



Universidad de las Ciencias
Informáticas

Facultad 6.

***Título: Estrategia de Aseguramiento de la Calidad para la
Factoría de Software Aplicativos SIG.***

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS
INFORMÁTICAS**

Autor

Yarisleydis Rivera de la Peña

Tutor

Ing. Dayanis Elvia Alcántara Rabí

La Habana, Junio de 2011.

“Año 53 de la Revolución”

Declaración de Autoría

Declaro ser el único autor del trabajo Estrategia de SQA para la Factoría de Software Aplicativos SIG y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los _____ días del mes de _____ del año_____.

Yarisleydis Rivera De la Peña
Firma del Autor

Ing. Dayanis Elvia Alcántara Rabí
Firma del Tutor

Dedicatoria

A mi mamita, por darme su apoyo en todo momento y ser la mejor amiga que he tenido...

A mi papito, por siempre complacerme y dar la vida por su niñita...

A mis hermanos Migdalia, Tania, Reinaldo y Yoandri, por querer tanto a su manita...

A mi vido Yaroby por estar siempre conmigo, por todas las cosas que ha hecho por mí, por su amor y dedicación...

A Ivy y Melixa, por preocuparse por mi y ayudarme siempre que lo necesité...

A María Herminia, por ser una guía para mí...

Por ser las personas más importantes en mi vida, les dedico mi sueño.

Agradecimientos

Sería difícil agradecer a todas las personas que de una forma u otra me ayudaron durante toda la carrera, hasta llegar aquí.

Muchas gracias a todas las personas que pensaron que yo no podía estar donde estoy, pues me obligaron a esforzarme y demostrarles que si puedo cumplir mis sueños.

A todos aquellos que durante mis cinco años de carrera fueron compañeros y amigos y siempre estuvieron conmigo en los momentos más difíciles.

Especialmente a mi mamá y mi papá por apoyarme en todas mis decisiones...

A mi novio Yaroby y a su mamá Noridia, por ayudarme siempre que los necesité...

A mis hermanos, por su confianza y apoyo, especialmente a Migdalia y Tania, por ser las mejores hermanas del mundo...

A mis vecinos por confiar en mí en todo momento...

A Pedro y Zoraida por ser como unos padres para mí...

A Delia, Charo, Mati, Yuya y Elena, por ser mis niñeras aún después de grande...

A Yailín, Mónica, Laritza, Mineydis, Yailín, Arletis, Ailín y Yander, por ser mis amigos de toda la vida...

A mis tios y primos por tenerme siempre presente...

Agradecimientos

A todos mis compañeros por brindarme su confianza, especialmente a Ivy, Melixa, Lulu, Eli, Sachie, Saight, Lisette, Jose Angel y Pedro...

A Deylis, Lucho, Yansel y Alejandro, que aunque hace muy poco tiempo los conozco, se han portado muy bien conmigo...

A todos los profesores que me ayudaron durante mis cinco años de carrera...

A mi tutora Dayanis y a mi presidenta de tribunal Yeleny por aguantar mis malacrianzas...

Y muy especialmente a mi Comandante en Jefe Fidel Castro Ruz...

A todos los que de una forma u otra me han ayudado, gracias por haber estado a mi lado. Gracias por hacerme pasar buenos momentos, por hacerme reír cuando lo necesitaba y por hacer estos cinco años de universidad más agradables.

Resumen

Uno de los principales objetivos de un equipo de desarrollo es obtener un producto con calidad. Para ello es necesaria la implementación de una estrategia de aseguramiento de la calidad que le permita al producto final cumplir con las expectativas y necesidades del cliente.

El presente trabajo propone para la Factoría de Software Aplicativos SIG una estrategia para lograr el aseguramiento de la calidad de sus productos. Durante la investigación se abordaron aspectos teóricos sobre el tema de la calidad de software, además de analizar cada una de las características necesarias de la factoría para la elaboración de la estrategia. En la misma se propone el uso de métricas, normas, revisiones, pruebas, y una serie de actividades que deben ser realizadas dentro de la factoría para la organización y el entrenamiento del equipo de desarrollo, todo esto basándose en las necesidades y objetivos de la factoría. Para determinar el éxito de la solución propuesta es necesario aplicar la estrategia a uno de los productos de la factoría.

Palabras Clave:

- ✓ Calidad.
- ✓ Aseguramiento de la Calidad
- ✓ Estrategia de Aseguramiento de la Calidad.
- ✓ Plan de Aseguramiento de la Calidad.
- ✓ Factoría de Software.

Índice

| | |
|--|----|
| Introducción | 1 |
| Capítulo I: Fundamentación Teórica del Aseguramiento de la Calidad del Software..... | 5 |
| 1.1. Introducción | 5 |
| 1.2. Calidad..... | 5 |
| 1.3. Calidad de Software | 6 |
| 1.4. Aseguramiento de la Calidad | 7 |
| 1.5. Modelos y Estándares Calidad | 8 |
| 1.5.1. CMMI | 9 |
| 1.5.2. ISO | 11 |
| 1.5.3. ISO y CMMI | 13 |
| 1.6. Fábricas o Factoría de Software | 14 |
| 1.7. Líneas de Productos de Software (LPS)..... | 17 |
| 1.8. Los Sistemas de Información Geográfica | 19 |
| 1.9. Factoría de Software Aplicativos SIG | 21 |
| 1.9.1. Proceso de Desarrollo de un producto en Aplicativos SIG | 23 |
| 1.10. Conclusiones Parciales..... | 25 |
| Capítulo 2: Estrategia de Aseguramiento de la Calidad para la Factoría de Software Aplicativos SIG . | 26 |
| 2.1 Introducción | 26 |
| 2.2 Estrategia de Aseguramiento de la Calidad para Aplicativos SIG | 26 |
| 2.1 Organizar el trabajo de los Recursos Humanos de la Factoría..... | 27 |
| 2.2 Entrenar los miembros de la factoría..... | 29 |
| 2.3 Plan de Aseguramiento de la Calidad | 32 |
| 2.3.1 Objetivos de Calidad | 32 |
| 2.3.2 Gestión del Aseguramiento de la Calidad..... | 33 |
| 2.3.3 Estándares y Guías | 35 |
| 2.3.4 Herramientas, Técnicas y Metodologías | 35 |
| 2.3.5 Registro de Calidad | 36 |
| 2.3.6 Gestión de la Configuración del Software..... | 36 |
| 2.3.7 Revisiones y Auditorías | 37 |

| | | |
|--|--|----|
| 2.3.8 | Pruebas | 39 |
| 2.3.9 | Métricas | 41 |
| 2.4 | Conclusiones | 44 |
| Capítulo 3: Validación de la Propuesta de Solución de la Estrategia de Aseguramiento de la Calidad para la Factoría de Software Aplicativos SIG..... | | 45 |
| 3.1 | Introducción | 45 |
| 3.2 | Estrategia de SQA..... | 45 |
| 3.2.1 | Organizar y Entrenar el trabajo de los Recursos Humanos..... | 46 |
| 3.2.2 | Actividades del PSQA..... | 49 |
| 3.2.2.1 | Revisiones..... | 49 |
| 3.2.2.2 | Pruebas | 50 |
| 3.2.2.3 | Métricas | 50 |
| 3.3 | Conclusiones | 52 |
| Conclusiones Generales..... | | 53 |
| Recomendaciones..... | | 54 |
| Bibliografía..... | | 55 |
| Anexos..... | | 58 |

Índice de Tablas

| | |
|---|----|
| Tabla 1: Comparación entre ISO y CMMI (Elaboración Propia)..... | 14 |
| Tabla 2. Grupos de Factorías de Aplicativos SIG (Elaboración Propia)..... | 25 |
| Tabla 3: Tareas y Responsabilidades de la Factoría Aplicativos SIG. (Elaboración Propia)..... | 29 |
| Tabla 4: Roles del SQA. (Elaboración Propia)..... | 34 |
| Tabla 5: Estándares y Guías. (Elaboración Propia)..... | 35 |
| Tabla 6: Organización de los Recursos Humanos. (Elaboración Propia)..... | 48 |

Índice de Figuras

| | |
|--|----|
| Figura 1: Relación de la Estrategia de SQA con el ciclo de vida de un producto en la Factoría..... | 27 |
| Figura 2: Actividades para la organización de los Recursos Humanos de la factoría..... | 27 |
| Figura 3: Actividades para el entrenamiento de los miembros de la factoría. | 30 |
| Figura 4: Actividades del PSQA para la factoría. | 32 |
| Figura 5: Porcentaje de cumplimiento de la Estrategia de SQA. (Elaboración Propia)..... | 45 |

Introducción

La eficacia que pueda tener la producción de software es un indicador de acierto imprescindible para su comercialización. En la actualidad las empresas productoras de software están enfocadas en contribuir al desarrollo y la excelencia de su producto, para incrementar su competitividad a través de la mejora continua y el conocimiento en tecnologías de la información, con la misión de satisfacer a todos los usuarios. Mantener la calidad como una característica imprescindible en un producto es de vital importancia, pues están fuertemente ligados a la vida diaria, siendo utilizados en la economía, la medicina y la educación.

