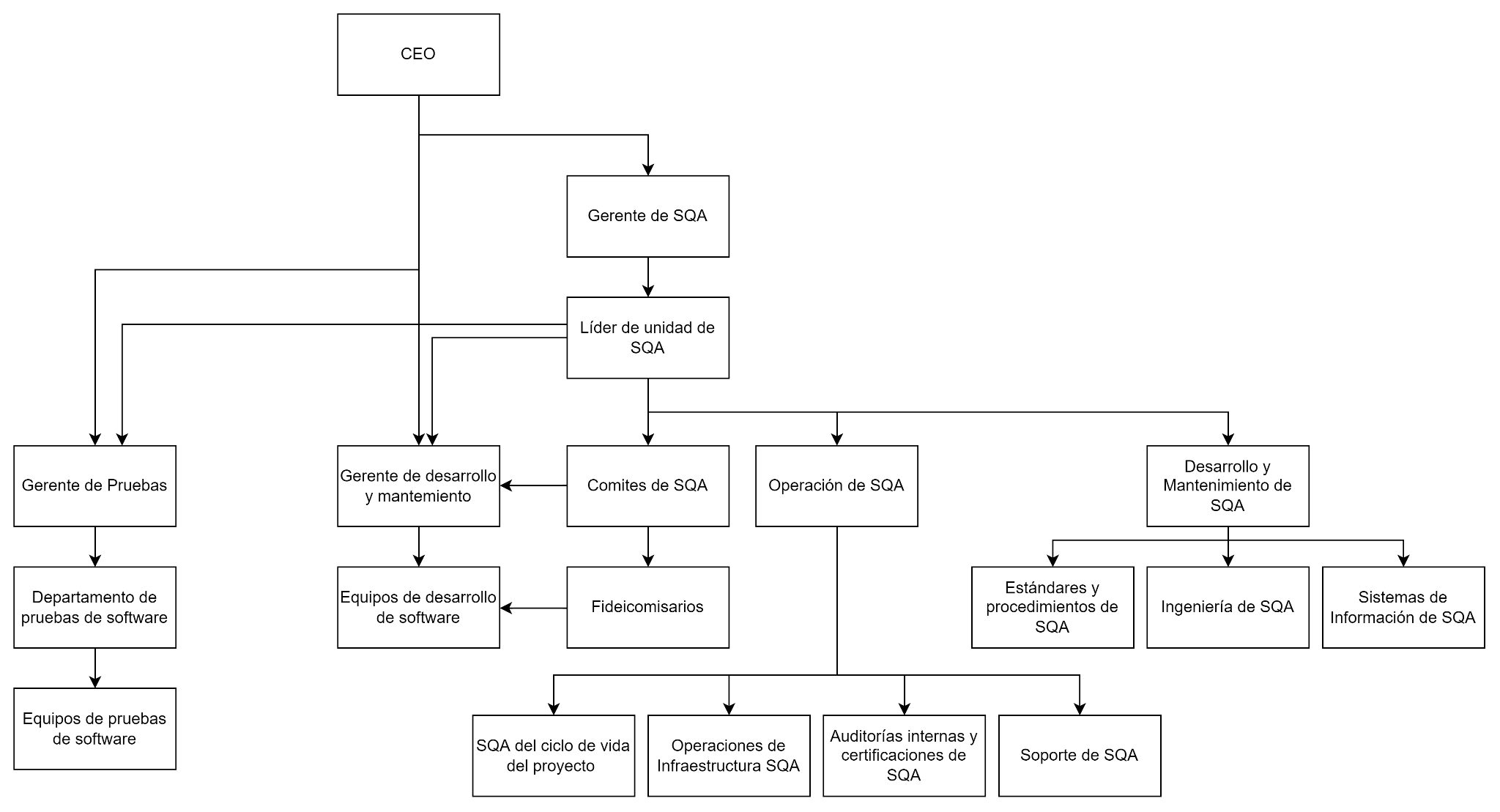
IEEE Std 12207-2008, IEEE Standard for Software and System Test Documentation

Bourque, P. & Fairley, R. E. (eds.) (2014). *SWEBOK: Guide to the Software Engineering Body of Knowledge*. Los Alamitos, CA: IEEE Computer Society. ISBN: 978-0-7695-5166-1

# 

# Gestión

## Organización

**

Organigrama para el Aseguramiento de la Calidad del Software.

A continuación, se describirán los elementos externos a SQA que interactúan directamente con elementos de la unidad SQA.

* **Gerente de SQA:** Es la persona responsable de planear y dirigir el trabajo relacionado con el área específica de SQA, teniendo a su cargo la unidad de SQA.
* **Gerente de desarrollo y mantenimiento:** Es la persona responsable de planear y dirigir el trabajo relacionado con el área específica del desarrollo y mantenimiento del software.
* **Equipos de desarrollo de software:** Es el equipo encargado del proceso de creación, diseño, despliegue y compatibilidad de software.
* **Gerente de pruebas:** Es la persona responsable de planear y dirigir el trabajo relacionado con el área específica de pruebas del software.

El líder de la unidad de SQA delega responsabilidades a las subunidades de SQA encargadas de la operación, el mantenimiento y el desarrollo de SQA.

Las subunidades de la operación, el mantenimiento, el desarrollo de SQA, así como también los comités y los fideicomisarios reportan al líder de la unidad de SQA ya que estos dependen de la gestión del líder el cual a su vez reporta al gerente encargado de SQA sobre la realización de los diferentes elementos que incorporan SQA.

Para la liberación del producto la autoridad responsable es el gerente de SQA ya que él tiene la decisión sobre si un producto cumple con la calidad esperada para un producto de software.

Los elementos de la organización encargados de la aprobación del Plan de SQA son el líder de la unidad de SQA y el gerente de SQA.

En dado caso que existan conflictos dependiendo del origen, en la línea del escalado del conflicto se debe reportar al líder de la unidad de SQA por medio de un Reporte de no conformidad para que el líder de la unidad de SQA tome las responsabilidades necesarias e informe sobre una resolución a los interesados en el conflicto.

**Roles de la Unidad de SQA**

|  |  |  |
| --- | --- | --- |
| Rol | Abreviatura | Descripción |
| Gerente de SQA | GER | Encargado de SQA relacionados con preparaciones del programa y presupuesto anual de actividades de SQA, así como de su aplicación. |
| Líder de unidad de SQA | LU | Responsable de las tareas realizadas por la unidad de SQA y las subunidades |
| Operación de SQA | OPE | Responsable de tareas relacionadas con el ciclo de vida, operaciones de infraestructura, auditorías y soporte de SQA |
| Desarrollo y mantenimiento de SQA | DES | Responsable de los estándares, procedimientos, ingeniería y sistemas de información de SQA. |
| Comités de SQA. | COM | Trata temas como el control de la configuración, acciones correctivas y procedimientos. |
| Fideicomisarios. | FID | Proporcionan el apoyo interno necesario para implementar con éxito los componentes de SQA. |

**Roles de la Subunidad de Operaciones de SQA**

|  |  |  |
| --- | --- | --- |
| Rol | Abreviatura | Descripción |
| SQA del ciclo de vida del proyecto | CV | Encargados de tareas gerenciales de seguimiento y aprobación del control del ciclo de vida del proyecto y tareas de participación activa. |
| Operaciones de Infraestructura SQA | OPI | Encargados de procedimientos e instrucciones, dispositivos de soporte, entrenamiento, acciones preventivas y correctivas, gestión de la configuración y control de la documentación. |
| Auditorías internas y certificaciones de SQA | AIC | Encargados de auditorías internas y externas para la evaluación de planes de SQA y certificaciones. |
| Soporte de SQA | SOP | Apoyan a los gerentes de proyecto, líderes de equipo y fideicomisarios de SQA en la implementación de procedimientos SQA. |

