Interpool

Plan de SQA

Versión 1.3

Historia de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión | Descripción | Autor |
| 16/08/10 | 1.0 | Primera versión | Javier Madeiro |
| 20/08/10 | 1.1 | Actualización del documento. | Javier Madeiro |
| 26/08/10 | 1.2 | Actualización del documento. | Javier Madeiro |
| 29/08/10 | 1.2 | Revisión de Documento | Alejandro García |
| 03/09/10 | 1.3 | Actualización del Documento | Javier Madeiro |

Contenido

[1. Propósito 4](#_Toc270774448)

[2. Referencias 4](#_Toc270774449)

[3. Gestión 5](#_Toc270774450)

[3.1. Organización 5](#_Toc270774451)

[3.2. Actividades 7](#_Toc270774452)

[3.2.1. Ciclo de vida del software cubierto por el Plan 7](#_Toc270774453)

[3.2.2. Actividades de calidad a realizarse 7](#_Toc270774454)

[3.2.3. Revisar cada producto 7](#_Toc270774455)

[3.2.4. Revisar el ajuste al proceso 8](#_Toc270774456)

[3.2.5. Realizar Revisión Técnica Formal (RTF) 8](#_Toc270774457)

[3.2.6. Asegurar que las desviaciones son documentadas 8](#_Toc270774458)

[3.2.7. Relaciones entre las actividades de SQA y la planificación 8](#_Toc270774459)

[3.3. Responsables 9](#_Toc270774460)

[4. Documentación 10](#_Toc270774461)

[4.1. Propósito 10](#_Toc270774462)

[4.2. Documentación mínima requerida 10](#_Toc270774463)

[4.2.1. Especificación de requerimientos del software 10](#_Toc270774464)

[4.2.2. Descripción del diseño del software 12](#_Toc270774465)

[4.2.3. Plan de Verificación & Validación 12](#_Toc270774466)

[4.2.4. Reportes de Verificación & Validación 13](#_Toc270774467)

[4.2.5. Documentación de usuario 13](#_Toc270774468)

[4.2.6. Plan de Gestión de configuración 13](#_Toc270774469)

[4.2.7. Plan de Proyecto 13](#_Toc270774470)

[5. Estándares, prácticas, convenciones y métricas 13](#_Toc270774471)

[5.1. Estándar de documentación 13](#_Toc270774472)

[5.2. Estándar de verificación y prácticas 14](#_Toc270774473)

[5.3. Otros Estándares 14](#_Toc270774474)

[5.3.1. Estándares de programación 14](#_Toc270774475)

[6. Revisiones y auditorías 15](#_Toc270774476)

[6.1. Objetivo 15](#_Toc270774477)

[6.2. Requerimientos mínimos 15](#_Toc270774478)

[6.2.1. Revisión de requerimientos 15](#_Toc270774479)

[6.2.2. Revisión de diseño preliminar 15](#_Toc270774480)

[6.2.3. Revisión de diseño crítico 15](#_Toc270774481)

[6.2.4. Revisión del Plan de Verificación & Validación 15](#_Toc270774482)

[6.2.5. Auditoría funcional 15](#_Toc270774483)

[6.2.6. Auditoría física 15](#_Toc270774484)

[6.2.7. Auditorías internas al proceso 15](#_Toc270774485)

[6.2.8. Revisiones de gestión 15](#_Toc270774486)

[6.2.9. Revisión del Plan de gestión de configuración 15](#_Toc270774487)

[6.2.10. Revisión Post Mortem 16](#_Toc270774488)

[6.2.11. Agenda 16](#_Toc270774489)

[6.3. Otras revisiones 16](#_Toc270774490)

[6.3.1. Revisión de documentación de usuario 16](#_Toc270774491)

[7. Verificación 16](#_Toc270774492)

[8. Reporte de problemas y acciones correctivas 17](#_Toc270774493)

[9. Herramientas, técnicas y metodologías 17](#_Toc270774494)

[10. Gestión de riesgos 17](#_Toc270774495)

[Apéndices 18](#_Toc270774496)

[A. Documentación 18](#_Toc270774497)

[B. CheckLists 19](#_Toc270774498)

[I. Revisión de Requerimientos 19](#_Toc270774499)

[II. Revisión del Modelo de Casos de Uso 20](#_Toc270774500)

[III. Revisión de las Pautas para la Interfaz de Usuario 21](#_Toc270774501)

[IV. Revisión de la Interfaz de Usuario 21](#_Toc270774502)

[V. Revisión de la Arquitectura 21](#_Toc270774503)

[VI. Revisión de Descripción la Arquitectura 22](#_Toc270774504)

[VII. Revisión del Modelo de Diseño de los Casos de Uso 23](#_Toc270774505)

[VIII. Revisión para Diseño Orientado a Objetos 24](#_Toc270774506)

[IX. Revisión de la Gestión de Configuración del Software 24](#_Toc270774507)

[X. Revisión del Plan de Gestión de Configuración del Software 25](#_Toc270774508)

[XI. Revisión de la Implementación 25](#_Toc270774509)

[XII. Revisión del Plan de Proyecto 26](#_Toc270774510)

[XIII. Revisión de la Estimaciones y Mediciones 26](#_Toc270774511)

[XIV. Revisión de la Documento de Riesgos 26](#_Toc270774512)

[XV. Revisión del Plan de Verificación y Validación 27](#_Toc270774513)

[XVI. Revisión de la Documentación 27](#_Toc270774514)

Propósito

El propósito de este Plan de Calidad es definir y describir las pautas a seguir durante el proceso de desarrollo, con el fin de poder asegurar la calidad del mismo y del producto a elaborar, detallando todo lo referente a la planificación del seguimiento de la calidad en el proyecto.

Para cumplir con este cometido se definirán mecanismos para monitorear la calidad tanto de los productos como del proceso, conjuntamente se fijarán los niveles de calidad y las metas que se pretenden alcanzar.

Se buscara establecer métodos para tratar de detectar las desviaciones del proceso que potencialmente podrían significar defectos del producto final. Tomando una postura proactiva ante esto, nos permitirá solucionar los defectos en etapas donde no sea tan costosa su corrección.