En Cuba la Universidad de las Ciencias Informáticas (UCI) es uno de los centros dedicados al desarrollo de software. Trazándose como objetivo principal, mantener la perfección de sus productos, para obtener demanda en el mercado y hacer aportes a la economía del país. Con sus 9 años de creada, la UCI se ha convertido en una ciudad digital avanzada, basándose principalmente en la integración de los procesos de formación, investigación y producción. Esta integración se denomina Centros de Desarrollo, ubicados en las diferentes facultades, en la Facultad 6 están ubicados el de Geoinformática y Señales Digitales (GEYSED) y el Centro de Tecnologías y Gestión de Datos (DATEC).

La Factoría de Software Aplicativos de Sistema de Información Geográfico pertenece al departamento Geoinformática de GEYSED. La Factoría o Fábrica de Software se refiere al sentido de producir con rapidez y calidad a través de procesos conocidos, repetibles, gerenciales, y principalmente, mejorables continuamente, no solo por la incorporación de técnicas y herramientas en el desarrollo del software, sino porque se mantiene constantemente el foco sobre el mejoramiento del proceso de producción y cada uno de los pasos que esto acarrea. (Lazo Ochoa, y otros, 2010)

Esta factoría de software cuenta con la Plataforma Soberana para el desarrollo de Sistemas de Información Geográfica (GeneSIG), la que se utiliza para el desarrollo de otras aplicaciones SIG en entornos Web con tecnologías libres. Dicha factoría tiene definido para el proceso de

desarrollo de sus productos las Líneas de Productos de Software, para una reducción en el número promedio de defectos por productos. Aunque tiene un problema que influye en la aceptación de sus productos, y es que, debido a la forma en que está dividida la factoría para desarrollar los productos, no saben qué Pruebas aplicar al producto, ni en qué momento se realizarán, no tienen definida ninguna métrica para sus productos y no saben cuando solicitar la revisión para alguno de sus productos al Grupo de Calidad del Departamento Geoinformática. Todos estos factores hacen difícil controlar la calidad del producto al terminar en cada grupo. Por lo que es necesario planificar actividades que permitan darle solución a estos problemas.

Teniendo en cuenta la importancia que tienen los Sistemas de Información Geográfico (SIG) para esta fábrica, el **Problema a Resolver** quedaría expuesto de la siguiente manera: ¿Cómo contribuir que los productos de la Factoría de Software Aplicativos SIG satisfagan las necesidades del cliente?

Planteado el problema a resolver se establece como **Objeto de Estudio** de esta investigación el proceso de aseguramiento de la calidad del software y como **Campo de Acción** el proceso de aseguramiento de la calidad en la Factoría de Software Aplicativos SIG.

Para esto se plantea como **Objetivo General** definir una estrategia para el aseguramiento de la calidad en la Factoría de Software Aplicativos SIG.

Como **Idea a Defender** se plantea que, con la elaboración de una Estrategia de Aseguramiento de la Calidad para la Factoría de Software Aplicativos SIG se contribuirá a que los productos que se desarrollen en la misma satisfagan las necesidades del cliente.

Para dar cumplimiento al objetivo general de esta investigación se desarrollarán las diferentes **Tareas de Investigación:**

1. Sintetizar los modelos y estándares que existen internacionalmente para el proceso de aseguramiento de la calidad de un software.
2. Caracterizar las métricas que se utilizan en el proceso de evaluación del software.
3. Elaborar un Plan de Aseguramiento de la Calidad para la Factoría de Software Aplicativos SIG.

4. Confeccionar una propuesta de entrenamiento para el equipo de desarrollo de la Factoría de Software Aplicativos SIG.
5. Validar la estrategia propuesta, a través de su aplicación.

Se espera como principal **Aporte Práctico** del trabajo una Estrategia de Aseguramiento de la Calidad para la Factoría de Software Aplicativos SIG.

Para el desarrollo del trabajo se utilizarán los siguientes **Métodos Científicos**:

Los **Métodos Teóricos** permitirán analizar y explicar los resultados obtenidos, para poder llegar a buenas conclusiones y poder resolver el problema planteado, estos métodos son:

Histórico Lógico: Se utiliza con el objetivo de recopilar toda la información que se posee hasta el momento sobre las Estrategias de Aseguramiento de la Calidad y resumir los aspectos fundamentales necesarios para la investigación en curso.

Analítico Sintético: Este método se utiliza para un mejor estudio de la situación, permitiendo analizar todo lo relacionado con el Aseguramiento de la Calidad en el desarrollo de software.

Los **Métodos Empíricos** darán los argumentos necesarios para poder defender la idea, estos métodos son:

Entrevista: Se utiliza con el objetivo fundamental de obtener información que permita seleccionar la Estrategia de Aseguramiento de la Calidad más adecuada para la Factoría de Software.

Como esta investigación está encaminada a asegurar la calidad de los productos en Aplicativos SIG, la población está constituida por todos los miembros de la factoría y la muestra de la población, en este caso, es el Líder de Proyecto y los Jefes de Grupo de las factorías. Para la selección de la muestra de la población, se utilizó la técnica de muestreo intencional pues fueron seleccionados los miembros de la factoría que pueden brindar una mejor información sobre todo lo que sucede en la factoría.

Estructura del Trabajo

El presente trabajo está estructurado por tres capítulos:

Capítulo I: Fundamentación Teórica del Aseguramiento de la Calidad.

Se define el marco teórico y conceptual para la Estrategia de Aseguramiento de la Calidad, como son los diferentes conceptos asociados al término Calidad, se definen las normas, modelos y estándares de calidad existentes y sus mejores prácticas. Además de definir lo referido al marco relacionado con la Factoría de Software Aplicativos SIG.

Capítulo II: Estrategia de Aseguramiento de la Calidad para la Factoría de Software Aplicativos SIG.

Se describe la propuesta de solución, con un Plan de Aseguramiento de la Calidad, la organización de los Recursos Humanos de la factoría y la propuesta de entrenamiento para el equipo de desarrollo de dicha factoría.

Capítulo III: Validación de la Propuesta de Solución de la Estrategia de Aseguramiento de la Calidad para la Factoría de Software Aplicativos SIG.

En el capítulo se presenta la validación de la propuesta a partir de la aplicación de la misma y se analizan los resultados obtenidos.

Capítulo I: Fundamentación Teórica del Aseguramiento de la Calidad del Software

1.1. Introducción

Mantener la calidad durante todo el desarrollo de un producto de software, es uno de los principales objetivos que deben tener los centros dedicados a la producción de software. En la actualidad la poca calidad que tienen los productos es uno de los problemas más frecuentes, haciendo que los clientes pierdan toda la confianza en el producto. Por lo que se hace necesario que en toda empresa desarrolladora de software se tenga el conocimiento necesario para aplicar estrategias que permitan el aseguramiento de la calidad de sus productos.

Este capítulo se refiere a los diferentes conceptos y definiciones que permitirán un mejor entendimiento del trabajo. Se abordarán también, los modelos y estándares que existen y otros procedimientos que permiten el aseguramiento de la calidad en un proyecto de software.

1.2. Calidad

Según Joseph Jurán la calidad se “entiende como la ausencia de deficiencias que pueden presentarse como: retraso en las entregas, fallos durante los servicios, facturas incorrectas, cancelación de contratos de ventas, etc. Calidad es adecuarse al uso.” (Padilla, 2002).

Para Edwards Deming la calidad es la “Satisfacción a las demandas o exigencias del cliente (interno y externo), lo que configura uno de los pilares fundamentales, la actual cultura de calidad.” (Guillermo Domínguez, Luz Lozano, 2003).

Para la International Standards Organization (ISO 9000:2000) se define la calidad como “Grado en el que un conjunto de características inherentes cumple con los requisitos”. (Ana M. López, Cesar Cabrera, Luz E. Valencia Ayala, 2008).

Para Feigenbaum “La calidad es la más poderosa herramienta de negocio como agente de cambio.” (Guillermo Domínguez, Luz Lozano, 2003).

Luego de analizar las diferentes definiciones de estos autores, se puede afirmar que la calidad es la capacidad que pueda tener el producto de satisfacer todas las necesidades.

1.3. Calidad de Software

Para Philip Crosby la Calidad de Software es “la conformidad con los requisitos que la propia compañía ha establecido para sus productos basados directamente en las necesidades de sus clientes”. (Pello, 2010).

Según la ISO/IEC DEC 9126 se puede decir que el software tiene calidad si cumple o excede las expectativas del usuario en cuanto a (Vega Lebrúm, y otros, 2008):

1. Funcionalidad (que sirva para un propósito).
2. Ejecución (que sea práctico).
3. Confiabilidad (que haga lo que debe).
4. Disponibilidad (que funcione bajo cualquier circunstancia).
5. Apoyo, a un costo menor o igual al que el usuario está dispuesto a pagar.