**Roles de la Subunidad de Desarrollo y mantenimiento de SQA**

|  |  |  |
| --- | --- | --- |
| Rol | Abreviatura | Descripción |
| Estándares y procedimientos de SQA | EST | Está involucrada en decidir qué estándares de SQA se adoptarán, así como en el desarrollo y mantenimiento de los procedimientos de la organización. |
| Ingeniería de SQA | ING | Se encarga del seguimiento de los avances profesionales, la solución de las dificultades operativas y un análisis experto de las fallas. |
| Sistemas de Información de SQA | SI | Están destinados a facilitar y mejorar el funcionamiento de los sistemas SQA. |

**Actividades del desarrollo del software.**

|  |  |  |
| --- | --- | --- |
| Actividad | Rol | Responsabilidad |
| **Fase de requisitos** | | |
| La capacitación, certificación e identificación de competencias del personal. | LU  AIC | * Identificar competencias del personal. * Realizar las auditorías y certificaciones pertinentes. * Asegurar que se cumple con lo establecido según el estándar IEEE 9000-3 - 2008. |
|
| Gestionar el control de la documentación de la Especificación de Requerimientos. | OPI  COM | * Asegurar la información. * Determinar el nivel de control del documento. * Asegurar el orden y formato de los documentos de control. * Asegurar que se cumple con lo establecido según el estándar ISO 9001 - 2015. |
| Revisión del documento Especificación de Requerimientos. | OPI | * Asegurar que se cumple con lo establecido. * Asegurar que los requisitos se revisen para determinar si son factibles de implementar, están claramente establecidos y son consistentes. * Asegurar que se cumple con lo establecido según el estándar IEEE 1028 - 2002. |
| Analizar el reporte de la revisión para realizar acciones correctivas en caso de no conformidades en el documento de Especificación de Requerimientos. | OPI  COM | * Identificar no conformidades. * Asegurar que las no conformidades se documenten. * Asegurar que los informes de problemas y acciones correctivas sean correctamente evaluados y validados. * Asegurar que se cumple con lo establecido según el estándar ISO 9001 - 2015. |
| Asegurar el documento de Especificación de Requerimientos. | OPI  CV | * Asegurar que los participantes correctos participen en el proceso de definición y asignación de requisitos para identificar todas las necesidades de los usuarios. * Asegurar que los requerimientos son factibles de implementar, están claramente establecidos y son consistentes. * Asegurar que los requisitos asignados identificados como posibles problemas se revisen con el grupo responsable de analizar los requisitos, documentos del sistema y ​​que se realicen los cambios necesarios. * Asegurar que se cumple con lo establecido según el estándar ISO 12207 - 2008. |
| Verificar la Especificación de Requerimientos. | OPI  LU | * Debe tener conocimiento del documento a verificar. * Asegurar la consistencia del documento con el sistema de software. |
| Validar la Especificación de Requerimientos. | LU | * Debe identificar que se requieren funciones adicionales o diferentes a las que pidieron. * Asegurar que los requerimientos son factibles de implementar, están claramente establecidos y consistentes. |
| Gestionar el control del Plan de Pruebas. | OPI  COM | * Asegurar la información. * Determinar el nivel de control del documento. * Mantener orden y formato de los documentos de control. * Asegurar que se cumple con lo establecido según el estándar ISO 9001 - 2015. |
| Revisión del Plan de pruebas utilizando una revisión técnica formal. | OPI | * Asegurar que se cumple con lo establecido. * Asegurar que se cumple con lo establecido según el estándar IEEE 1028- 2002. |
| Analizar el reporte de la revisión para realizar acciones correctivas en caso de no conformidades en el Plan de Pruebas. | OPI  COM | * Identificar no conformidades. * Asegurar que las no conformidades se documenten. * Asegurar que los informes de problemas y acciones correctivas sean correctamente evaluados y validados. * Asegurar que se cumple con lo establecido según el estándar ISO 9001 - 2015. |
| Asegurar el Plan de Pruebas. | OPI  CV | * Asegurar que las pruebas son factibles de implementar, están claramente establecidos * Asegurar que se cumple con lo establecido según el estándar ISO 12207 - 2008. |
| Verificar el Plan de Pruebas de Sistema. | OPI  LU | * Debe tener conocimiento del documento a verificar. * Asegurar la consistencia del documento con la Especificación de Requerimientos. |
| Gestionar el control del Manual de usuario. | OPI  COM | * Asegurar la información. * Determinar el nivel de control del documento. * Asegurar el orden y formato de los documentos de control. * Asegurar que se cumple con lo establecido según el estándar ISO 9001 - 2015. |
| Revisión del Manual de Usuario. | OPI | * Asegurar que se cumple con lo establecido según el estándar IEEE 1028 - 2002. |
| Analizar el reporte de la revisión para realizar acciones correctivas en caso de no conformidades en el Manual de Usuario. | OPI  COM | * Identificar no conformidades. * Asegurar que las no conformidades se documenten. * Asegurar que los informes de problemas y acciones correctivas sean correctamente evaluados y validados. * Asegurar que se cumple con lo establecido según el estándar ISO 9001 - 2015. |
| Asegurar el Manual de Usuario | OPI  CV | * Asegurar que se cumple con lo establecido de acuerdo a ISO 12207-2008. |
| **Fase de análisis y diseño** | | |
| Verificar el Análisis y Diseño y el Registro de Rastreo. | LU  OPI | * Asegurar el cumplimiento del diseño con los requisitos. * Asegurar que se cumple con las necesidades y expectativas acordadas con el cliente. * Debe tener conocimiento del documento a verificar. * Asegurar la consistencia del documento con el sistema de software. |
| Validar el Análisis y Diseño. | LU | * Debe identificar que se requieren funciones adicionales o diferentes a las que pidieron. |
| Verificar el Plan de Pruebas de Integración. | OPI | * Debe tener conocimiento del documento a verificar. * Asegurar la consistencia del documento con Análisis y Diseño. |
| Incorporar Análisis y Diseño, Registro de Rastreo y Plan de Pruebas de Integración como versión base a la Configuración de Software. | OPI  COM | * Asegurar y establecer la versión base del Análisis y Diseño, Registro de Rastreo y Plan de Pruebas en la Configuración del Software. |
| **Fase de construcción.** | | |
| Definir pruebas unitarias para verificar que el funcionamiento de cada componente esté acorde con la parte detallada del Análisis y Diseño. | OPI | * Definir de acuerdo al estándar de la IEEE 829-2008. |
| Verificar el Registro de Rastreo. | OPI | * Asegurar la consistencia del documento con la Especificación de Requerimientos. |
| Incorporar componentes y Registro de Rastreo cómo versión base a la Configuración de Software. | OPI  COM | * Asegurar y establecer la versión base de los componentes y del Registro de Rastreo en la Configuración del Software. |
| **Fase de Integración y pruebas** | | |
| Verificar el Manual de Operación. | OPI  LU | * Debe tener conocimiento del documento a verificar. * Asegurar la consistencia del documento con el software. |
| Verificar el Manual de Usuario. | OPI  LU | * Debe tener conocimiento del documento a verificar. * Asegurar la consistencia del documento con Especificación de Requerimientos. |
| Incorporar Software, Reporte de Pruebas de Integración, Registro de Rastreo, Manual de Operación y Manual de Usuario como versiones base a la Configuración de Software. | OPI  COM | * Asegurar y establecer la versión base del Software, Registro de Rastreo, Manual de Operación y Manual de Usuario en la Configuración del Software. |

## Tareas de la Unidad de SQA

La programación de las tareas de SQA está impulsada por el desarrollo de software. Por lo tanto, una tarea de SQA se realiza en relación con las actividades de desarrollo de software que se están llevando a cabo. Se pueden realizar una o más tareas de SQA simultáneamente hasta que se complete una tarea. Las siguientes tareas, requieren coordinación y cooperación con el equipo del proyecto, serán realizadas por el equipo de SQA.