El software a desarrollar es un juego para Windows Phone 7, el cual se basara en “Where in the world is Carmen San Diego?” con algunas adaptaciones como soporte para incluir la red social Facebook y permitir realizar búsquedas a través de Bing.

Es importante destacar, que este plan sólo cubre la parte del ciclo de vida correspondiente a las fases de evaluación, elaboración, construcción, pero no cubre la parte del ciclo de vida sobre la transición del software y mantenimiento.

Durante este proyecto tomaremos como base el modelo de calidad propuesto por la norma ISO 9126, haciendo énfasis la primera parte de la misma (ISO 9126-1) donde se clasifica la calidad del software en un conjunto estructurado de características, de las cuales para este proyecto un punto clave será la usabilidad, entendida como la capacidad de un producto de software en ser comprendido, aprendido, usado y a la vez de ser atractivo para el usuario, en condiciones específicas de uso. Se observa que la usabilidad no depende sólo del producto, sino también del usuario.

Para que este plan sea efectivo deberá de existir en todos los integrantes del equipo un alto grado de responsabilidad y compromiso por la calidad, para ello será fundamental una adecuada comprensión y apego a las directivas de este documento.

Referencias

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

[2] Modelo de proceso MUM.

[3] Memoria organizacional – SQAP.

[4] http://www.fing.edu.uy/inco/cursos/gestsoft

[5] http://www.infor.uva.es

[6] https://developer.windowsphone.com

[7] http://www.ingenierosoftware.com/calidad

[8] Documento de especificación de requerimientos.

[9] Acta de reunión de requerimientos.

[10] [http://www.calidaddelsoftware.com](http://www.calidaddelsoftware.com/)

[11] <http://www.rspa.com/checklists/>

Gestión

La Gestión de Calidad es la disciplina que se encarga de realizar el seguimiento de la calidad del producto en desarrollo, los procedimientos utilizados y el proceso seguido para el mismo. En esta sección se describe la organización, tareas y responsabilidades de aseguramiento de calidad.

* 1. Organización

Dentro de la organización del equipo las líneas de trabajo que tienen influencia y controlan la calidad del software se pueden clasificar en disciplinas básicas y disciplinas de gestión.

Las disciplinas básicas son las que involucran las actividades de ingeniería “tradicionales” de desarrollo de software y están divididas en:

* Requerimientos.
* Análisis.
* Diseño.
* Implementación.
* Verificación.

Por su parte, las disciplinas de gestión comprenden actividades que brindan “soporte” a las básicas y se realizan en forma paralela a éstas, dividiéndose en:

* SCM.
* SQA.
* Gestión de proyecto.
* Comunicación.

Todas las actividades que conciernen al aseguramiento y gestión de la calidad se realizarán bajo la supervisión del responsable de SQA y del asistente de SQA, los mismos se verán involucrados con las líneas de trabajo de la siguiente manera:

* La disciplina requerimientos hace partícipe al responsable de SQA de las reuniones de relevamiento de requerimientos, para que éste tenga conocimiento sobre el alcance del sistema que se pretende desarrollar y de las propiedades de calidad que el cliente requiere que se cumplan.
* La disciplina gestión de proyecto informa de las actividades que se realizarán y los resultados obtenidos referentes a la gestión de proyecto.
* La disciplina verificación informa de las actividades que se realizarán y los resultados obtenidos referentes a la verificación.
* La disciplina gestión de configuración y control de cambios informa sobre las actividades que se realizarán y los resultados obtenidos referentes a la gestión de configuración y control de cambio.
* Con el resto de las disciplinas se tendrá relación en la medida que evalúe los procedimientos seguidos para la realización de actividades y la calidad de los elementos o productos generados por cada disciplina.

El equipo de trabajo se divide según los siguientes roles:

|  |  |
| --- | --- |
| **Combinación de Roles** | **Nombre** |
| Administrador - Asistente de Verificación - Responsable de la Comunicación | Juan Carlos Ghiringhelli |
| Analista - Documentador de Usuario -Asistente de Verificación | José Cordero |
| Analista – Implementador | Diego Ricca |
| Analista - Implementador | Martín Taruselli |
| Responsable de SQA –  Ayudante de Verificación | Javier Madeiro |
| Analista-Diseñador de Interfaz de Usuario - Implementador | Leticia Vilariño |
| Responsable de Verificación –  Asistente de SQA | Alejando García |
| Arquitecto - Coordinador de Desarrollo - Asistente de Verificación | Marcos Sander |
| Especialista Técnico - Implementador | Federico Andrade |
| Especialista Técnico - Implementador | Federico Trinidad |
| Especialista Técnico - Implementador - Responsable de Integración | Ignacio Infante |
| Responsable de SCM –  Especialista Técnico - Implementador | Vicente Acosta |

De manera complementaria, se han definido responsables para cada una de las líneas de trabajo:

|  |  |  |
| --- | --- | --- |
|  | **Disciplina** | **Nombre** |
| **Básicas** | Requerimientos - Análisis | Diego Ricca |
| Diseño | Marcos Sander |
| Implementación | Ignacio Infante |
| Verificación | Alejando García |
| Implantación | Vicente Acosta |
| **Gestión** | Gestión de configuración | Vicente Acosta |
| Gestión del proyecto | Juan Carlos Ghiringhelli |
| Gestión de la calidad | Javier Madeiro |

* 1. Actividades
     1. Ciclo de vida del software cubierto por el Plan

El ciclo de vida del software se conforma de sucesivas fases secuenciales las cuales concluyen por un hito en el que se toman decisiones críticas, en cada fin de fase, se realiza una evaluación o valoración para determinar si se han cumplido los objetivos planteados para la fase. En este sentido, se entiende que una evaluación satisfactoria permite avanzar a la próxima fase del proyecto.

En etapas tempranas, se presentará al usuario prototipos del producto a fin de evitar riesgos en el ámbito de la calidad (satisfacción del cliente) así como también para validar los requerimientos analizados.

Como ya se ha mencionado anteriormente, el alcance de este plan cubre las siguientes etapas del ciclo de vida del software:

* Etapa de requerimientos y análisis.
* Etapa de diseño.
* Etapa de implementación.
* Etapa de verificación.