Según Pressman la Calidad de Software es “Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente”. (Cueva Lovell, 1999).

Otro concepto plantea que “La Calidad del Software es el conjunto de cualidades que la caracterizan y que determinan su utilidad y existencia, plantea un adecuado balanceo de eficiencia, confiabilidad, facilidad de mantenimiento, portabilidad, facilidad de uso, seguridad e integridad.” (Scalone, 2006).

Por lo antes definido se puede asegurar que la calidad del software está presente cuando se desarrolla un producto que tenga la menor cantidad de errores y que cumpla con los requisitos establecidos por el cliente.

1.4. Aseguramiento de la Calidad

La calidad ha pasado por diferentes etapas de desarrollo hasta llegar al Aseguramiento de la Calidad o SQA (*Software Quality Assurance*). En estos momentos se puede decir que el “SQA es el conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza en que el producto (software) satisfará los requisitos dados de calidad.” (Cueva Lovell, 1999).

Para poder asegurar los buenos resultados de un producto, se deben tener presente varias actividades, como son:

1. Mediciones de software para el control del proyecto.
2. Verificación y validación del software a lo largo del ciclo de vida (Incluye las pruebas y los procesos de revisión e inspección).
3. La gestión de la configuración del software.
4. Aplicación de metodologías técnicas.
5. Realización de revisiones técnicas formales.
6. Ajuste a los estándares.
7. Control de cambios.
8. Registro y realización de informes.

Después de estar definidas estas actividades se deben conocer los métodos que permitirán la realización de estas con mayor facilidad:

1. Revisiones técnicas y de gestión (su objetivo es la evaluación).
2. Inspección (su objetivo es la verificación). ¿Se está construyendo el producto correcto?
3. Pruebas (su objetivo es la validación). ¿Se está construyendo el producto correctamente?
4. Auditorías (su objetivo es la confirmación del cumplimiento).

El SQA tiene como función principal determinar si las necesidades del usuario están siendo satisfechas, para esto se deben evaluar tres áreas:

- Objetivos: Estos están primero que los requerimientos del usuario y sus objetivos están en armonía con los de la organización.

- Métodos: Se deben utilizar métodos que observen o contengan las políticas, procedimientos y estándares de la organización.
- Ejecución: Optimización del uso del hardware y el software al implementar los productos de software. (Perry, 1983).

Para la Norma ISO 9000: 2000 el Aseguramiento de la Calidad es la parte de la gestión de calidad orientada a proporcionar confianza en que se cumplirán los requisitos de calidad. (Scalone, 2006).

Según Yoram Malevski el Aseguramiento de la Calidad “son todas aquellas acciones planificadas y sistemáticas necesarias para proporcionar la confianza adecuada de que un producto o servicio satisface los requisitos de calidad establecidos. Para que sea efectivo, el Aseguramiento de la Calidad requiere, generalmente, una evaluación permanente de aquellos factores que influyen en la adecuación del diseño y de las especificaciones según las aplicaciones previstas, así como también verificaciones y auditorías a las operaciones de producción, instalación e inspección. Dentro de una organización, el Aseguramiento de la Calidad sirve como una herramienta de la Gestión”. (Malevski, 1995).

Luego de analizar las definiciones anteriores se puede afirmar que el aplicar los procesos de Aseguramiento de la Calidad puede traer como resultados muchos y variados beneficios, como es la detección de problemas en tempranas etapas del desarrollo de productos de software, permitiendo al desarrollador corregirlos inmediatamente. También permite crear y seguir estándares de trabajo, los cuales apoyan a uniformizar y consolidar el proceso de desarrollo. Otro de estos beneficios es la posibilidad de planear la calidad, es decir, buscar la calidad, por lo que se debe de planear, construir e implantar en el producto.

1.5. Modelos y Estándares Calidad

Los Modelos y Estándares de Calidad son utilizados con el objetivo de desarrollar productos con calidad, permitiendo así una mejor comprensión de las características que tiene un producto de software determinado. El uso de estos modelos tiene muchas ventajas, pues evita

repetir errores que se hayan tenido anteriormente, además de convertirse en algo concreto, que se puede definir, que se puede medir y, sobre todo, que se puede planificar.

Aunque existen varios modelos y estándares de calidad, en este trabajo se profundizará más en ISO y la CMMI.

1.5.1. CMMI

Integración del Modelo de Capacidad y Madurez (*Capability Maturity Model Integration*) representa la fusión de un conjunto de modelos orientados a la mejora de procesos de ingeniería del software, ingeniería de sistemas, desarrollo de productos y adquisición de aplicaciones. Creado en 1991 por el Software Engineering Institute (SEI) como CMM y posteriormente actualizado como CMMI en 2002, está orientado a la garantía de calidad del software, y a la acreditación de empresas desarrolladoras de software en función del nivel de madurez de sus procesos de producción. (Asociación de Técnicos de Informática, 2009).

CMMI es un modelo de calidad del software que clasifica las empresas en niveles de madurez. Estos niveles sirven para conocer la madurez de los procesos que se realizan para producir software.

Enfoques de CMMI

Estos tienen como objetivo principal poder atender todas las necesidades que tenga una organización en cuanto a mejora de sus procesos. Estos enfoques son:

El Enfoque Continuo: Se basa principalmente en la capacidad que pueda tener un área determinada para realizar sus actividades de forma correcta.

Este utiliza niveles de capacidad para poder medir el mejoramiento de los procesos y se aplican a cada Área de Proceso (AP), estos niveles son:

0. Incompleto.
1. Ejecutado.
2. Gestionado.
3. Definido.
4. Cuantitativamente Gestionado.
5. Optimizado.

El Enfoque Escalonado: Hace hincapié en el grado de madurez de los procesos. Este utiliza niveles de madurez, aplicados a la madurez de la organización en su conjunto. Para esto existen cinco niveles de madurez:

1. Inicial.
2. Repetible.
3. Definido.
4. Cuantitativamente Gestionado.
5. Optimizado.

Estos niveles sirven para determinar el grado de madurez total que tiene una empresa y se van alcanzando cuando se ha puesto en práctica todas las AP aplicables a ese nivel y a los inferiores.

PPQA en el Nivel 2 de Madurez de CMMI

El Enfoque Escalonado en el Nivel 2 de Madurez tiene entre una de sus AP la de Aseguramiento de la Calidad en el Proceso y en el Producto (PPQA). Esta área tiene como objetivo principal planificar actividades que sean adecuadas para el aseguramiento del software, la verificación de la adhesión de los productos y actividades a estándares, procedimientos y requisitos aplicables y la información al equipo de trabajo. Se centra también en la medición de costes y calendario de ejecución del proyecto, proponiendo para esto diferentes medidas como:

- Cumplimiento de hitos para las actividades propias del aseguramiento del software.
- Trabajo realizado, esfuerzo utilizado e inversiones en actividades propias del aseguramiento del software.
- Cantidad de inspecciones del producto y revisiones de las actividades.

El proceso de Aseguramiento de la Calidad en el Nivel 2 de Madurez de CMMI proporciona personas y gestión para que los procesos y los elementos de trabajo cumplan los procesos. Esto puede hacerse mediante la identificación de los elementos de trabajo que componen una línea base y controlando los cambios en los elementos. También se pueden proporcionar las formas de construir los elementos de trabajo a partir del sistema de control de la configuración

y proporcionar información precisa de los datos de la configuración a desarrolladores y clientes. (García, 2005).

PPQA implica varias actividades, entre ellas, proporcionar retroalimentación al personal del proyecto y a los administradores sobre los resultados de las actividades de control de calidad; otra de estas actividades es evaluar objetivamente los procesos realizados, los productos de trabajo y los servicios en contra de la descripción de procesos de aplicación, normas y procedimientos. También apoya la entrega de productos de alta calidad, proporcionando al personal del proyecto una visibilidad adecuada sobre los procesos y productos de trabajo asociados a la vida del proyecto.

1.5.2. ISO

Las normas son reglas metodológicas y modelos adoptados por organizaciones nacionales o internacionales que se dedican al establecimiento de estándares. Estas organizaciones se apoyan en foros de expertos que se suelen constituir en comités muy cualificados. (Minguet Melián, y otros, 2003).

La Norma ISO (Organización Internacional de Normalización) propone varias normas en diferentes campos tecnológicos, aunque estas no deben ser aceptadas por la empresa u organización como una obligación para ellas, aunque son de gran beneficio, sería opcional usarlas. Una de las Normas establecida por la ISO es la 9000. Esta surgió en el año 1987 bajo la dirección de la ISO, con el objetivo de facilitar el comercio global, dándole valor añadido a las empresas certificadas bajo esta norma.