Tareas de planificación.

* Preparación de la propuesta de programa y presupuesto anual de actividades para la unidad.
* Planificación y actualización del sistema de gestión de la calidad del software de la organización.
* Preparación de los programas de actividades anuales recomendados de SQA para el departamento de desarrollo y mantenimiento de software; Se pueden solicitar asistencia de las subunidades SQA al realizar esta tarea.
* Preparación de planes de desarrollo de sistemas SQA recomendados para los departamentos de desarrollo y mantenimiento de software; Se pueden solicitar asistencia de subunidades SQA al realizar esta tarea.

Tareas de gestión

* Gestión de las actividades del equipo SQA.
* Monitoreo de la implementación del programa de actividades de SQA.
* Nominación de miembros del equipo, miembros del comité de SQA y fideicomisarios de SQA.
* Preparación de informes especiales y periódicos, por ejemplo, el estado de los problemas de calidad del software dentro de la organización e informes mensuales de rendimiento.

Tareas de profesionales SQA.

* Participación en comités conjuntos de proyectos.
* Participación en revisiones formales de diseño.
* Revisión y aprobación de las desviaciones de las especificaciones (cuando lo requieran los procedimientos).
* Consulta con gerentes de proyecto y líderes de equipo.
* Participación en comités y foros de SQA.

Tareas de la subunidad SQA relacionadas con el ciclo de vida del proyecto, pueden clasificarse en dos grupos:

* Tareas gerenciales de seguimiento y aprobación “puras” (tareas de control del ciclo de vida del proyecto).
* Participación práctica o activa en las actividades de SQA del equipo del proyecto, donde se requieren aportes profesionales (tareas de participación).

Tareas de control del ciclo de vida del proyecto.

* Seguimiento del cumplimiento de SQA por parte de los equipos de desarrollo y mantenimiento procedimientos e instrucciones de trabajo.
* Aprobación o recomendación de productos de software (informes de diseño y código) de acuerdo con los procedimientos pertinentes.
* Supervisión de la prestación de servicios de mantenimiento de software a los clientes externos.
* Seguimiento de la satisfacción del cliente (mediante encuestas, etc.) y mantenimiento del contacto con los representantes de control de calidad de los clientes.

Tareas de participación.

* Revisiones de contratos.
* Elaboración y actualización de planes de calidad y desarrollo de proyectos.
* Revisiones formales de diseño.
* Revisiones formales de diseño de los subcontratistas.
* Pruebas de software, incluidas las pruebas de aceptación del cliente.
* Pruebas de aceptación de software de productos de software de subcontratistas.
* Instalación de nuevos productos de software.

Tareas de operaciones de infraestructura de la subunidad SQA.

* Procedimientos e instrucciones de trabajo.
* Dispositivos de apoyo a la calidad (plantillas, listas de verificación).
* Capacitación, instrucción y certificación del personal.
* Acciones preventivas y correctivas.
* Gestión de la configuración.
* Control de documentación.

Tareas de apoyo a la subunidad SQA.

* Elaboración de planes de proyectos y planes de calidad de proyectos.
* Equipos de revisión de dotación de personal.
* Elección de metodologías y herramientas de desarrollo que reflejen el fracaso.
* Datos de experiencia acumulados por la unidad SQA.
* Elección de medidas para resolver los riesgos de desarrollo de software identificados.
* Elección de medidas para resolver los retrasos en el cronograma y los sobrecostos presupuestarios.
* Elección de métricas de SQA y componentes de costos de software.
* Uso de los sistemas de información de SQA.

Estándares y procedimientos de la subunidad SQA: desarrollo y tareas de mantenimiento.

* Preparar un programa anual para el desarrollo de nuevos procedimientos y actualizaciones de procedimientos, incluyendo.
* Seguimiento de desarrollos y cambios en SQA e ingeniería de software normas; introducción de procedimientos adicionales y cambios relevantes para la organización.
* Inicio de actualizaciones y adaptaciones de procedimientos en respuesta a cambios en los estándares profesionales, incluida la adopción o eliminación de estándares aplicados por la organización.

Desarrollo de ingeniería de la subunidad SQA y tareas de mantenimiento.

* Probar aspectos de calidad y productividad con respecto al nuevo desarrollo herramientas y nuevas versiones de las herramientas de desarrollo utilizadas actualmente.
* Evaluación de la calidad y productividad de nuevos métodos de desarrollo y mantenimiento y mejoras de métodos.
* Desarrollo de soluciones a las dificultades enfrentadas en la aplicación de herramientas y métodos de desarrollo de software utilizados actualmente.
* Desarrollo de métodos para medir la calidad del software y el equipo productividad.
* Provisión de soporte tecnológico a los comités del CAB durante el análisis de fallas en el desarrollo de software y formulación de propuestas de solución.

Tarea de los fideicomisarios de SQA.

* Apoyar los intentos de sus colegas para resolver las dificultades que surjan en la implementación de procedimientos de calidad de software e instrucciones de trabajo.
* Ayudar a su gerente de unidad a realizar sus tareas de SQA (p. ej., preparación de las instrucciones de trabajo de un proyecto, recopilación de datos para calcular métricas SQA).
* Promover el cumplimiento y monitorear la implementación de los procedimientos de SQA e instrucciones de trabajo de los compañeros.
* Reportar eventos de incumplimiento sustanciales y sistemáticos a la unidad SQA.
* Informar fallas graves en la calidad del software a la unidad SQA.

## Responsabilidades de la Unidad de SQA

Responsabilidades de la organización con respecto a SQA.

* Gerente de SQA.
  + Preparación del programa y presupuesto anual de las actividades de SQA.
  + Preparación de los planes de desarrollo de los sistemas SQA.
  + Control de la implementación del programa anual de actividades de SQA y de los proyectos de desarrollo planificados de SQA.
  + Intervenir para resolver y minimizar problemas en las actividades de SQA.
  + Asegurar la calidad de los productos de la compañía y de los servicios de mantenimiento de software.
  + Asegurar el funcionamiento satisfactorio y el pleno cumplimiento de los requisitos del cliente.
  + Asegurar que los objetivos del sistema de SQA se establezcan y se realicen.
  + Asegurar la disponibilidad de los recursos para los sistemas de SQA.
* Líder de unidad de SQA.
  + Planear la propuesta de actividades de SQA, así como el presupuesto destinado a la unidad de SQA.
  + Preparación de los planes de desarrollo de los sistemas SQA.
  + Control del desempeño del programa anual de actividades SQA del departamento y proyectos de desarrollo.
  + Monitoreo de la implementación del programa de SQA.
  + Participar en comités, juntas y consultas con otros gerentes de la compañía.
  + Presentación de las cuestiones de SQA del departamento a la alta dirección.
* Operación de SQA.
  + Realizar las tareas relacionadas al ciclo de vida del proyecto de SQA.
  + Realizar las operaciones de infraestructura de SQA.
  + Realizar las auditorías y certificaciones pertinentes.
  + Soporte de SQA realizando consultas sobre la calidad del plan del proyecto.
* Desarrollo y mantenimiento de SQA.
  + Realizar las tareas relacionadas a los estándares y procedimientos de SQA.
  + Realizar las tareas de ingeniería de SQA.
  + Realizar las tareas del sistema de información SQA.
* Comités de SQA.
  + Realizar las tareas de Control de cambio de software, acciones correctivas y procedimientos.
  + Realizar las tareas de actualización de procedimientos específicos, análisis y soluciones a fallas de software, la elaboración de métricas y la actualización de costos de software.
* Fideicomisarios.
  + Realizar actividades de soporte a otras unidades o equipos en la resolución de problemas.
  + Realizar las tareas relacionadas con la organización.