Todos los entregables de este proyecto serán revisados por el responsable y asistente de SQA.

* + 1. Actividades de calidad a realizarse

Las tareas deberán reflejar las evaluaciones a realizar, los estándares a seguir, los productos a revisar, los procedimientos a seguir en la elaboración de los distintos productos así como también, 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.
  + 1. 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 la siguiente 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, el mismo debe ser distribuido a los responsables del producto y se debe asegurar que son conscientes de desviaciones o discrepancias encontradas.

* + 1. Revisar el ajuste al proceso

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

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

Esta información se obtiene de los siguientes documentos:

* Plan del Proyecto.
* Plan de la Iteración.
* Plan de Verificación.

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

Como salida se obtiene el Informe de revisión de SQA correspondiente a la evaluación de ajuste al proceso, este informe debe ser distribuido a los responsables de las actividades y se debe asegurar que los mismos son conscientes de las desviaciones o discrepancias encontradas.

* + 1. Realizar Revisión Técnica Formal (RTF)

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

En este sentido, es notorio que es un proceso de revisión riguroso, su objetivo es detectar, 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 formalmente a la reunión 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.

* + 1. 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.

* + 1. Relaciones entre las actividades de SQA y la planificación

En esta sección se incluye una lista con las actividades de calidad a realizarse durante el proyecto, especificando en que semana del proyecto se realizarán. Se debe de tener en cuenta que esta planificación es estimada, por lo que la misma se irá ajustando conforme avance el proyecto.

|  |  |
| --- | --- |
| **Actividad** | **Semana cuando se realiza** |
| Identificar las propiedades de calidad | 1,2,3,4 |
| Plan de calidad | 2,4 |
| Evaluar y ajustar el plan de SQA | 9,11 |
| Revisión técnica formal | 5, 7, 9,11 |
| Revisar las entregas | 1,2,3,4,5,6,7,8,9,10,11,12,13,14 |
| Revisar el ajuste al proceso | 3, 5, 7, 9, 11, 13, 14 |
| Evaluar la calidad de los productos | 3, 5, 7, 9, 11, 13, 14 |
| Realizar el informe final de calidad | 14 |

* 1. Responsables

Las actividades correspondientes al aseguramiento de la calidad, involucran principalmente al responsable de SQA y al asistente de SQA, en particular en las revisiones técnicas formales, se participa al responsable del área a revisar (punto 3.1 de este documento).

Documentación

* 1. Propósito

Identificación de la documentación relativa al 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.

* 1. Documentación mínima requerida

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

Consideramos que la documentación mínima debe conformarse con los siguientes documentos:

* Especificación de requerimientos del software.
* Modelo de casos de uso.
* Pautas para la interfaz de usuario.
* Descripción y diseño del software (Arquitectura).
* Plan de calidad.
* Plan de verificación y validación.
* Plan de configuración.
* Plan de proyecto.
  + 1. 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 de su realización, en los que se cubren los aspectos que se haya acordado a detallar con el cliente.

La especificación debe:

* Ser completa:

1. Externa, respecto al alcance acordado.
2. Internamente, no deben existir elementos sin especificar.

* Ser consistente, no pueden haber elementos contradictorios.
* No ser 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’ y se detallan a continuación:

Funcionalidad

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

Fiabilidad

* Madurez.
* Tolerancia a faltas.
* Recuperabilidad.

Usabilidad

* Comprensible.
* Aprendible.
* Operable.
* Atractivo.

Eficiencia

* Comportamiento respecto al tiempo.
* Utilización de recursos.

Mantenibilidad

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

Portabilidad

* Adaptable.
* Instalable.
* Co-existencia.

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

Con el fin de satisfacer las necesidades específicas expuestas por el usuario, volcaremos nuestros esfuerzos en el cumplimiento de los siguientes atributos de calidad:

* Facilidad de uso (Usabilidad): El sistema a desarrollar tendrá un bajo costo de aprendizaje y compresión para un usuario inexperto. Interpool deberá poder definirse como un juego intuitivo, entretenido y con una interfaz gráfica sumamente atractiva para el usuario.
* Eficiencia: Para cumplir con los objetivos de jugabilidad es fundamental que la performance sea apropiada para las condiciones establecidas, entendemos así que el tiempo de respuesta y el adecuado uso de los recursos serán factores importantes para la dinámica del juego.
* Funcionalidad: El sistema deberá satisfacer las necesidades implícitas y establecidas, será idóneo para el usuario y tendrá como punto clave la seguridad de la información y datos de los usuarios.
  + 1. 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 hayan acordado 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:

1. Todo elemento del diseño debe contribuir a algún requerimiento.
2. 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.
  + 1. Plan de Verificación & Validación

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

* La verificación de que:

1. 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.
2. Los requerimientos descritos en el documento de requerimientos, son implementados en el diseño expresado en el documento de diseño.
3. El diseño expresado en el documento de diseño se encuentra implementado en código.

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

* + 1. Reportes de Verificación & Validación

Estos documentos deben especificar los resultados de la ejecución de los procesos descritos en el Plan de V & V.

* + 1. Documentación de usuario

Debido al alto grado de usabilidad que deberá presentar el producto, se puede decir que la documentación de usuario en este proyecto será un punto intrascendente. Por este motivo, se ha acordado con el cliente que obtendrá como resultado del proyecto una documentación para el usuario básica y concisa. La forma de presentación de esta documentación será en formato web.

* + 1. Plan de Gestión de configuración

El Plan de gestión de configuración debe contener métodos para identificar componentes de software, control e implementación de cambios, registro y reporte del estado de los cambios implementados.

* + 1. Plan de Proyecto

El objetivo de la monitorización y control de proyectos es proporcionar una compresión del estado del proyecto, con el fin de implementar acciones correctivas cuando la ejecución del mismo se desvíe del plan.

El documento del plan de proyecto es la base para monitorizar las actividades, comunicar el estado y tomar acciones correctivas. El progreso se determina comparando los actuales elementos de trabajo: tareas, horas realizadas, coste y calendario actual, con los estimados en el plan de proyecto. Una apropiada visibilidad nos permitirá tomar acciones correctivas antes de que el trabajo real se desvíe del plan.