La Norma ISO 9000 es un sistema normas universales que definen la Garantía de Calidad, fue desarrollado por la Organización Internacional de Normalización (ISO) y adoptado por 90 países en todo el mundo. En realidad, las Normas ISO 9000 son una compilación universal de las mejores prácticas de calidad. (Sené, 1997).

La familia de normas denominada ISO 9000 es un conjunto de documentos sobre las buenas prácticas de los sistemas de gestión de la calidad. (Minguet Melián, y otros, 2003).

La Norma ISO 9000 no puede aplicarse completamente, ya que deben ser seleccionadas de acuerdo a la necesidad de la organización. Para este tipo de cuestiones están las demás normas que tienen que ver con la calidad y derivan de esta ISO:

- La Norma ISO 9001, que debe aplicarse en todo momento, pues esta es la que establece los requerimientos del sistema de calidad que se decida implantar en una empresa. También posibilidad diseñar y desarrollar, en este el producto que se quiera. Esta norma es gran importancia para la organización y por ello es imprescindible su aplicación.
- La Norma ISO 9002, se aplica cuando la empresa se hace cargo de la producción e instalación del producto.
- La Norma ISO 9000-3, que proporciona una guía para las organizaciones respecto a la ISO 9001, en la adquisición, suministro, desarrollo, operación y mantenimiento de software y servicios de soporte. Esta norma se basa principalmente en el Sistema de Gestión de la Calidad, la Responsabilidad de la Dirección, la Gestión de Recursos, la Realización del Producto y las Medida, el Análisis y la Mejora. Esta norma es para cuando la empresa puede demostrar la calidad de su producto mediante una inspección final.

Algunos de los beneficios que proporciona la ISO 9000-3, es el mejoramiento en la documentación de los sistemas, el incremento en la eficiencia y productividad, mayor percepción de calidad y amplia satisfacción del cliente.

SQA en la ISO 9000-3

Para el estándar ISO 9000-3 el aseguramiento de la calidad tiene como finalidad proporcionar la seguridad de que los productos y los procesos en el ciclo de vida del proyecto están de acuerdo a los requisitos específicos y que se ajustan a los planes establecidos. El aseguramiento de la calidad puede ser interno o externo dependiendo de si la evidencia de la calidad del producto o del proceso se demuestra a la gerencia del proveedor o del cliente. (Rivero Canino, y otros, 2088). Este proceso implementa diferentes actividades:

Implementación del proceso: Esta actividad cuenta con varias tareas como: establecer un proceso de aseguramiento de la calidad ajustado al proyecto. Elaborar, documentar,

implementar y mantener actualizado un plan de ejecución de las actividades y tareas del proceso. Implementar normas, metodologías, procedimientos y herramientas para asegurar la calidad. Proporcionar al cliente los registros de las actividades y tareas de aseguramiento de la calidad.

Aseguramiento del producto: Dentro de esta actividad se encuentran diversas tareas como: asegurar que todos los planes requeridos por el contrato estén documentados. Asegurar que todos los productos de software usados y la documentación relacionada con ellos cumplan lo establecido. En la preparación de la entrega de los productos, deberá ser asegurado que ellos satisfacen totalmente sus requisitos contractuales y que son aceptables para el cliente.

Aseguramiento del proceso: Esta actividad está formada por las siguientes tareas: asegurar que los procesos del ciclo de vida del producto cumplen con lo establecido y se ajustan a los planes. Asegurar que las prácticas internas del diseño del producto, del ambiente de desarrollo y del ambiente de prueba cumplen lo establecido en el contrato.

Aseguramiento del sistema de calidad. Esta actividad la integran un grupo de tareas que se mencionan a continuación: asegurar actividades adicionales de gestión de la calidad de acuerdo con las cláusulas de la norma ISO 9001.

1.5.3. ISO y CMMI

ISO 9000 y CMMI surgieron para ayudar a establecer buenas prácticas en el desarrollo de Software. Ambas buscan obtener calidad en el desarrollo del producto, aunque para uno y otro, depende mucho el objetivo y la visión de la empresa, es decir es cuestión de decisión.

La principal diferencia entre los dos es que ISO 9000 especifica un nivel aceptable mínimo de calidad para procesos de desarrollo de software, mientras que CMMI establece una estructura conceptual de referencia para medir la mejora continua de procesos, ya sea con los niveles de madurez o con los niveles de capacidad, con el objeto de dar a conocer el camino hacia un estado óptimo de mejora continua.

ISO 9000 es un conjunto de enunciados, que indica que elementos deben integrar un sistema de calidad, no definen como debe ser el sistema de calidad, sino que fija requisitos mínimos que debe cumplir. Permite que la empresa defina su propio sistema de calidad de acuerdo a sus características, respetando los requisitos. Mientras que CMMI es un modelo que establece

un conjunto de prácticas o procesos agrupados en áreas de procesos claves de la organización, y se basa en el modelo IDEAL, para desarrollar el sistema de calidad de la organización y requiere que para tener éxito se mejore el ambiente laboral.

A continuación se muestran 3 diferencias identificadas entre estos dos estándares:

| ISO 9000 | CMMI |
|--|---|
| Es muy general, ya que su concepción fue pensada para abarcar cualquier tipo de empresa. | Está enfocada hacia el desarrollo concreto de software. |
| Es de carácter estático, proporcionando mecanismos de mejora mucho más débiles. | Es de carácter incremental, lo cual motiva al personal. |
| Provoca una realidad empresarial en la que el único reto es mantener la certificación. | Mueve al sistema de calidad de la organización en una dirección de mejora continua. |

Tabla 1: Comparación entre ISO y CMMI (Elaboración Propia).

1.6. Fábricas o Factoría de Software

La expresión “Fábrica de Software” aparece por primera vez en el año 1968, cuando Robert William Bemer presenta una propuesta de esta al congreso de la International Federation of Information Processing (IFIP). La primera Fábrica de Software instalada fue en el año 1968, luego de esto vinieron un sin número de ellas. (Fábrica Académica de Software, 2008). El software no se fabrica, por lo que la expresión Fábrica de Software, simplemente trata de expresar la necesidad de copiar las buenas prácticas que tienen algunas fábricas, para tratar de desarrollar el software con mayor rapidez y calidad.

Una de las definiciones de Fábrica de Software, plantea que: Una factoría de software es una organización con procesos estructurados, controlados y mejorados de forma continua, considerando principios de Ingeniería Industrial, orientados a dar respuesta a múltiples demandas de distintas naturaleza y alcance. Dirigida a la creación de productos de software, conforme a los requerimientos documentados de los usuarios y clientes, de la forma más productiva y económica posible. (Hoz, 2005).

Las Factorías de Software, buscan industrializar el desarrollo de sistemas, rescatar ese proceso del ámbito del cliente y trasladarlo a un medio donde se establezcan unos procedimientos y métodos definidos que cuenten con herramientas que ayuden en su

implantación. Se perfilan, en la unión del conocimiento y la metodología, en la que se acumule todo lo desarrollado, lo que permite conseguir altos porcentajes de reutilización. (Trujillo, 2009).

Las Fábricas de Software se dirigen hacia la producción organizada y estructurada de software, que ofrezcan servicios diferenciados y orientadas a incrementar la calidad del producto final.

Objetivos de una Factoría de Software

Entre los principales objetivos trazados por una factoría de software están:

- 1- Industrializar el desarrollo de sistemas de software.
- 2- Producción de software a gran escala.
- 3- Lograr una alta productividad en el desarrollo de software.
- 4- Establecer una línea de producción.
- 5- Mejora continua de los procesos.
- 6- Estimación de costos y plazos extremadamente precisa.
- 7- Reducción de los costos de producción.
- 8- Lograr un buen control de la calidad.
- 9- Especializar al profesional en una tarea específica del proceso, concentrando sus esfuerzos en dicha tarea.

Una Fábrica de Software es una línea de producto que configura herramientas de desarrollo extensibles, con contenido empaquetado y guías, cuidadosamente diseñadas para construir tipos específicos de aplicaciones. Si se lleva más allá la analogía, los productos son como comidas servidas en un restaurante:

Los clientes de la fábrica de software son como los clientes que ordenan comidas de un menú. Una especificación de producto es como una orden de comida específica. Los desarrolladores del producto son como cocineros que preparan las comidas descritas en las órdenes, los cuales podrían modificar las definiciones de las comidas, o preparar comidas fuera del menú. Los desarrolladores de la línea de producto son como chefs que deciden qué aparecerá en el menú, y qué ingredientes, procesos, y equipo de cocina se usará para prepararlos. (Casal, 2005).

SQA en las Factorías de Software

El SQA en las Fábricas de Software que lo han puesto en práctica, están orientadas a garantizar que tanto el proceso de desarrollo como el producto final cumplan con las expectativas del usuario, dentro de las limitaciones de tiempo y presupuesto.