## Recursos estimados para el aseguramiento de la calidad

Para llevar a cabo el plan de aseguramiento se ocupará recursos humanos tales como los miembros del equipo de SQA, para el aseguramiento de las diferentes actividades que se realizarán en el plan se necesitarán estándares y herramientas.

# 

# Documentación

## Propósito

Identificar la documentación relativa al desarrollo, verificación, validación, uso y mantenimiento del software.

Establecer como los documentos van a ser revisados para verificar consistencia.

## Documentación mínima requerida

Para asegurar que la implementación de la aplicación satisface los requerimientos definidos es necesaria la elaboración de los siguientes documentos.

### 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, así como interfaces externas, deberá ser creado siguiendo la plantilla del documento ERS.docx.

La revisión de este documento se hará utilizando una revisión técnica formal. El plan de revisión será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

Para la verificación del documento se usará el estándar IEEE 1012 - 2016.

Para la validación del documento se usará el estándar IEEE 1012 - 2016.

### Plan de Pruebas de Sistema

El Plan de Pruebas de Sistema debe identificar las pruebas requeridas para el cumplimiento de los requerimientos especificados. La elaboración del plan de pruebas de sistema se hará de acuerdo al estándar de la IEEE 829-2008 “Software and System Test Documentation”.

La revisión de este documento se hará utilizando una revisión técnica formal. El plan de revisión será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

Para la verificación del documento se usará el estándar IEEE 1012 - 2016.

### Manual de Usuario

El Manual de Usuario deberá especificar y describir los datos requeridos y las entradas de control, las secuencias de entrada, las opciones, las limitaciones del programa y otras actividades o elementos necesarios para la ejecución exitosa del software. Para la creación del manual de usuario se utilizará la guía para la documentación de usuarios IEEE std 1063-2001 “Software User Documentation”.

La revisión de este documento se hará utilizando una revisión de recorridos. El plan de revisión será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

Para la verificación del documento se usará el estándar IEEE 1012 - 2016.

### Manual de Mantenimiento

El Manual de Mantenimiento describe los procedimientos a realizar para el mantenimiento y soporte del producto de software que se desarrollará en el proyecto. Para la creación del manual de mantenimiento se utilizará la guía para la documentación de usuarios IEEE std 1219-1998 “Software Maintenance”.

Para la verificación del documento se usará el estándar IEEE 1012 - 2016.

### Plan de Pruebas de Integración

El Plan de Pruebas de Integración debe contener el orden de integración de los componentes o subsistemas y pruebas que se aplicarán para verificar la interacción entre los componentes. Para la creación del plan de pruebas de integración se utilizará la guía IEEE Std 1008-1997 “Software Unit Testing”.

La revisión de este documento se hará utilizando una revisión técnica formal, según el estándar IEEE 1028 - 2002.

Para la verificación del documento se usará el estándar IEEE 1012 - 2016.

### 

### Descripción del diseño del software

La Descripción del Diseño del software específica como el software será construido para satisfacer los requerimientos. Deberá describir los componentes y subcomponentes del diseño del software, incluyendo interfaces internas. Para la descripción del diseño de software se utilizará el estándar IEEE 1016-2009 “Information Technology—Systems Design—Software Design Descriptions”.

El cliente deberá obtener como resultado del proyecto el diseño de un producto de software que cubra aquellos aspectos que se hayan acordado con el cliente incorporar al diseño, en función de la importancia que se le haya asignado a cada uno de ellos.

### Plan de Verificación y Validación

El Plan de Verificación y Validación deberá describir e identificar 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. En este caso sería que cumplan con el acuerdo logrado entre el cliente y el equipo.
* 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 está implementado en código.

Validar que el código, cuando es ejecutado se adecua a los requerimientos expresados en el documento de requerimientos.

Para la elaboración del Plan de Verificación y Validación del se utilizará el estándar IEEE 1012-2016 “System, Software, and Hardware Verification and Validation”.

### Plan de Gestión de configuración

El Plan de Gestión de Configuración debe contener métodos para identificar componentes se software, control e implementación de cambios, registro y reporte del estado de los cambios implementados. Para realizar el Plan de Gestión de Configuración se debe seguir el estándar IEEE 828-2012 “Configuration Management in Systems and Software Engineering”.

### Plan de desarrollo de software

El objetivo del Plan de Desarrollo de software es la definición de las actividades de desarrollo en términos de las fases necesarias para la implementación de un servicio.

### Manual de Operación

El Manual de Operación debe contener la información indispensable para la instalación y administración del software, así como el ambiente de operación (sistema operativo, base de datos, servidores, etc.). Esté deberá ser redactado en términos comprensibles al personal responsable de la operación

Para la verificación del documento se usará el estándar IEEE 1012 - 2016.

# 

### Análisis y Diseño

El Análisis y Diseño debe contener la descripción textual y gráfica de la estructura de los componentes de software, la cual consta de la arquitectónica y la detallada.

Para la verificación del documento se usará el estándar IEEE 1012 - 2016.

Para la validación del documento se usará el estándar IEEE 1012 - 2016.

### Registro de rastreo

El Registro de Rastreo debe contener la relación entre los requerimientos, elementos análisis y diseño, componentes y planes de pruebas.

Para la verificación del documento se usará el estándar IEEE 1012 - 2016.

### Reportes de Verificación y Validación

Los Reportes de Verificación y Validación deben especificar los resultados de la ejecución de los procesos descritos en el Plan de Verificación y Validación.

# 

# Estándares, prácticas, convenciones y métricas

## Propósito

En esta sección se identifican los estándares, prácticas, convenciones y métricas que todos los involucrados deben utilizar en cada fase del ciclo de vida.