Estas acciones que tomaremos, harán que tengamos que rehacer/ajustar nuestro plan de proyectos.

Estándares, prácticas, convenciones y métricas

En esta sección del plan de SQA se identifican estándares, prácticas, convenciones y métricas que serán aplicadas para la evaluación de calidad de este proyecto. También se indicará como será monitoreado y asegurado el cumplimiento con estos elementos.

* 1. Estándar de documentación

Como estándares de documentación se definirán dos documentos:

* Estándar de documentación técnica y
* 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.

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.
* Historia de revisiones.
* Índice de contenido.

La documentación debe reflejar con los siguientes atributos de calidad:

* Legibilidad:
  + 1. Estructura.
    2. Tamaño.
    3. Ilustraciones.
    4. Facilidad para ubicar información relevante.
* Completo.
* Correcto.

El responsable y el asistente de SQA velaran por el cumplimiento de estos estándares y reportarán los eventuales defectos a los correspondientes responsables.

En el apéndice A de este documento se detallan con una mayor granularidad los estándares para la documentación.

* 1. 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.

* 1. Otros Estándares
     1. Estándares de programación

Se utilizan las técnicas de codificación propuestas por Microsoft para el desarrollo de software, complementariamente se utiliza el utilitario FxCop el cual analiza el assembly verificando si las buenas prácticas descriptas en las guidelines del .NET Framework fueron aplicadas, lo cual nos permite realizar un código más robusto y de mayor calidad, reduciendo así los riesgos.

Revisiones y auditorías

* 1. 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.

* 1. Requerimientos mínimos

En esta sección se especifican las revisiones y auditorías que deben realizarse como mínimo, así como la agenda para la realización de las mismas.

* + 1. Revisión de requerimientos

Esta revisión se realiza para asegurar que se cumplió con los requerimientos especificados por el Cliente.

* + 1. Revisión de diseño preliminar

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

* + 1. Revisión de diseño crítico

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

* + 1. Revisión del Plan de Verificación & Validación

Esta revisión se realiza para asegurar la consistencia y completitud de los métodos especificados en el Plan de V & V.

* + 1. 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.

* + 1. Auditoría física

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

* + 1. Auditorías internas al proceso

Estas auditorías son 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.

* + 1. 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 (en caso de que sea posible).

* + 1. 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.

* + 1. 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.

* + 1. Agenda

Se pasa a detallar una agenda tentativa, en la cual se especifican las semanas en que se realizarán las revisiones y auditorias; cabe aclarar que la misma se encuentra sujeta a la disponibilidad de los recursos humanos y las prioridades del equipo.

|  |  |
| --- | --- |
| **Revisión** | **Semanas en las que se realizan** |
| Revisión de Requerimientos | 3, 4, 5, 6, 7 |
| Revisión de Diseño preliminar | 4, 5, 6 |
| Revisión de Diseño Crítico | 7, 8, 9, 10 |
| Revisión de Plan de V&V | 5, 7, 10 |
| Auditoria funcional | 12 |
| Auditoria física | 12 |
| Auditorías internas al proceso | 4, 6, 8, 10, 12 |
| Revisión de gestión | 4, 6, 8, 10, 12 |
| Revisión del plan de gestión | 4, 6, 8, 10, 12 |
| Revisión Post Mortem | 14 |

* 1. Otras revisiones
     1. Revisión de documentación de usuario

Se revisa la completitud, claridad, correctitud y aplicación de uso.

Verificación

Todo lo relativo a las verificaciones y métodos a utilizar han sido incluidos en el Plan de Verificación & Validación, por dicho motivo este tema no se detallara en este documento.

Reporte de problemas y acciones correctivas

Como se detallo en el punto 3.1 de este documento, hemos definido un responsable por cada línea de trabajo, el mismo será el encargado de hacer llegar al responsable de SQA los entregables de la semana.

El responsable y el asistente de SQA revisarán cada uno de los productos, y en caso de detectar potenciales defectos se notificará sobre los mismos al debido responsable, quien evaluará con el resto de los integrantes de la línea de trabajo a fin de verificar si realmente es un error, de lo contrario se descarta la observación.

A su vez, todos los lunes se acordarán fechas límites para realizar las entregas de la semana al equipo de SQA, las mismas deberán contemplar un margen de tiempo para la revisión y para la corrección de los posibles defectos encontrados. Se decidió adoptar este sistema para no sobrecargar al equipo de SQA con los entregables sobre la hora de la entrega.

En caso de que una corrección no se finalice al momento de la entrega, se hará la correspondiente observación en el Informe semanal de SQA, y se le realizará un seguimiento durante la siguiente semana.

Herramientas, técnicas y metodologías

Se utilizará la herramienta FxCop con el fin de analizar la calidad del código fuente de los prototipos y del producto final.

Para realizar las distintas revisiones del producto, se utilizarán las CheckLists definidas en el Apéndice B.

Gestión de riesgos

Los riesgos identificados, la estrategia de mitigación, monitoreo y plan de contingencia a ser llevados a cabo, serán descritos en el Documento de Gestión de Riesgos, GPDRIG2vx.pdf.

Apéndices

1. Documentación

Para documentar se usará la suite de Microsoft Office 2007.

Los documentos de texto se guardarán con extensión DOCX (formato por defecto de Microsoft Word 2007). El responsable de SQA pasará estos documentos a PDF para su correspondiente entrega.

Las planillas de cálculo se guardarán y se entregarán con extensión XLS.

Se utiliza como base el formato definido en las plantillas:

* Estilo normal: Fuente Verdana 10.
* Titulo primer nivel: Fuente Verdana tamaño 11, negrita.
* Titulo segundo nivel: Fuente Verdana tamaño 10, negrita.
* Titulo tercer nivel: Fuente Verdana tamaño 10, negrita.
* Titulo cuarto nivel: Fuente Verdana tamaño 10, cursiva.
* Índice: Times New Roman tamaño 10.

Los nombres de los archivos serán con letra mayúscula y deberán respetar el formato prefijado, solamente se le deberá sustituir la X por el número de grupo (2) y la Y por el número de versión (por ejemplo 1\_0).