Las actividades de SQA en proyectos grandes, como las fábricas de software, son desarrollados por un grupo específico, una Gerencia de SQA. El Plan de Aseguramiento de la Calidad del Software (SQAP) define las actividades específicas a llevar a cabo en un proyecto específico. El SQAP contiene una lista de comprobación para las actividades que se deben llevar a cabo para asegurar la calidad del producto. Para cada fase del proyecto, se debe crear un plan para su monitoreo. (Universidad Católica de Chile. Departamento de Ciencia de la Computación, 2002).

En el proceso de Gerencia SQA se realizan diferentes actividades para poder lograr una buena adherencia con los estándares. Para ello se debe medir cuantitativamente, donde sea posible, los aspectos de calidad (complejidad, confiabilidad, mantenimiento, seguridad, defectos, número de problemas) utilizando métricas bien establecidas.

Para cumplir con esto, se deben realizar chequeos de:

- Administración.
- Documentación.
- Estándares, prácticas, convenciones y métricas.
- Revisiones e intervenciones.
- Actividades de testeo.
- Reporte de errores y acciones correctivas.
- Herramientas, técnicas y métodos.
- Control del código

- Control de medios.
- Colección de registros, mantenimiento y retención.
- Control de los proveedores.
- Entrenamiento.
- Administración del riesgo.

La forma en que se llevarán a cabo estas actividades se define en el SQAP, el cual irá evolucionando en las sucesivas fases del desarrollo. Para guiar a la Gerencia de SQA, en el siguiente punto se apuntan las guías para cada una de las actividades. (Universidad Católica de Chile. Departamento de Ciencia de la Computación, 2002).

También existen centros que se dedican a la preparación de estrategias de SQA para fábricas de software. Son puestas en práctica en todas las fases del ciclo de vida del software, identificando y administrando los riesgos en los proyectos de manera preventiva, también abarcan la estrategia y planificación de pruebas, la automatización de la prueba, la ejecución de la prueba (Estático, Funcional, Integración, Aceptación del Usuario, Rendimiento, Regresión, Usabilidad) y el Entrenamiento. (CRS IT CONSULTING, 2009)

1.7. Líneas de Productos de Software (LPS)

Reutilización de software es el proceso de crear sistemas de software a partir de software existente, en lugar de desarrollarlo desde el comienzo. (Montilva, 2006). La reutilización fue evolucionando hasta convertirse en lo que hoy conocemos como Líneas de Productos de Software o Familias de Productos de Software.

Las LPS se definen como un conjunto de sistemas software, que comparten un conjunto común de características, las cuales satisfacen las necesidades específicas de un dominio o segmento particular de mercado, y que se desarrollan a partir de un sistema común de activos base de una manera preestablecida. (Díaz, y otros, 2010).

En estos momentos no se quiere producir un único producto, sino una cadena que gestione eficiente y eficazmente las diferentes variaciones que pueden existir entre los productos. Ya

no es centrarse en un producto para un cliente (un SIG con los hospitales de Cuba), sino en un dominio (ejemplo, un SIG - Salud, abarca todo lo relacionado con Salud Pública).

Modelo Básico de una LPS

La entrada: Activos de Software: Una colección de partes de software (requisitos, diseños, componentes, casos de prueba, etc.) que se configuran y componen de una manera prescrita para producir los productos de la línea.

El control: Modelos de Decisión y Decisiones de Productos: Los Modelos de Decisiones describen los aspectos variables y opcionales de los productos de la línea. Cada producto de la línea es definido por un conjunto de decisiones (decisiones del producto).

El proceso de producción: Establece los mecanismos o pasos para componer y configurar productos a partir de los activos de entrada. Las decisiones del producto se usan para determinar que activos de entrada utilizar y como configurar los puntos de variación de esos activos.

La salida: Productos de software: Conjunto de todos los productos que pueden o son producidos por la línea de productos.

(Ver Anexo 5)

Beneficios de las LPS

Los beneficios que se pueden obtener con este modelo de desarrollo se pueden atribuir a la estrategia que representa la reutilización del software, esto contribuye a eliminar toda duplicación de esfuerzos en los procesos de ingeniería e implementación. Algunos de estos beneficios son:

- Reducción en el tiempo promedio de creación y entrega de nuevos productos.
- Reducción del esfuerzo promedio requerido para desarrollar y mantener los productos.
- Reducción del costo promedio de producción, del tiempo de salida al mercado y el tiempo de recepción de ingresos de nuevos productos.
- Incremento en el número total de productos que pueden ser efectivamente desplegados y mantenidos.
- Mejoras en el valor competitivo, la calidad y la reputación del producto y la empresa.

- Mayores márgenes de ganancia.
- Escalabilidad mejorada del modelo de negocio en términos de productos y mercados.
- Una mayor agilidad para expandirse en nuevos mercados.
- Reducción del riesgo en las implementaciones y entrega de productos.

1.8. Los Sistemas de Información Geográfica

En la actualidad estos sistemas se han convertido en una primordial herramienta para el manejo de información espacial o territorial en el cual se apoyan especialistas e investigadores de la rama.

Los SIG son un sistema de hardware, software y procedimientos, diseñados para soportar la captura, el manejo, la manipulación, el análisis, el modelado y el despliegue de datos espacialmente referenciados (georeferenciados), para la solución de los problemas complejos del manejo y planeamiento territorial. (Díaz, y otros, 1994).

Un Sistema de Información Geográfica (SIG o GIS, en su acrónimo inglés) es una integración organizada de hardware, software, datos geográficos y personal, diseñado para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente referenciada con el fin de resolver problemas complejos de planificación y gestión. (Pantoja Zaldívar, 2010).

Un SIG es cualquier sistema de información capaz de integrar, almacenar, editar, analizar, compartir y mostrar la información geográficamente referenciada.

Funciones de los SIG

Los SIG pueden resolver diferentes cuestiones, entre estas:

1. Localización: Preguntar por las características de un lugar concreto.
2. Condición: El cumplimiento o no de unas condiciones impuestas al sistema.
3. Tendencia: Comparación entre situaciones temporales o espaciales distintas de alguna característica.
4. Rutas: Cálculo de rutas óptimas entre dos o más puntos.
5. Pautas: Detección de pautas espaciales.

6. Modelos: Generación de modelos a partir de fenómenos o actuaciones simuladas.

Aplicaciones de los SIG

Los SIG pueden ser utilizados para investigaciones científicas, la gestión de los recursos, gestión de activos, la arqueología, la evaluación del impacto ambiental, la planificación urbana, la geodesia, la geoinformática, la sociología, la geografía histórica, el marketing, la logística y otros. Estos pueden ser utilizados como una herramienta de ayuda a la gestión y toma de decisiones, algunas de sus aplicaciones principales son:

Infraestructura: Para empresas encargadas del desarrollo, mantenimiento y administración de redes de electricidad, gas, agua, teléfono y alcantarillado. Estos sistemas almacenan información relativa a la conectividad de los elementos representados gráficamente, con el fin de realizar un análisis de redes. También son utilizados en trabajos de ingeniería, inventarios, planificación de redes, gestión de mantenimiento, entre otros.

Planimetría: Se refiere a la representación bidimensional del terreno proporcionándole al usuario la posibilidad de proyectar su trabajo sobre un papel o en pantalla sin haber estado antes en el sitio físico del proyecto.

Gestión territorial: Son aplicaciones SIG dirigidas a la gestión de entidades territoriales y permiten un rápido acceso a la información gráfica y alfanumérica, y suministran herramientas para el análisis espacial de la información. Facilitan labores de mantenimiento de infraestructura, mobiliario urbano, entre otros, y permiten realizar una optimización en los trabajos de mantenimiento de empresas de servicios.

Medio Ambiente: Son aplicaciones implementadas por instituciones de medio ambiente, que facilitan la evaluación del impacto ambiental en la ejecución de proyectos. Facilitan una ayuda fundamental en trabajos tales como reforestación, explotaciones agrícolas, estudios de representatividad, caracterización de ecosistemas, estudios de fragmentación, estudios de especies, entre otros.

Recursos Mineros: El diseño de estos SIG facilitan el manejo de un gran volumen de información generada en varios años de explotación intensiva de un banco minero, suministrando funciones para la realización de análisis de elementos puntuales (sondeos o

puntos topográficos), lineales (perfiles, tendido de electricidad), superficies (áreas de explotación) y volúmenes (capas geológicas).

Demografía: Se evidencian en este tipo de SIG un conjunto diverso de aplicaciones cuyo vínculo es la utilización de las variadas características demográficas, y en concreto su distribución espacial, para la toma de decisiones. Algunas de estas aplicaciones pueden ser: el análisis para la implantación de negocios o servicios públicos y zonificación electoral. El origen de los datos regularmente corresponde a los censos poblacionales elaborados por alguna entidad gubernamental.

1.9. Factoría de Software Aplicativos SIG

Aplicativos SIG utiliza como base de apoyo la Plataforma Soberana para el desarrollo de Sistemas de Información Geográfica (GeneSIG). La que surge por la necesidad de contar con un producto que sirva como soporte de aplicaciones SIG en entornos Web con tecnologías libres.