## Fase de requerimientos

Conjunto de actividades cuya finalidad es lograr la documentación de la Especificación de Requerimientos y Plan de pruebas de Sistema, así como un avance del manual del usuario, para alcanzar un mismo entendimiento entre el cliente y el proyecto.

|  |  |
| --- | --- |
| **Estándar/Práctica/Convención** | **Uso** |
| IEEE 1063-2001 | Realización y modificación de la documentación del usuario. |
| IEEE Std 610.12 1990 | Es un glosario de la terminología de la ingeniería del software. |
| IEEE Std 829-2008 | Describe la forma y el contenido de un conjunto básico de documentación para la planificación, ejecución, y la presentación de las pruebas del software. |
| IEEE Std 830-1998 | Recomienda el contenido y características de una especificación de requisitos del software. |
| IEEE Std 1028-2002 | Define 5 tipos de revisiones del software y los procesos para su ejecución. Los tipos de revisión incluyen gestiones de las revisiones, revisiones técnicas, inspecciones, visitas guiadas y auditorías. |

## Fase de Análisis y Diseño

Conjunto de actividades en las cuales se examinan los requerimientos especificados para crear una descripción de la estructura de los componentes de software, la cual servirá de base para la construcción. Como consecuencia se obtiene la documentación del Análisis y Diseño y Plan de Pruebas de Integración.

|  |  |
| --- | --- |
| **Estándar/Práctica/Convención** | **Uso** |
| IEEE Std 610.12 1990 | Es un glosario de la terminología de la ingeniería del software. |
| IEEE Std 1016-2009 | Recomienda el contenido y la organización de una descripción del diseño de un software. |
| IEEE Std 1028-2002 | Define 5 tipos de revisiones del software y los procesos para su ejecución. Los tipos de revisión incluyen gestiones de las revisiones, revisiones técnicas, inspecciones, visitas guiadas y auditorías. |
| IEEE Std 982.1-1998 | Provee un conjunto de medidas para evaluar la fiabilidad de un producto software y para obtener fiabilidad de un producto en desarrollo. |
| Estándar de diseño | Documento construido basado en la publicación de Object Management Group (OMG) |

## 

## Fase de Construcción

Conjunto de actividades para crear Componente(s) de software que correspondan al Análisis y Diseño, así como la ejecución de pruebas unitarias. Como consecuencia se obtienen los Componentes de software probados.

|  |  |
| --- | --- |
| **Estándar/Práctica/Convención** | **Uso** |
| IEEE Std 610.12 1990 | Es un glosario de la terminología de la ingeniería del software. |
| IEEE 829-2008 | Describe la forma y el contenido de un conjunto básico de documentación para la planificación, ejecución, y la presentación de las pruebas del software. |
| IEEE Std 1008-1997 | Describe con cierta aproximación las pruebas unitarias del software, así como los conceptos y supuestos en los cuales están basados. |
| IEEE Std 1028-2002 | Define 5 tipos de revisiones del software y los procesos para su ejecución. Los tipos de revisión incluyen gestiones de las revisiones, revisiones técnicas, inspecciones, visitas guiadas y auditorías. |
| IEEE Std 982.1-1998 | Provee un conjunto de medidas para evaluar la fiabilidad de un producto software y para obtener fiabilidad de un producto en desarrollo. |
| IEEE Std 1061-1992 | Describe la metodología para establecer los requisitos de calidad, y para identificar, implementar y validar las medidas correspondientes. |
| Prácticas de Codificación | Documento construido basado en las prácticas recomendadas por Google JavaScript Style Guide |

## Fase de Integración y Pruebas

Conjunto de actividades para integrar y probar los componentes de software, basados en los Planes de Prueba de Integración y de Sistema, con la finalidad de conseguir el software que satisfaga los requerimientos especificados. Se genera la versión final del Manual de Usuario, Manual de Operación y Manual de Mantenimiento.

|  |  |
| --- | --- |
| **Estándar/Práctica/Convención** | **Uso** |
| IEEE Std 1008-1997 | Describe con cierta aproximación las pruebas unitarias del software, así como los conceptos y supuestos en los cuales están basados. |
| IEEE Std 610.12 1990 | Es un glosario de la terminología de la ingeniería del software. |
| IEEE 829-2008 | Describe la forma y el contenido de un conjunto básico de documentación para la planificación, ejecución, y la presentación de las pruebas del software. |
| IEEE Std 982.1-1998 | Provee un conjunto de medidas para evaluar la fiabilidad de un producto software y para obtener fiabilidad de un producto en desarrollo. |

## Métricas seleccionadas de productos y procesos de aseguramiento de la calidad del software

El proceso de medición se debe realizar de acuerdo con el estándar IEEE 15939-2007.

# 

# Revisiones y Auditorías de software

## Objetivo

Se utilizarán las Revisiones Técnicas Formales que tienen como objetivo descubrir errores en la función, la lógica ó la implementación de cualquier producto de software, verificar que satisface sus especificaciones, que se ajusta a los estándares establecidos, señalando las posibles desviaciones detectadas.

En la reunión deben participar el líder de la Unidad 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.

## Requerimientos mínimos

Como mínimo deberán revisarse todas las entregas semanales, basado en los estándares definidos. Estas revisiones serán realizadas por el responsable de SQA.

## Revisión de requerimientos

Se aplicará una Revisión Técnica Formal para verificar la claridad de la redacción de la Especificación de Requerimientos y su consistencia con la descripción del producto y con el estándar requerido. Adicionalmente, se revisará que los requerimientos sean completos y no ambiguos o contradictorios. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión del Plan de Pruebas de Sistema

Se aplicará una Revisión Técnica Formal para verificar consistencia del Plan de Pruebas de Sistema con la Especificación de Requerimientos y con el estándar de documentación requerido en el proceso específico. Adicionalmente, los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión de Manual de Usuario

Se aplicará una Revisión por recorrido para verificar consistencia del Manual de Usuario con la Especificación de Requerimientos y con el estándar de documentación requerido en el proceso específico. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión Análisis y Diseño

Se aplicará una Revisión Técnica Formal para verificar la claridad de la documentación del Análisis y Diseño, su factibilidad y la consistencia con la Especificación de Requerimientos y con el estándar de documentación requerido en el proceso específico. Adicionalmente, se debe verificar que el Registro de Rastreo contenga las relaciones adecuadas entre los requerimientos y los elementos de Análisis y Diseño. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión de Plan de Pruebas de Integración

Se aplicará una Revisión Técnica Formal para verificar consistencia del Plan de Pruebas de Integración con el Análisis y Diseño y con el estándar de documentación requerido en el proceso específico. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión de Registro de Rastreo

Se aplicará una Revisión Técnica Formal para verificar que el Registro de Rastreo contenga las relaciones adecuadas entre los elementos de Análisis y Diseño y los componentes. Los defectos encontrados se documentan en el Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión de Manual de Operación

Se aplicará una Revisión Técnica Formal para verificar la consistencia del Manual de Operación con el Software y con el estándar de documentación requerido en el proceso específico. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión de Manual de Usuario

Se aplicará una Revisión Técnica Formal para verificar la consistencia del Manual de Usuario con el sistema de software y con el estándar de documentación requerido en el proceso específico. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión de Manual de Mantenimiento

Se aplicará una Revisión Técnica Formal para verificar la consistencia del Manual de Mantenimiento con la Configuración de Software y con el estándar de documentación requerido en el proceso específico. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión del Plan de gestión de configuración

Se aplicará una Revisión Técnica Formal para asegurar la consistencia y completitud de los métodos especificados en el Plan de gestión de configuración. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## 

## Revisiones de gestión

Se aplicará una Revisión Técnica Formal para asegurar la ejecución de todas las actividades identificadas en este Plan. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## Revisión de diseño preliminar

Se aplicará una Revisión Técnica Formal para evaluar el progreso, la consistencia y la adecuación técnica del diseño de alto nivel seleccionado, el enfoque de prueba, la compatibilidad entre los requisitos del software y el diseño preliminar, y la versión preliminar de los documentos de operación y soporte. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## 

## Revisión del Plan de Verificación y Validación

Se aplicará una Revisión Técnica Formal para asegurar la consistencia y completitud de los métodos especificados en el Plan de Verificación y Validación. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

## 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 y satisfacen las necesidades del cliente.

## Auditoría física

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

## Auditorías internas al proceso

En esta auditoría se verifica que haya consistencia entre el código y el documento de diseño, especificaciones de interfaces, implementaciones de diseño y los requerimientos funcionales, además de requerimientos funcionales versus descripciones de testeo.