Utilizar la viñeta: Punto redondo o números según corresponda.

Los índices deberán estar sincronizados y actualizados.

1. CheckLists
2. Revisión de Requerimientos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **Pregunta** | **Si** | **No** | **NA** |
| 1 | ¿Se utiliza un lenguaje conocido por el cliente? |  |  |  |
| 2 | ¿Se utiliza un lenguaje conocido por los desarrolladores? |  |  |  |
| 3 | ¿Cada requerimiento está definido una única vez? (no hay requerimientos redundantes) |  |  |  |
| 4 | ¿Los requerimientos están especificados con un nivel apropiado de detalle? (ni más, ni menos) |  |  |  |
| 5 | ¿Los requerimientos funcionales cubren todas las situaciones anormales? (ejemplo: caída del sistema) |  |  |  |
| 6 | ¿El documento contiene sólo detalles de implementación necesarios? |  |  |  |
| 7 | ¿Cada característica del producto final es descripta con una única terminología? |  |  |  |
| 8 | ¿Se corresponde con las necesidades explicitadas en los documentos referentes a las reuniones de requerimientos que se mantuvieron con el cliente? |  |  |  |
| 9 | ¿Hay un índice? |  |  |  |
| 10 | ¿Este documento ha sido validado por el cliente? |  |  |  |
| 11 | ¿Los requerimientos están priorizados? |  |  |  |
| 12 | ¿Todas las unidades de medida están definidas? |  |  |  |
| 13 | ¿Esta definido perfectamente el alcance del sistema? |  |  |  |
| 14 | ¿Algún requerimiento debería estar especificado con más detalle? |  |  |  |
| 15 | ¿Algún requerimiento debería estar especificado con menos detalle? |  |  |  |
| 16 | ¿Se ha definido qué información falta si es que falta alguna? |  |  |  |
| 17 | ¿Están descriptas claramente las principales interfaces externas e internas del producto? |  |  |  |
| 18 | ¿Están incluidos los requerimientos de calidad? (como ser robustez, confiabilidad) |  |  |  |
| 19 | ¿Están incluidos todos los requerimientos relacionados con las restricciones de Diseño? |  |  |  |
| 20 | ¿Están incluidos todos los requerimientos relacionados con el hardware? |  |  |  |
| 21 | ¿Están incluidos todos los requerimientos relacionados con entradas? |  |  |  |
| 22 | ¿Están incluidos todos los requerimientos relacionados con salidas? |  |  |  |
| 23 | ¿La estructura del documento se adhiere a un estándar aceptado? |  |  |  |
| 24 | ¿El documento de requerimientos está organizado clara y lógicamente? |  |  |  |
| 25 | ¿El documento de requerimientos está libre de contradicciones? |  |  |  |
| 26 | ¿Cada requerimiento es relevante al problema y a su solución? |  |  |  |
| 27 | ¿Hay algún requerimiento que pueda tener más de una interpretación? |  |  |  |
| 28 | ¿Hay un glosario en el cual el significado específico de cada término está definido? |  |  |  |
| 29 | ¿Hay algún requerimiento que es imposible de verificar? |  |  |  |
| 30 | ¿Cada requerimiento puede ser trazado hacia el documento donde se describe el problema o hacia una necesidad específica del cliente? |  |  |  |
| 31 | ¿Todos los requerimientos son trazables hacia un documento específico de diseño? |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| **Nº** | **Pregunta** | **Si** | **No** | **NA** |
| 32 | ¿Todos los requerimientos son trazables hacia un módulo específico? |  |  |  |
| 33 | ¿Algunos de los requerimientos definidos son en realidad detalles de diseño? |  |  |  |
| 34 | ¿Algunos de los requerimientos definidos son en realidad detalles de verificación? |  |  |  |
| 35 | ¿Algunos de los requerimientos definidos son en realidad detalles de gestión del proyecto? |  |  |  |
| 36 | ¿Está definida la plataforma de software requerida? |  |  |  |
| 37 | ¿Están definidas las herramientas de software requeridas? |  |  |  |
| 38 | ¿Los requerimientos de desempeño mínimos de la red están especificados? |  |  |  |

1. Revisión del Modelo de Casos de Uso

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **Pregunta** | **Si** | **No** | **NA** |
| 1 | ¿Todos los actores del modelo son exactamente los que se desprenden del Documento de Requerimientos? |  |  |  |
| 2 | ¿Los actores tienen los nombres intuitivos y descriptivos? |  |  |  |
| 3 | ¿Todos los actores están claramente definidos y son consistentes con el Documento de Requerimientos? |  |  |  |
| 4 | ¿Se puede ver claramente desde el diagrama de casos de uso y sus descripciones qué actores están involucrados en cada caso de uso? |  |  |  |
| 5 | ¿Todos los actores están conectados con los casos de uso correctos de acuerdo al Documento de Requerimientos? |  |  |  |
| 6 | ¿Todos los casos de uso del modelo son exactamente los que se desprenden del Documento de Requerimientos? |  |  |  |
| 7 | ¿Todos los casos de uso llevan a cumplir un sólo objetivo claramente definido? |  |  |  |
| 8 | ¿Todos los casos de uso tienen nombres que trasmiten claramente cuál es su objetivo? |  |  |  |
| 9 | ¿Todas las interacciones de los actores con el sistema son consistentes con las descripciones de los actores? |  |  |  |
| 10 | ¿Todas las descripciones de los casos de uso son consistentes con el Documento de Requerimientos? |  |  |  |
| 11 | ¿Todas las entradas y salidas están correctamente definidas para cada caso de uso? |  |  |  |
| 12 | ¿Todos los flujos alternativos han sido cubiertos? |  |  |  |
| 13 | ¿Todas las pre y postcondiciones para cada caso de uso están especificadas? |  |  |  |
| 14 | ¿Todos los diagramas de los casos de uso concuerdan con las descripciones de los mismos? |  |  |  |
| 15 | ¿Todos los casos de uso están escritos como casos de uso esenciales? |  |  |  |
| 16 | ¿Todos los casos de uso están libres de detalles de implementación? |  |  |  |