Sus objetivos principales son permitir la representación geoespacial de la información asociada a cualquier negocio que lo requiera. Proporcionar servicios de acceso a la información geográfica, para su consulta, análisis y visualización, mediante una interfaz de usuario sencilla y de fácil manejo que pueda ser utilizada por usuarios no especializados en tecnología SIG. Integrar la información socioeconómica existente (Recursos Humano, Activos Fijos, Entidades de Servicios, Lugares de interés, etc.) con la información geográfica asociada.

Actualmente se encuentran en desarrollo varios productos que responden a necesidades de diversas ramas de la sociedad:

- SIG-Salud: Es una aplicación diseñada con el objetivo de permitir la representación geoespacial de la información asociada a las instituciones de salud pública de Venezuela. Proporciona servicios de acceso a la información geográfica, para su consulta, análisis y visualización, mediante una interfaz de usuario sencilla y de fácil manejo que pueda ser utilizada por usuarios no especializados en tecnología de Sistemas de Información Geográfica. Además de brindar mecanismos de edición de mapas, creación de gráficos y personalización de funcionalidades por roles de usuarios.

- SIG-Rutas: Permite consultar y analizar la información referente a las rutas de transporte obrero de la Universidad de las Ciencias Informáticas. Presenta cuatro módulos, que engloban todas las funcionalidades de:
 - Módulo de navegación: Este modulo contempla el trabajo con los mapas y es fundamental su presencia no solo en esta aplicación sino en toda aquella que trabaje con información geoespacial.
 - Módulo de localización: En esta parte se han agrupado las funcionalidades asociadas con la localización de los elementos georeferenciables.
 - Módulo de opciones: Este módulo contempla una serie de funcionalidades propias del trabajo con este sistema. Ejemplo: Localizar parada más cercana.
 - Módulo administración: Interactúa con datos y funciones críticas del sistema, tales como agregar y modificar los datos de los elementos existentes en la base de datos así como agregar o desactivar funcionalidades del mismo sistema.
- SIG-INSMET: Responde a las necesidades del Instituto de Meteorología de Cuba de contar con un sistema informático que le permita conocer en tiempo real el comportamiento de un número de parámetros que se capturan en las estaciones meteorológicas automatizadas.
- SIG-Verano: Es una aplicación en desarrollo diseñada con el objetivo de permitir la representación geoespacial de la información asociada a la disponibilidad de los distintos locales que son utilizados para el recreo y disfrute de la familia cubana de organismos e instituciones de nuestro país en cada una de las provincias durante la etapa veraniega. La misma cuenta con una interfaz de usuario sencilla y de fácil manejo que pueda ser utilizada por usuarios no especializados en tecnología de Sistemas de Información Geográfica (SIG). Con esta herramienta SIG la Oficina Nacional de la UJC podrá lograr una mejor toma de decisiones en cuanto a gestionar de una forma automatizada y sencilla toda la información referente a aquellos locales disponibles para la recreación y el entretenimiento.
- SIG-INRH: Necesidad del Instituto Nacional de Recursos Hidráulico de la República de Cuba de contar con un sistema que le permita obtener datos y georeferenciar los recursos hidráulicos existentes en la isla.

- SIG-MIC: El producto contará con un conjunto de funciones necesarias para la representación, modelación y análisis de la información geográfica. Permitirá la representación geoespacial de la información asociada a las entidades del MIC que tienen contratos de trabajo en Venezuela. Brindará además servicios de acceso a la información geográfica. Los usuarios del sistema podrán consultar, analizar y visualizar toda la información referente a dichas entidades mediante una interfaz de usuario sencilla y de fácil manejo.
- SIG-UCI: Es un producto encaminado a realizar la representación y análisis geoespacial de información geográfica referente a objetivos socioeconómicos de la Universidad de las Ciencias Informáticas (UCI), como edificios docentes, edificios de residencia, plazas, manzanas, consultorios, cajeros, etc. Los usuarios, que son los estudiantes, trabajadores y profesores de la UCI, podrán contar con una aplicación de características Web y sostenida sobre una plataforma de desarrollo libre. Existe un solo entorno de trabajo para el desarrollo de la solución que se propone, se obtendrá una aplicación Web utilizando para el desarrollo la tecnología PHP, MapServer y PostgreSQL; y se trabaja sobre el Sistema Operativo Linux (Ubuntu).

1.9.1. Proceso de Desarrollo de un producto en Aplicativos SIG

La Factoría Aplicativos SIG realiza su ciclo de vida siguiendo la metodología de Rational Unified Process (RUP), sus fases de desarrollo están divididas en: Inicio, Elaboración, Construcción y Transición.

Está constituido por 5 grupos de factorías para el desarrollo de un producto, que funcionan como las fases del RUP, por cada una de estos tienen que pasar el producto que esté en desarrollo y terminada la iteración quedaría terminado el SIG, para cada grupo fue designado un jefe, que planifica las tareas y actividades que se realizarán en su grupo. Este proceso de desarrollo fue implementado por la factoría, por lo que es necesaria una explicación de cada una de estos para un mejor entendimiento (Ver Tabla 1).

| No. | Grupos | Descripción | Roles | Herramientas |
|-----|-------------------------------|--|---|--|
| 1 | Ingeniería | <p>Esta se divide en:</p> <p>Preliminar: Se identifican y se toman los acuerdos con el cliente, se establece el alcance del proyecto, los criterios de éxito, la identificación de los riesgos, la estimación de recursos necesarios del proyecto, además de identificar cualquier otra característica de la aplicación.</p> <p>Requisitos: Los desarrolladores y clientes deben acordar qué es lo que el sistema debe hacer. Se confecciona toda la documentación asociada al expediente de proyecto.</p> | <p>Jefe de Grupo.</p> <p>Analista.</p> <p>Planificador.</p> | Visual Paradigm. |
| 2 | Integración y Componente (IC) | Se integran y se hacen los componentes. Se une la Interfaz de Usuario con la Base de Datos. | <p>Líder de Factoría.</p> <p>Jefe de Grupo</p> <p>Desarrollador.</p> <p>Planificador.</p> | <p>Netbeans.</p> <p>PHP como Lenguaje de Programación.</p> |
| 3 | Base de Datos Espaciales (BD) | Se hacen las actividades de cartografía, funciones y almacenado de los SIG. | <p>Jefe de Grupo.</p> <p>Diseñador de Base de Datos.</p> <p>Planificador</p> | <p>PGAdmin.</p> <p>PostgreSQL.</p> <p>PostGIS.</p> |
| 4 | Interfaz de Usuario (IU) | Se encarga de toda la parte visual de los SIG, como es el diseño, las imágenes y los componentes visuales. Se preparan los componentes principales de la interfaz de manera que sean fáciles de integrar. | <p>Jefe de Grupo.</p> <p>Diseñador de Interfaz de Usuario.</p> <p>Planificador.</p> | <p>Biblioteca EXT.</p> <p>Biblioteca EXT.JS.</p> |

| | | | | |
|---|---------|--|--------------------------------|----------|
| 5 | Gestión | Este grupo gestiona todo lo relacionado con los Recursos Humanos, los Recursos Materiales, el Marketing y la Contabilidad. | Jefe de Grupo. Planificador | REDMINE. |
|---|---------|--|--------------------------------|----------|

Tabla 2. Grupos de Factorías de Aplicativos SIG (Elaboración Propia)

1.10. Conclusiones Parciales

En este capítulo se trató el tema de las factorías de software, su evolución, las definiciones hechas por algunos autores, las características de estas y otros temas que se refieren a la factoría Aplicativos SIG, como las LPS y los SIG. Se estudiaron las definiciones que de una forma u otra tienen que ver con el aseguramiento de la calidad como: calidad y calidad de software; también se estudiaron los estándares, modelos y procedimientos utilizados internacionalmente y en la UCI.

Los temas tratados en el capítulo permiten una mejor comprensión a la hora de hablar sobre una estrategia de aseguramiento de la calidad a una factoría, haciendo imprescindible el conocimiento y la preparación de quien proponga o aplique dicha estrategia.

Capítulo 2: Estrategia de Aseguramiento de la Calidad para la Factoría de Software Aplicativos SIG

2.1 Introducción

Cuando se habla de calidad se abarcan todas las cosas buenas que pueda tener un producto. Si al desarrollarlo se asegura también la calidad de este, todos los implicados quedarán satisfechos, pero para lograrlo, se deben trazar una serie de actividades que permitan un buen SQA.

Luego de analizadas las particularidades de Aplicativos SIG en el capítulo anterior, se realiza la propuesta de estrategia de SQA con todas las actividades que se realizarán para darle cumplimiento.

2.2 Estrategia de Aseguramiento de la Calidad para Aplicativos SIG

Una estrategia de SQA es una serie de pasos planificados que se deben aplicar desde etapas tempranas del proyecto, con el objetivo de reducir los errores que puedan aparecer en el desarrollo del producto.