## Revisión Post-implementación

Se aplicará una Revisión Técnica Formal al concluir el proyecto para especificar las actividades de desarrollo implementadas durante el proyecto para detectar posibles mejoras en el proceso de desarrollo. Los defectos encontrados se documentan en un Reporte de Verificación.

La descripción del proceso para la ejecución de la revisión será definida en el Plan de Revisión “Plan de Revisión.docx”, que será elaborado con base en el estándar IEEE Std 1028-2002, IEEE Standard for Software Reviews.

# 

# 

# Pruebas

Las diferentes pruebas se realizarán de acuerdo al documento **PlanDePruebas-2022.docx**, el cual fue construido siguiendo el estándar **IEEE 829-2008**.

Las actividades de prueba que se realizarán en el proyecto son:

* Prueba unitaria: Se encargan de pequeñas unidades o módulos de software.
* Prueba de integración: Se encargan de varias unidades que se combinan en un subsistema.
* Pruebas del sistema: Se encargan de todo el sistema o paquete de software.
* Prueba de estrés: Se sobrecarga el sistema para encontrar un punto de ruptura.
* Prueba de carga: Mide el rendimiento del sistema con una carga esperada.

El procedimiento de aplicación de estas pruebas se encuentra definido en el plan de pruebas anterior mencionado.

# 

# Informes de problemas y acciones correctivas

## Propósito

Describir los lineamientos para el reporte y control de las acciones correctivas para evitar la ocurrencia de no conformidades, así como para la implementación y seguimiento de las acciones de mejora necesarias.

El procedimiento aplica desde el momento en que se detecta una no conformidad real hasta la verificación de la eficacia de la acción de corrección.

## Objetivo

* Garantizar que las no conformidades se documenten, tengan un seguimiento, y sean corregidos adecuadamente.
* Garantizar que los informes de los problemas y las acciones correctivas sean correctamente evaluadas y validadas.

## Identificación de no conformidades

Cualquier miembro de la unidad de SQA puede identificar una no conformidad, la cual se debe registrar en el formato **Reporte de no conformidad en el cual se detallan aspectos como**:

* Fecha de la identificación de la no conformidad.
* Identificación de la no conformidad.
* Descripción, etc.

Para mayor detalle consultar en el Anexo A1 el Reporte de no conformidad.

Se pueden identificar no conformidades a partir de las siguientes fuentes:

* Reportes de las revisiones y auditorías.

Este reporte deberá ser firmado como significado de acuerdo por las personas que identificaron el problema y deberá ser entregado al líder de Unidad de SQA.

## Presentación del reporte al líder de Unidad de SQA

Una vez que el reporte esté llenado por completo este deberá ser entregado al líder de Unidad de SQA. Se le debe describir el problema y posteriormente él debe dar las recomendaciones necesarias para solucionar el problema. El líder de la Unidad utiliza el reporte para:

* Establecer las actividades a realizar, orientadas a eliminar la causa raíz definida para la no conformidad.
* En caso de ser necesario, puede hacer modificaciones a los procesos para su mejora.

## Implementación de Acciones correctivas

## 

## Las acciones correctivas que se generaron en la reunión de revisiones o auditorías, se deben entregar al equipo de trabajo que fue o fueron afectados por el problema de esta manera podrán implementar las acciones correctivas.

## 

## 

# Herramientas, técnicas y metodologías

Esta sección detalla las diferentes herramientas y técnicas tales como suites ofimáticas, depuradores, procesadores de texto, analizadores de código, herramientas de pruebas de carga, monitoreos de rendimiento, generadores de casos de prueba, herramientas para prototipos, herramientas para la gestión de configuración.

## Herramientas

* **Apache JMeter:** Es un software hecho en Java y de código abierto. Se utilizará para cargar, probar el comportamiento funcional y medir el rendimiento del sistema.
* **Jira:** Herramienta en línea para la administración de tareas de un proyecto, el seguimiento de errores e incidencias y para la gestión operativa de proyectos. Se utilizará para la gestión y mejora de los procesos, gracias a sus funciones para la organización de flujos de trabajo.
* **Figma:** Herramienta de prototipado web y editor de gráficos vectorial, que se aloja en la web.
* **Canva:** Web de diseño gráfico y composición de imágenes para la comunicación.
* **JavaScript:** Lenguaje de programación interpretado, se utilizará principalmente del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario.
* **TypeScript:** Lenguaje de programación libre y de código abierto, es usado para desarrollar aplicaciones JavaScript que se ejecutarán en el lado del cliente o del servidor, o extensiones para programas.
* **HTML:** Lenguaje de marcado que se utilizará para estructurar y desplegar una página web y sus contenidos.
* **CSS:** Lenguaje de estilos utilizado para describir la presentación de documentos HTML o XML. Describe cómo debe ser renderizado el elemento estructurado en la pantalla, en papel, en el habla o en otros medios.
* **Visual Studio Code:** Editor de código fuente que permite trabajar con diversos lenguajes de programación, admite gestionar tus propios atajos de teclado y refactorizar el código.
* **Microsoft Word:** Es un procesador de textos, utilizado para la creación de una amplia variedad de documentos profesionales de forma rápida y sencilla.
* **Microsoft Excel:** Hoja de cálculo, utilizado para obtener información con significado a partir de grandes cantidades de datos, realizar cálculos sencillos y para realizar el seguimiento de casi cualquier tipo de información.
* **Git**: Para el control de versiones se sugiere el uso de un sistema de control de versiones como Git. Es un software pensado en la eficiencia, confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente.
* **GitHub:**  GitHub es un servicio de alojamiento de repositorios de software que usa Git y que ha sido pensado para compartir código de manera ágil.

## 

## Técnicas

* **IEEE Std 830-1998, IEEE Standard Recommended Practice for Software Requirements Specifications:** Describe los enfoques recomendados para la especificación de requisitos de software y se utilizará para la elaboración del mismo.
* **IEEE Std 1016-2009, IEEE Standard for Information Technology Systems Design, Software Design Descriptions:** Especifica los requisitos sobre el contenido de la información y la organización de las descripciones de diseño de software (SDD). Se utilizará para la descripción del diseño de software.
* **IEEE Std 1008-1987, IEEE Standard for Software Unit Testing:** Define un enfoque integrado para las pruebas unitarias sistemáticas y documentadas. Se utilizará para la aplicación de las pruebas.
* **IEEE Std 1063-2001, IEEE Standard for Software User Documentation:** Establece ciertos formatos de ilustraciones, instrucciones, entre otros. Se utilizará para la documentación del usuario.
* **IEEE Std 1028-2002, IEEE Standard for Software Reviews:** Describe cómo llevar a cabo una revisión. Se utilizará para la realización de las revisiones y también para el proceso de auditorías.

## Metodologías

Las metodologías de Aseguramiento de Calidad serán conjuntos integrados de técnicas y herramientas de las mencionadas anteriormente. Como principales metodologías se utilizarán:

* **Métodos orientados a objetos:** El sistema es visto como una colección de objetos más que de funciones.
* **Diseño centrado en el usuario:** Es un proceso iterativo que se enfoca en la comprensión de los usuarios y su contexto en todas las etapas de diseño y desarrollo.

# 

# Control de Código

## Propósito

Definir los métodos que se utilizarán para controlar y dar mantenimiento a versiones de código.

El equipo de desarrollo se debe asegurar de:

* Realizar copias de seguridad al código del proyecto.
* Realizar el manejo de versiones adecuadamente.
* El almacenamiento y recuperación del código está garantizado.