1. Revisión de las Pautas para la Interfaz de Usuario

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **Pregunta** | **Si** | **No** | **NA** |
| 1 | ¿Definió los formatos de los distintos tipos de pantallas? |  |  |  |
| 2 | ¿Definió los controles y elementos de diseño de interfaz de pantalla sus características, su disposición, y cómo se gestionan los eventos relacionados con ellos? |  |  |  |
| 3 | ¿Indicó la forma de navegar dentro de la pantalla y entre ellas? |  |  |  |
| 4 | ¿Definió los distintos tipos de formatos de impresión que se utilizarán en el sistema? |  |  |  |

1. Revisión de la Interfaz de Usuario

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **Pregunta** | **Si** | **No** | **NA** |
| 1 | ¿La interacción es intuitiva de usar? |  |  |  |
| 2 | ¿Es fácil de aprender? |  |  |  |
| 3 | ¿Es correcta la rapidez de respuesta a las acciones del usuario? |  |  |  |
| 4 | ¿Se usó el color de forma efectiva? |  |  |  |
| 5 | ¿El usuario recibe un correcto feedback de la aplicación? |  |  |  |
| 6 | ¿Se usó el color de forma efectiva? |  |  |  |

1. Revisión de la Arquitectura

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **Pregunta** | **Si** | **No** | **NA** |
| 1 | ¿La arquitectura esta compuesta de capas? |  |  |  |
| 2 | ¿Los paquetes están bien definidos, incluyen su funcionalidad e interfaz para otros módulos? |  |  |  |
| 3 | ¿Todas las estructuras de datos importantes están descriptas y justificadas? |  |  |  |
| 4 | ¿Todas las estructuras de datos importantes están ocultas mediante funciones de acceso? |  |  |  |
| 5 | ¿La organización y contenido de la base de datos están especificados? |  |  |  |
| 6 | ¿Todos los objetos importantes están descriptos y justificados? |  |  |  |
| 7 | ¿La interfaz con el usuario está modularizada tal que los cambios en ella no afecten el resto del programa? |  |  |  |
| 8 | ¿Los aspectos claves de la interfaz de usuario están definidos? |  |  |  |
| 9 | ¿Se ha estimado el uso de memoria, se ha descripto y justificado una estrategia para la gestión de la memoria? |  |  |  |
| 10 | ¿Se ha incluido una estrategia coherente de manejo de errores? |  |  |  |
| 11 | ¿Se ha definido un nivel de robustez? |  |  |  |
| 12 | ¿La arquitectura esta diseñada para acomodar cambios? |  |  |  |
| 13 | ¿Los objetivos principales del sistema están claramente establecidos? |  |  |  |
| 14 | ¿El diseño de alto nivel es independiente de la máquina y lenguaje que se usará para implementarlo? |  |  |  |
| 15 | ¿Se dan motivaciones para todas las decisiones importantes? |  |  |  |

1. Revisión de Descripción la Arquitectura

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **Pregunta** | **Si** | **No** | **NA** |
| 1 | ¿Identificó los casos de uso y actores más significantes para definir la Arquitectura del sistema (tomando en cuenta los más importantes para el cliente y que mitiguen los riesgos más serios)? |  |  |  |
| 2 | ¿Se han considerado varios estilos arquitectónicos diferentes antes de la definición de la arquitectura resultante? |  |  |  |
| 3 | ¿Realizó la trazabilidad entre el Modelo de Casos de Uso y el Modelo de Diseño, identificando los objetos y subsistemas de diseño que intervienen en el caso de uso y sus relaciones (Diagrama de paquetes)? |  |  |  |
| 4 | ¿Se especifican en el documento las correspondencias entre los requerimientos y el Modelo de Diseño? |  |  |  |
| 5 | ¿La arquitectura seleccionada ha sido ejercitada en escenarios reales? |  |  |  |
| 6 | ¿Realizó un diagrama mostrando la descomposición del Modelo de Diseño en Subsistemas, sus interfaces y las dependencias entre ellos, mostrando para cada Subsistema la Capa de la Arquitectura en la que se encuentra? |  |  |  |
| 7 | ¿Realizó una breve descripción del Subsistema, incluyendo un diagrama conteniendo las clases y los subsistemas que lo componen y para cada uno de ellos una breve descripción? |  |  |  |
| 8 | ¿Identificó las Clases de Diseño claves del Sistema? |  |  |  |
| 9 | ¿Describió las clases del Diseño y los subsistemas que corresponden a la Arquitectura del Diseño de cada Caso de Uso? |  |  |  |
| 10 | ¿Realizó la trazabilidad entre las entidades de diseño del Modelo de Diseño y los componentes del Modelo de Implementación? |  |  |  |
| 11 | ¿Indicó para cada entidad de diseño qué elemento en el ambiente de implementación le corresponde, por ejemplo para cada subsistema el módulo que lo implementa; para una funcionalidad importante cuál es el conjunto de objetos que la implementan, a qué módulo pertenece? |  |  |  |
| 12 | ¿Especificó cómo están organizados los componentes de acuerdo a la estructura y mecanismos de modularización disponibles en el ambiente de implementación y en el(los) lenguaje(s) de programación, y cuáles son las dependencias entre los distintos componentes? |  |  |  |
| 13 | ¿Muestra la correspondencia entre la Arquitectura del Software y la Arquitectura del Hardware? |  |  |  |
| 14 | ¿Se tienen en cuenta todas las propiedades de calidad que debe tener el sistema? |  |  |  |
| 15 | ¿Se ha alcanzado un grado adecuado de modularidad? |  |  |  |
| 16 | ¿Se ha diseñado para el cambio? |  |  |  |