La estrategia se aplicará y se mantendrá a lo largo de todo el ciclo de vida de la factoría, confeccionando un Plan de Aseguramiento de la Calidad (PSQA), con el objetivo de planificar todas las acciones necesarias, para garantizar la mayor eficacia en cada grupo de desarrollo del producto. Además de organizar el trabajo de los Recursos Humanos de la factoría y entrenarlos correctamente para que puedan realizar su trabajo. (Ver Anexo 1)

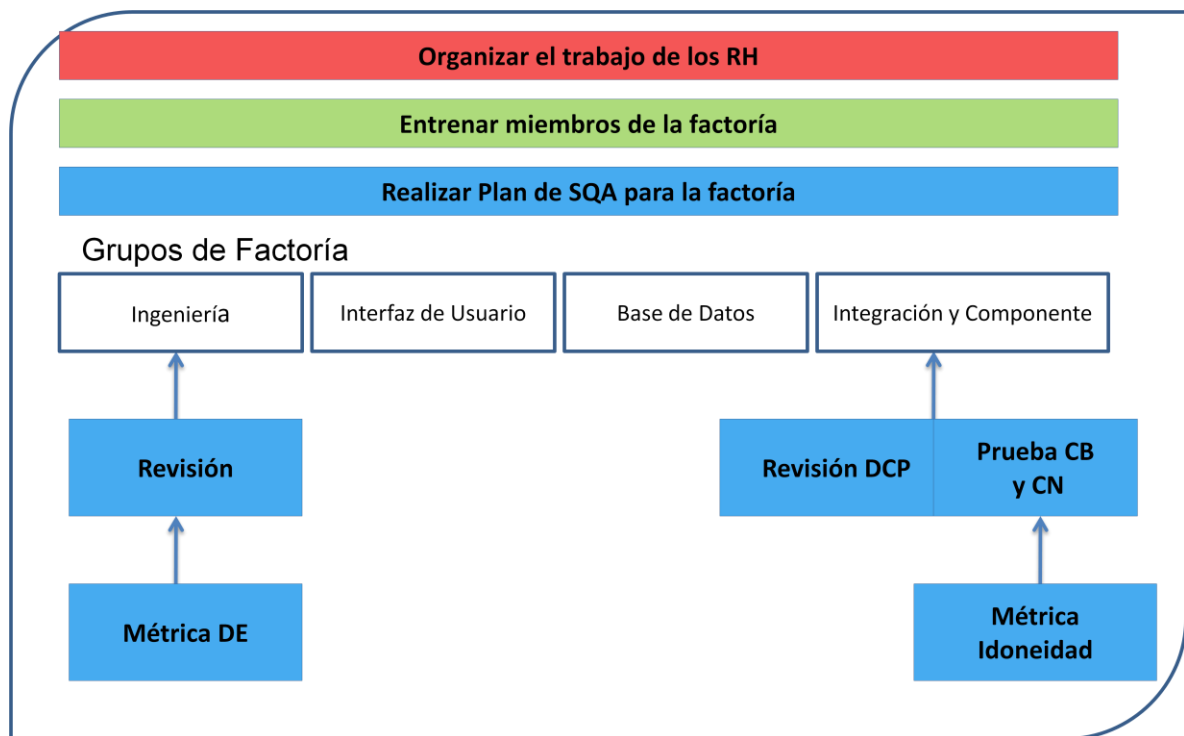


Figura 1: Relación de la Estrategia de SQA con el ciclo de vida de un producto en la Factoría

Estas actividades están conformadas por otras actividades que permiten un buen desarrollo de la estrategia.

2.1 Organizar el trabajo de los Recursos Humanos de la Factoría

Para la organización del trabajo de los Recursos Humanos de la factoría se proponen las siguientes actividades:

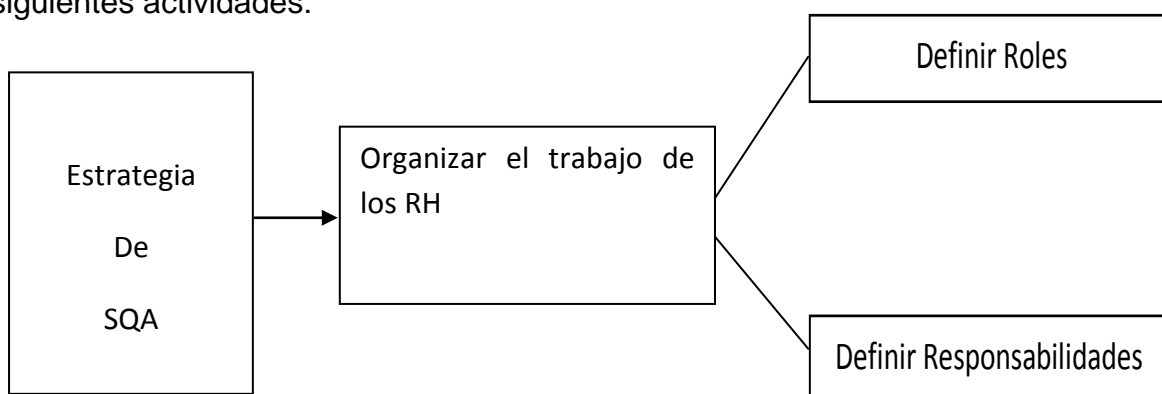


Figura 2: Actividades para la organización de los Recursos Humanos de la factoría.

En la factoría se definieron 7 roles, pero no tiene bien organizado el trabajo de cada uno de ellos, se propone redefinir los roles y agregar un nuevo rol, además de asignarle sus responsabilidades, para una mejor distribución del trabajo que debe realizar el equipo de trabajo.

| Roles de Factoría | Responsabilidades | Competencias |
|---|---|---|
| Líder de Factoría | Participar en la definición del proyecto. Tomar las decisiones en el proyecto. Aprobar las tecnologías a usar en el desarrollo del proyecto. Coordinar y organizar las tareas que se asignan a los miembros del equipo de desarrollo. Llevar a cabo todo el proceso de gestión de proyecto. | Poseer habilidades de liderazgo, trabajo en equipo, comunicación y toma de decisiones. Conocer las tecnologías generales. |
| Jefe de Grupo de Factoría. | Guiar todo el trabajo que se realizará en su grupo de factoría. | Conocer las tecnologías que utiliza el grupo. Poseer habilidades de liderazgo, trabajo en equipo, comunicación y toma de decisiones. |
| Diseñador de Base de Datos | Construir procedimientos almacenados, diseñar y normalizar la Base de Datos. | Dominar la utilización de lenguajes de consulta. Conocer los sistemas de gestión de bases de datos relacionales. Dominar la teoría de diseño de bases de datos. Normalizar esquemas de bases de datos en búsqueda de mejoras competitivas en tiempo y forma para gestión de la información. |
| Diseñador de Interfaz de Usuario | Crear el prototipo de interfaz de usuario. Es responsable de la codificación de la aplicación. En general, ejecutar cualquier tarea directamente involucrada con la producción de la interfaz de usuario. Colaborar con el diseñador gráfico para desarrollar un prototipo funcional. | Dominar las diferentes técnicas de diseño. Poseer conocimientos y habilidades con las tecnologías de diseño. |
| Desarrollador | Programar los servicios y funcionalidades del software construido para la Web. | Poseer habilidad en las técnicas de programación. Identificar la mejor alternativa de solución. |

| | | |
|---------------------------------|--|---|
| Analista | Participar en la definición del proyecto. Intervenir en la modelación del negocio. Interactuar con el usuario final en la definición de los requisitos de la aplicación. Crear el modelo del negocio. Identificar los requisitos de la aplicación. Crear el modelo del sistema. Definir el prototipo de interfaz de usuario elemental. Traducir la comunicación entre usuarios finales y desarrolladores. Gestionar los requisitos adicionales que aparezcan durante el desarrollo del software. | Identificar y comprender los problemas y oportunidades. Ser un buen moderador y tener habilidades de comunicación. Conocer sobre Ingeniería de Software y otras metodologías de desarrollo de software. |
| Planificador | Mantener actualizado el plan de proyecto, cronogramas y fechas de entregas. Controlar el cronograma de ejecución del proyecto. Controlar y planificar el uso de los recursos del proyecto. Emitir informes periódicos del estado de avance del proyecto. Medir la eficiencia del desarrollo, controlar los tiempos de ejecución, imprevistos, contratiempos. | Poseer conocimiento de las actividades del proyecto. Conocer todos los recursos y necesidades del proyecto. Poseer habilidades para la comunicación y el trabajo en equipo. |
| Administrador de Calidad | Realizar las actividades que se realicen para asegurar la calidad en la factoría. | Conocer la metodología de desarrollo de software de la factoría, para definir la estrategia acorde a las necesidades de ella. Poseer conocimiento de las revisiones y las pruebas, para proporcionar una guía al equipo de revisores. |