## Documentación afectada por el control de código

* Especificación de requisitos
* Plan de verificación y validación de software
* Documentación del usuario

## Elemento de configuración

* Código fuente

## Mantenimiento y control de la gestión de la configuración

Cuando se solicite un cambio que afecta al código fuente bajo gestión de configuración (línea base), debe entrar en funcionamiento el proceso de control de cambios. En este proceso, que se detalla a continuación, se deben identificar y valorar los cambios y, si son admitidos, modificar los productos afectados, siguiendo el procedimiento establecido. Estos cambios realizados deben comunicarse a todas las personas que resulten afectadas por los mismos.

## Proceso de control de cambios

Las peticiones de cambio sobre elementos de configuración pueden iniciarse por cualquiera y en cualquier punto del proceso de desarrollo y pueden incluir una sugerencia de solución y prioridad de la petición. Se deberá generar una solicitud de petición de cambio. Consultar Anexo A2. Solicitud de Cambio de Software para ver el formato de la solicitud.

La autoridad responsable de aceptar o rechazar los cambios propuestos reside en el comité de control de configuración (CCB).

Una vez aprobado el cambio, deberá generarse el documento de identificación del código. Consultar Anexo A3. Software Configuration Item Version Document para ver el formato.

Adicionalmente, se debe contar con un documento para el historial de las versiones de cada liberación de software. Consultar Anexo A4. Software Configuration Release Documentation para ver la estructura del documento.

## Recuperación de los elementos de configuración

Para la recuperación de los elementos de la configuración es necesario la definición de un repositorio general del proyecto utilizando la herramienta GitHub.

La estructura del desarrollo se conformará por dos ramas principales:

* **master:** Esta rama contendrá todas las versiones estables del producto, con lo cual cualquier *commit* que se realice sobre ella, implica que el producto está listo para ser usado por los usuarios. Este paso implica que el producto haya pasado el conjunto de pruebas correspondiente y que esté debidamente documentado.
* **develop:** En esta rama se aloja el código que conformará la siguiente versión. Sobre esta rama se trabajará con versiones inestables, que buscan agregar los cambios necesarios para cumplir con los requerimientos.

Además de estas dos ramas principales se sugiere el uso de ramas de soporte. Dichas ramas son:

* **Feature**: Nuevas características del producto.
* **Release**: Últimos cambios antes de pasar a producción.
* **Hotfix**: Cambios en “caliente”, corrección de *bugs* reportados mediante el uso de *tickets* en versiones del servicio que se encuentran en la rama master.

Se sugiere que las ramas adopten la siguiente forma: *nombrerama/descripción*, en donde el *nombrerama* identifica algunas de las ramas mencionadas anteriormente, y descripción es un texto breve que identifica al requerimiento.

Reglas para las distintas ramas para su correcta utilización:

***Feature***

* Se origina a partir de la rama *develop*, dado que se emplea el código que actualmente se está empleando.
* Se incorporan siempre a la rama *develop*. Una vez hecho esto, se deberán eliminar las ramas que ya no trabajarán sobre esa *feature*. De esta forma se busca un esquema de trabajo limpio en donde no se tenga constantemente múltiples ramas, sino que existan las ramas *master* y *develop*, y que el resto sean temporales.
* El nombre de esta rama puede ser cualquiera, pero no debe entrar en conflicto con el nombre de *master*, *develop*, *release* o *hotfix*.

***Release***

* El nombre de la rama deberá corresponder con un *tag* que se ajuste al esquema de identificación definido más adelante.
* Se originan a partir de la rama *develop*.
* Se incorporan a *master* y *develop*. La rama dejará de existir una vez que se efectúe el *merge* sobre dichas ramas, con lo cual las versiones estables quedarán sobre la rama *master* y el trabajo en curso sobre la rama *develop*. En caso de existir *bugs* de último momento, se creará la rama de tipo *Hotfix*.

***Hotfix***

El uso de esta rama permite:

* Resolver el *bug*.
* Incorporar la corrección del bug en la rama *master* para desplegarlo en producción lo más rápido posible. También permite identificar sobre la rama *master* que ha habido una corrección, modificando el número de *patch* en el identificador de la versión.
* Incorporar la corrección del *bug* en la rama *develop*, si es necesario.
* Retomar el trabajo en la rama *develop*.

## Identificaciones de versiones

## 

El esquema sugerido para la identificación de las distintas versiones del producto es un esquema numérico conformado de la siguiente manera:

## 

*<major\_version>.<minor\_version>.<patch\_number>*

* **Major Version:** Indica un cambio de funcionalidad sustancial a otras versiones del sistema.
* **Minor Version**: Indica que el sistema es funcionalmente idéntico, pero es distinto desde el punto de vista no-funcional a otras versiones.
* **Patch Version**: Indica el número de un parche aplicado al sistema, que significa la corrección de un defecto encontrado.

# Control de medios

## Propósito

Definir los métodos que se utilizarán para proteger el medio físico de accesos no autorizados, así como de daños y degradaciones no voluntarias. Además, de asignar a los responsables para llevar a cabo el control.

El equipo de desarrollo de software deberá asegurar que:

* El almacenamiento y recuperación de software está garantizado.
* El software es accesible únicamente para aquellas personas autorizadas.
* Realizar copias de seguridad del software crítico.
* Se controla el entorno para que el medio físico donde está almacenado el software no se degrade.

## Medio de almacenamiento

El medio del programa de computadora se define como aquellos medios sobre los cuales los datos son almacenados.

En la organización se utilizarán los siguientes medios:

* Un servidor tipo NAS como dispositivo primario, dado que es un dispositivo de almacenamiento conectado a la red que permite a los usuarios acceder a todo su contenido de forma sencilla.
* Los discos duros y servicio en la nube para guardar las copias de seguridad.
* La documentación producto de cada fase del ciclo de vida del software.

## Procesos de copias de seguridad

Para realizar una copia de seguridad de NAS se utilizará un proceso automatizado que permita que la misma se realice en el horario establecido por el gerente de desarrollo.

Se deben realizar dos copias distintas de los datos.

* Guardar las copias de seguridad en un disco duro. Estas copias estarán en el lugar de trabajo y podrán ser utilizadas en caso de necesidad.
* Una de las copias debe guardarse en una locación distinta, el almacenamiento en la nube se utilizará para este propósito.

## Acceso no autorizado

Se deben asignar cuentas privilegiadas para el acceso no autorizado, cada usuario tendrá su propia cuenta de acuerdo con el cargo que desempeñe dentro de la organización.

# Recopilación, mantenimiento y retención de registros

Los documentos a los que se le aplicará el control de la documentación son los siguientes:

* Plan de Aseguramiento de la calidad de software
* Descripción del diseño de software
* Especificación de requerimientos
* Plan de verificación y validación de software
* Documentación del usuario

El mantenimiento de los registros de software será realizado por versiones de actualización de estas, para poder acabo esto se realiza un control de documentación.

Los documentos verificados y validados deben ser documentados digitalmente, se crearán 2 copias que se almacenarán en un servidor tipo NAS y un disco duro respectivamente. La retención de registros se realizará en cada finalización de las fases del ciclo de vida de desarrollo de software y según los puntos de verificación y validación.

## Procedimiento de control de la documentación

### 

### Lista de documentos controlados

* + El comité de SQA debe decidir y clasificar los documentos como documentos controlados o registros de calidad.
  + Decidir si el nivel de control es adecuado para cada tipo de documento categorizado como un documento controlado.
  + Se debe incluir en las auditorías internas de calidad mencionadas anteriormente, el seguimiento del cumplimiento de la lista de tipos de documentos controlados.