1. Revisión del Modelo de Diseño de los Casos de Uso

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **Pregunta** | **Si** | **No** | **NA** |
| 1 | ¿Especificó los casos de uso, mediante subsistemas u objetos de diseño que interactúan, y determinó las operaciones e interfaces de los distintos subsistemas u objetos de diseño? |  |  |  |
| 2 | ¿El modelo de diseño permite cumplir con todos los requerimientos? |  |  |  |
| 3 | ¿Para cada objeto se definió parámetros, reglas, operaciones que realiza, requerimientos de implementación, interfaz de usuario y relaciones con otros objetos? |  |  |  |
| 4 | ¿Esta especificación se realiza mediante Diagrama de Paquetes, Diseño de flujo de eventos, Diagramas de interacción y Requerimientos especiales o de implementación? |  |  |  |
| 5 | ¿Identificó los objetos y subsistemas de diseño que intervienen en el caso de uso y sus relaciones? |  |  |  |
| 6 | ¿Realizó un diagrama de interacción para representar la realización de cada Caso de Uso? |  |  |  |
| 7 | ¿Se cuenta con una descripción escrita en términos de objetos y subsistemas de diseño, que explica y complementa el diagrama de interacción y sus niveles? |  |  |  |
| 8 | ¿Describió los requerimientos no funcionales en la realización de un caso de uso? |  |  |  |
| 9 | ¿Para cada subsistema específico: declaro que hace, cual es el propósito del mismo y cuáles son sus entradas y que salidas produce? |  |  |  |
| 10 | ¿Para cada subsistema identificó los subsistemas subordinados al mismo? |  |  |  |
| 11 | ¿Se identificó para cada subsistema la dependencia con otros subsistemas; indicando las condiciones para que se realice la iteración y las características de la iteración como ser pasaje de parámetros, mensajes, datos compartidos etc.? |  |  |  |
| 12 | ¿Identificó todos los recursos externos al diseño, que necesita el subsistema para realizar su función (dispositivos físicos, servicios de software, recursos de procesamiento etc.)? |  |  |  |
| 13 | ¿Realizó un diseño detallado del comportamiento de la interfaz de usuario a partir de la especificación de requerimientos y el mismo esta de acuerdo con el entorno tecnológico? |  |  |  |
| 14 | ¿Definió la estructura de datos que utilizará el sistema, a partir de los requisitos funcionales y no funcionales establecidos para el sistema y las particularidades del entorno tecnológico, que consiga una mayor eficiencia en el tratamiento de los datos? |  |  |  |
| 15 | ¿Para la creación del Modelo de datos identificó las principales entidades de datos y los datos por los que están compuestas a partir de la especificación de requerimientos y las dependencias entre dichas entidades? |  |  |  |
| 16 | ¿Especificó el modelo de distribución de datos indicando la ubicación de los manejadores de bases de datos o sistemas de archivos, así como los distintos elementos de la estructura física de datos (base de datos, tablas, índices)? |  |  |  |

1. Revisión para Diseño Orientado a Objetos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **Pregunta** | **Si** | **No** | **NA** |
| 1 | ¿Se han definido subsistemas como parte de la representación del diseño? |  |  |  |
| 2 | ¿Hay relativamente pocos subsistemas? |  |  |  |
| 3 | ¿Los límites de los subsistemas están bien definidos? |  |  |  |
| 4 | ¿Se ha establecido una bien definida interfaz para cada subsistema? |  |  |  |
| 5 | ¿Las clases que pueblan un subsistema tienen sentido dentro del contexto de los requerimientos funcionales/datos definidos para el subsistema? |  |  |  |
| 6 | ¿Se han establecido clases de comunicación para la comunicación entre subsistemas? |  |  |  |
| 7 | ¿Los subsistemas están diseñados para tener una interacción mínima con los demás? |  |  |  |
| 8 | ¿Los subsistemas están diseñados de tal forma que puedan ser usados en otros sistemas? |  |  |  |
| 9 | ¿Se han definido clases de gestión de tareas? |  |  |  |
| 10 | ¿Se han definido clases de gestión de datos? |  |  |  |
| 11 | ¿Se han definido clases de interfaz de usuario? |  |  |  |
| 12 | ¿El diseño diferencia entre los componentes del dominio del problema, de la interfaz de usuario, de gestión de tareas, y de gestión de datos? |  |  |  |
| 13 | ¿Toda la comunicación dentro del subsistema está claramente detallada en un diagrama de colaboración? |  |  |  |
| 14 | ¿Se utiliza un protocolo de definición para cada objeto? |  |  |  |
| 15 | ¿El protocolo de descripción describe claramente la estructura de los mensajes para cada clase/objeto? |  |  |  |
| 16 | ¿Se han utilizado patrones de diseño reconocidos como parte del diseño? |  |  |  |
| 17 | ¿El programa será fácil de mantener? |  |  |  |
| 18 | ¿El diseño tiene en cuenta futuras extensiones al programa? |  |  |  |
| 19 | ¿Todas las partes del diseño son estrictamente necesarias? |  |  |  |

1. Revisión de la Gestión de Configuración del Software

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N°** | **Pregunta** | **Sí** | **No** | **NA** |
| 1 | ¿Todos los integrantes del grupo están siguiendo los procedimientos definidos en el plan SCM? |  |  |  |
| 2 | ¿Está definida la línea base de la fase / iteración? |  |  |  |
| 3 | ¿Todos los ítems que indica el documento de la línea base están en el directorio correspondiente a esa la línea base? |  |  |  |
| 4 | ¿Coinciden las versiones de cada ítem? |  |  |  |
| 5 | ¿Todos los ítems dentro del directorio de la línea base se encuentran en el documento correspondiente? |  |  |  |

1. Revisión del Plan de Gestión de Configuración del Software

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N°** | **Pregunta** | **Sí** | **No** | **NA** |
| 1 | ¿Los productos de software a controlar están identificados claramente en el plan? |  |  |  |
| 2 | ¿Existe una regla para nombrar a cada producto de software? |  |  |  |
| 3 | ¿Se especifica cuándo se crea una nueva línea base? |  |  |  |
| 4 | ¿Se especifica qué productos son incluidos en la nueva línea base? |  |  |  |
| 5 | ¿Se especifica quienes son responsables por los productos incluidos en la línea base? |  |  |  |
| 6 | ¿Existe una regla para identificar las líneas base (versión)? |  |  |  |
| 7 | ¿Existe un procedimiento de control de cambios bien definido (qué debe hacer un integrante del grupo cuando quiere modificar cualquier producto presente en la línea base)? |  |  |  |
| 8 | ¿Se especifica en el plan si se reportará el estado de implementación de los cambios permitidos? |  |  |  |
| 9 | ¿El plan asigna a cada actividad SCM un responsable? |  |  |  |
| 10 | ¿El plan describe qué herramientas son utilizadas para implementar las actividades SCM? |  |  |  |

1. Revisión de la Implementación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N°** | **Pregunta** | **Sí** | **No** | **NA** |
| 1 | Conociendo la nueva reestructuración de grupos y funcionamiento dentro de ellos, ¿cree que le es más fácil a los implementadores lograr implementar habiendo participado en el diseño? ¿Han tenido algún problema de no entendimiento del diseño? ¿Se respeta el diseño a la hora de implementar, o se le hacen modificaciones sobre el avance? |  |  |  |
| 2 | ¿Realizan integración? |  |  |  |
| 3 | ¿Realizan algún tipo de verificación? |  |  |  |
| 4 | ¿La documentación técnica la hacen a medida que implementan o luego? ¿La realiza el implementador que construyó la clase o se encarga otra persona? |  |  |  |
| 5 | ¿Han tenido algún problema en seguir las normas de implementación? |  |  |  |
| 6 | ¿Realizan mediciones sobre la cantidad de líneas de código y tiempo de implementación? |  |  |  |
| 7 | ¿Implementan de a pares? ¿Realizan revisión por pares? ¿Han obtenido algún buen resultado de esto último? |  |  |  |
| 8 | ¿Hasta cuándo se tiene pensado implementar? ¿La implementación cubre el alcance previsto para la fase? |  |  |  |
| 9 | ¿Existe un estándar de implementación? |  |  |  |
| 10 | ¿El estándar de implementación fue aprobado por todos? |  |  |  |
| 11 | ¿Existe un estándar de documentación técnica? |  |  |  |
| 12 | ¿El estándar de documentación técnica fue aprobado por todos? |  |  |  |
| 13 | ¿Se identificaron todas las herramientas a utilizar? |  |  |  |
| 14 | ¿Todos los integrantes del grupo poseen las herramientas que deben utilizar? |  |  |  |
| 15 | ¿Todos los integrantes del grupo tienen un mínimo conocimiento acerca de las herramientas que deben utilizar? |  |  |  |
| 16 | ¿Se midió el esfuerzo y tamaño de lo implementado (en LOCs)? |  |  |  |

1. Revisión del Plan de Proyecto

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N°** | **Pregunta** | **Sí** | **No** | **NA** |
| 1 | ¿Las dependencias entre las actividades están claramente identificadas? |  |  |  |
| 2 | ¿Se ha calculado el camino crítico del proyecto? |  |  |  |
| 3 | ¿El camino crítico del proyecto es revisado regularmente en busca de cambios? |  |  |  |
| 4 | ¿Se han realizado planes de contingencia para manejar retrasos de actividades específicas? |  |  |  |
| 5 | ¿Se utiliza el análisis de valor ganado para calcular el progreso? |  |  |  |
| 6 | ¿Se han definido mecanismos para rastrear los errores durante el proyecto? |  |  |  |

1. Revisión de la Estimaciones y Mediciones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N°** | **Pregunta** | **Sí** | **No** | **NA** |
| 1 | ¿Se usó algún mecanismo formal (ej: casos de uso) para obtener los requerimientos básicos antes de realizar la estimación? |  |  |  |
| 2 | ¿Se estimó el tamaño del producto final y sus alcances intermedios? |  |  |  |
| 3 | ¿La estimación anterior se realizó utilizando varias técnicas? |  |  |  |
| 4 | ¿Las estimaciones de tamaño fueron generadas por más de una persona? |  |  |  |
| 5 | ¿Las personas que realizan la estimación tienen experiencia en el dominio de aplicación? |  |  |  |
| 6 | ¿La descomposición en funciones se utilizó para generar las estimaciones de tamaño? |  |  |  |
| 7 | ¿El alcance del proyecto y avance por fases es consistente con las estimaciones? |  |  |  |
| 8 | ¿Se realizó un análisis de factibilidad para las funciones más importantes? |  |  |  |
| 9 | ¿Se consideró el impacto de componentes reusables? |  |  |  |

1. Revisión de la Documento de Riesgos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N°** | **Pregunta** | **Sí** | **No** | **NA** |
| 1 | ¿Se identificaron riesgos en el proyecto? |  |  |  |
| 2 | ¿Los riesgos están priorizados? |  |  |  |
| 3 | ¿Se identificaron medidas para prevenirlos? |  |  |  |
| 4 | ¿Se identificaron medidas para minimizar su impacto? |  |  |  |
| 5 | ¿Se tomaron medidas para prevenir los riesgos más importantes? |  |  |  |
| 6 | ¿Se tomaron medidas para minimizar el impacto de los riesgos que ocurrieron? |  |  |  |

1. Revisión del Plan de Verificación y Validación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N°** | **Pregunta** | **Sí** | **No** | **NA** |
| 1 | ¿Todos los métodos de Verificación y Validación aseguran completitud y compatibilidad de los requerimientos funcionales y no funcionales? |  |  |  |
| 2 | ¿La configuración del Software a ser testeado, como así también los elementos de Software y Hardware de soporte están definidos y se adecuan a los requerimientos? |  |  |  |
| 3 | ¿Las pruebas están definidas, son completas y son consistentes con los requerimientos? |  |  |  |
| 4 | ¿Los casos y procedimientos de prueba tienen instrucciones definidas, claras y concisas? |  |  |  |
| 5 | ¿La agenda del Plan de Verificación y Validación está definida, indicando que prueba se harán a que, cuándo y por qué? |  |  |  |

1. Revisión de la Documentación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **Pregunta** | **Sí** | **No** | **NA** |
| 1 | ¿Se verificó que el documento siga el formato estandar? |  |  |  |
| 2 | ¿Todas las unidades de medida están definidas? |  |  |  |
| 3 | ¿Todas las figuras, tablas, y diagramas tiene referencias cruzadas? |  |  |  |
| 4 | ¿Están todas las figuras, tablas y diagramas necesarios? |  |  |  |
| 5 | ¿Todas las figuras, tablas y diagramas están rotulados? |  |  |  |
| 6 | ¿Hay un índice? |  |  |  |