### Preparación de documentos controlados

Para la creación de un nuevo documento o revisión de alguno existente se deben seguir los siguientes requisitos.

* La estructura del documento debe seguir el formato establecido en el documento GuiaTecnica.doc
* El responsable de la revisión debe remitir el documento al responsable de la aprobación.
* Cada documento debe tener una versión y revisión con identidad única.

|  |
| --- |
| **Nombre del producto** |
| **Tipo de documento** |
| **Número de versión y revisión** |

* Cada documento debe proporcionar la siguiente información para futuros accesos

|  |  |
| --- | --- |
| **Autores** |  |
| **Fecha de finalización** |  |
| **Confidencialidad** |  |

**Histórico de versiones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha de aprobación** | **Responsables de aprobación y puestos** | **Nombre del archivo** |
|  |  |  |  |

**Histórico de cambios**

|  |  |  |
| --- | --- | --- |
| **Versión** | **Fecha** | **Descripción de los cambios** |
|  |  |  |

# 

# 

# Entrenamiento

La organización de SQA deberá contar con una serie de características que permitan garantizar la calidad tanto en el proceso como en sus productos finales, por lo que la unidad de SQA deberá contar con experiencia.

Cada integrante de la unidad de SQA deberá contar con las siguientes habilidades respecto a la tarea que éste desempeñe como se muestra en la tabla siguiente:

|  |  |
| --- | --- |
| **Tareas** | **Habilidades requeridas** |
| Revisiones de código | * Lenguaje de programación * Revisiones por pares |
| Revisiones de la documentación | * Metodología de desarrollo de software * Documentación de estándares * Revisiones por pares * Revisión Técnica Formal |
| Auditorias del proceso | * Desarrollo de software del proceso de ciclo de vida |
| Pruebas de software | * Metodologías de pruebas |
| Administración de SQA | * Administración de proyectos |
| Métricas | * Colección de datos * Análisis de datos |
| Reporte de problemas y acciones correctivas | * Administración de la configuración de software   + Gestión de Informes   + Identificación de no conformidades   + Análisis de no conformidades |
| Herramientas | * Capacitación proporcionada por el proveedor |
| Código, medios y control de proveedores | * Administración de la configuración de software   + Manejo de versiones   + Copias de seguridad   + Control de entorno   + Manejo de registros |
| Gestión de riesgos | * Procesos de gestión de riesgos |

Cada integrante de la organización de SQA debe estar en constante adquisición de conocimientos y actualizaciones.

Además, la capacitación será de instrucción directa por los miembros de la unidad de SQA permitiendo de esta forma un mejor desempeño en la realización de las tareas y se buscará que se capaciten en las tareas que más prioridad tengan.

# 

# 

# Glosario

**SQA:** (Aseguramiento de la Calidad del Software) se define como un conjunto de actividades planificadas y sistemáticas, cuyo primer objetivo es evaluar la calidad y la adherencia de los productos de software a los estándares, procesos y procedimientos.

**IEEE:** (Instituto de Ingenieros Eléctricos y Electrónicos) es una asociación mundial de ingenieros dedicada a la normalización y el desarrollo en áreas técnicas.

**ISO:** (Organización Internacional de Normalización) es una organización cuya principal actividad es la elaboración de normas técnicas internacionales.

**Servidor NAS:** Es un dispositivo de almacenamiento conectado a la red. Su función es hacer copias de seguridad de los archivos que tú le indiques a la configuración.

**SQAP:** (Plan de Aseguramiento de la Calidad del Software) contiene una lista de comprobación para las actividades que se deben llevar a cabo para asegurar la calidad del producto.

**Git:** Es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente.

**GitHub:** Es un proveedor de alojamiento de Internet para el desarrollo de software y el control de versiones mediante Git. Ofrece el control de versiones distribuidas y la funcionalidad de gestión de código fuente de Git, además de sus propias características.

**Apache JMeter:** Es una herramienta de prueba de carga para analizar y medir el rendimiento de una variedad de servicios, con énfasis en aplicaciones web.

**Jira:** Es una herramienta en línea para la administración de tareas de un proyecto, el seguimiento de errores e incidencias y para la gestión operativa de proyectos.

# 

# Anexo

## A1. Reporte de no conformidad

# De Reporte: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_D\_/\_M\_/\_A\_ Hora: \_\_\_\_\_

Nombre del proyecto:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Identificación del problema:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Descripción:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

El evento ejecutado cuando se presentó el problema es:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Posibles orígenes del problema:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Acciones correctivas:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Equipo de trabajo

Nombre: Firma:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Firma Líder de Unidad de SQA

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## A2. Solicitud de Cambio de Software

**Descripción del cambio**

Nombre del proyecto: Nombre del cambio: Número:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Solicitado por: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Contacto: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_\_\_

Descripción del cambio:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Razón del cambio:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prioridad [Marca con “X”]: [ ] 1. Alta [ ] 2. Media [ ] 3. Baja

Impacto en los entregables:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Impacto de no responder al cambio (Explica por qué):

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Fecha necesaria: \_\_\_\_\_\_\_\_\_\_ Aprobación de solicitud: \_\_\_\_\_\_\_\_\_ Fecha: \_\_\_\_\_\_\_\_\_\_\_\_

**Impacto del cambio**

Tareas afectadas:

Evaluación de costos:

Evaluación de riesgos:

Evaluación de calidad:

Recursos adicionales:

Duración:

Esfuerzo adicional:

Impacto de la fecha límite:

Alternativas y recomendaciones:

Comentarios:

[ ] Aceptado [ ] Rechazado

Firma CCB:

\_\_\_\_\_\_\_\_\_\_\_\_\_

## A3. SCI Version Document

**Identificación**

Número de versión de SCI:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Nombre(s) de los ingenieros de software que implementarán el cambio:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Fecha de completada y aprobada la nueva versión:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Cambios en la nueva versión**

Número de versión anterior de SCI:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Breve descripción de los cambios introducidos:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Lista de otros SCIs que fueron modificados debido a los cambios actuales:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Lista de los reportes de problemas resueltos por la nueva versión:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Cambios operaciones y otras implicaciones introducidas en la nueva versión:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## A4. Software Configuration Release Documentation

**Identificaciones e Instalaciones**

Versión de liberación y número de revisión: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Fecha de liberación de la nueva versión: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Lista de instalaciones donde se ingresó la liberación (sitio, fecha, nombre de

técnico que instaló la versión), en su caso:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Configuración de la versión liberada**

Lista de SCI en la versión publicada, incluida la identificación de cada uno

versión SCI:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Lista de elementos de configuración de hardware necesarios para operar el especificado

versión, incluida la especificación de cada elemento de configuración de hardware:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Lista de sistemas de software de interfaz (incluida la versión) y hardware

sistemas (incluido el modelo):

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Instrucciones de instalación para la nueva versión:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Cambios en la nueva versión**

Versión de configuración de software anterior: \_\_\_\_\_\_\_\_\_\_\_\_

Lista de SCI que se han modificado, nuevos SCI introducidos por primera vez,

y SCI eliminados:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Breve descripción de los cambios introducidos:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Implicaciones operativas y de otro tipo de los cambios introducidos en el

nuevo lanzamiento:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Más problemas de desarrollo**

Lista de problemas del sistema de software que no se han resuelto en el

nueva versión:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Lista de SCR y propuestas de desarrollo del sistema de software para

cuya implementación del desarrollo se retrasó:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_