Tabla 3: Tareas y Responsabilidades de la Factoría Aplicativos SIG. (Elaboración Propia)

2.2 Entrenar los miembros de la factoría

Para la actividad de entrenamiento se proponen varias actividades que permitirán obtener buenos resultados en la factoría, gracias a la buena preparación de los integrantes. (Ver Anexo 2):

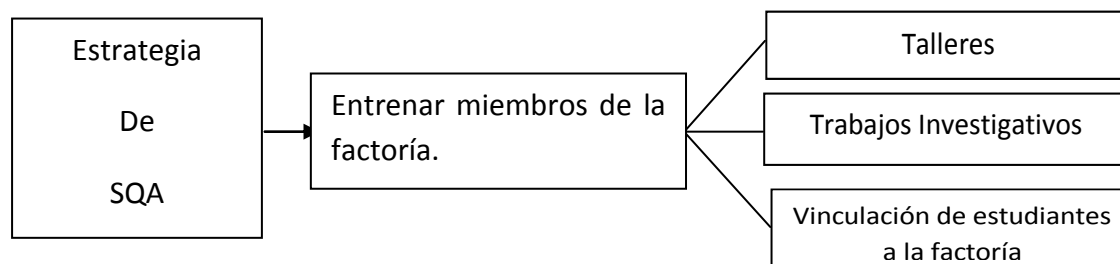


Figura 3: Actividades para el entrenamiento de los miembros de la factoría.

Para entrenar a los miembros de Aplicativos SIG y al equipo de calidad se definieron un conjunto de actividades, que les permitirán desarrollar conocimientos y habilidades a los involucrados en el desarrollo de los productos que se realicen:

Talleres:

1. Realizar taller sobre Expediente de Proyecto, tratar todos los aspectos relacionados con el tema.
Grupo en el que se realiza: Al comenzar el trabajo del grupo de ingeniería.
Responsable: Jefe de Grupo de Ingeniería.
2. Taller para explicar cómo trabajar con la herramienta Visual Paradigm.
Grupo en el que se realiza: Al comenzar el trabajo del grupo de ingeniería.
Responsable: Jefe del Grupo de Ingeniería.
3. Taller de revisiones al grupo de calidad, para la revisión del Expediente de Proyecto.
Grupo en el que se realiza: Antes de comenzar las revisiones al Expediente de Proyecto.
Responsable: Administrador de Calidad.
4. Taller a todos los miembros del grupo de BD, sobre las herramientas PGAdmin, PostGreSQL y PostGIS.
Grupo en el que se realiza: Antes de comenzar el trabajo del grupo de BD.
Responsable: Jefe de Grupo de BD.
5. Taller sobre el lenguaje de PHP a los miembros del grupo de IU e IC, como el lenguaje de programación que utiliza la factoría, por lo que se debe hacer siempre que sea necesario.
Grupo en el que se realiza: Antes de comenzar el trabajo de estos grupos.
Responsable: Líder de Factoría.
6. Taller sobre las Bibliotecas Ext y Ext.Java, para los miembros del grupo IU.
Grupo en el que se realiza: Antes de comenzar el trabajo del grupo IU.
Responsable: Jefe del Grupo IU.

7. Taller de revisiones al grupo de calidad, para las revisiones a los Diseños de Caso de Prueba.

Grupo en el que se realiza: Antes comenzar la revisión a los DCP en el grupo de IC.

Responsable: Administrador de Calidad.

8. Taller de Pruebas para los desarrolladores y probadores.

Grupo en el que se realiza: Antes de comenzar el trabajo del grupo IC.

Responsable: Administrador de Calidad.

Trabajos Investigativos:

1. Sobre los roles y responsabilidades de los roles que se han definido en cada grupo de la factoría.

Responsable: Jefes de Grupo de cada Factoría.

Grupo en el que se realiza: Todos los grupos de factoría deben hacerlo.

2. Sobre otras herramientas, que se pueden utilizar para el trabajo en los grupos de factoría.

Responsable: Jefes de Grupo de cada Factoría.

Grupo en el que se realiza: Todos los grupos de factoría deben hacerlo.

3. Responsabilidades de los roles definidos para el SQA.

Responsable: Jefes de Grupo de cada Factoría.

Grupo en el que se realiza: Antes de comenzar las actividades de SQA en la factoría.

Vinculación de Estudiantes:

1. Vincular a los estudiantes de 3er año a la producción, 4 horas por semana. Para una completa integración la factoría, se deben rotar por todos los grupo e involucrarlos en talleres y trabajos investigativos que se realicen en cada grupo, para poder determinar en cuál de estos tiene un mejor desenvolvimiento y poder valorar la posibilidad de integrarlo a uno de los grupos.

Responsable: Jefe de Grupo de Gestión y Líder de Factoría.

Grupo en el que se realiza: En todos los grupos de factoría.

2.3 Plan de Aseguramiento de la Calidad

El Plan de Aseguramiento de la Calidad, es un artefacto que provee una visualización clara de cómo debe ser seguro un producto, artefactos, y la calidad del proceso. Contiene el plan de revisiones y auditorías. Es mantenido durante todo el proceso de desarrollo del software. (Coba Santiesteban, 2008).

El PSQA es la actividad que permitirá asegurar la calidad del producto durante todo su desarrollo. Aquí están definidas las principales actividades que se llevarán a cabo para obtener resultados satisfactorios en el desarrollo del producto. (Ver Anexo 4):

La realización del plan fue basada en la guía propuesta por la Dirección de Calidad de Software (Calisoft) de la UCI y contiene las siguientes actividades:

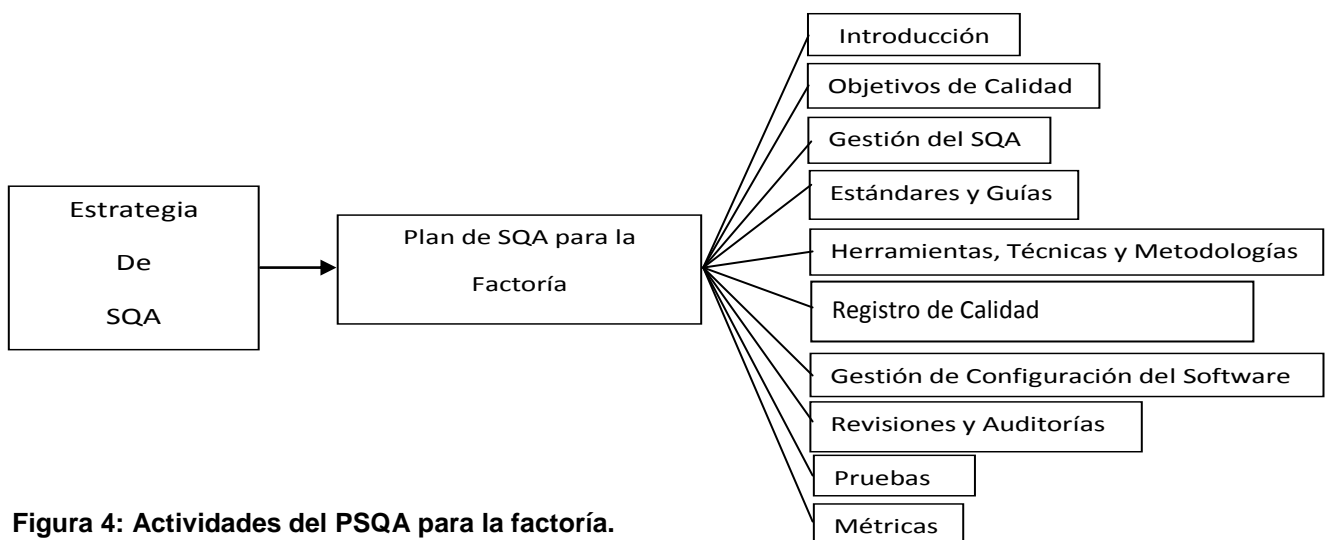


Figura 4: Actividades del PSQA para la factoría.

2.3.1 Objetivos de Calidad

Antes de que puedan iniciarse las actividades de aseguramiento de la calidad es importante definir cuáles son los objetivos de calidad de la factoría:

- Asegurar que el producto de Software le permitirá al usuario entender si este es idóneo para la aplicación a la cual lo destina.
- Asegurar que el producto de software sea comprendido, aprendido y utilizado por el usuario.

- Garantizar que los diferentes sistemas cumplan con las necesidades de los usuarios, y sean comprensibles.
- Asegurar que el software sea amigable y atractivo para el usuario.
- Asegurar que el producto pueda permitirle al usuario operarlo y controlarlo.

Responsable: Administrador de Calidad.

(Ver Anexo 3).

2.3.2 Gestión del Aseguramiento de la Calidad

La factoría de Software Aplicativos SIG tiene entre sus trabajadores estudiantes y profesores de la facultad 6. Por lo que el grupo de Aseguramiento de la Calidad debe estar integrado por los siguientes roles:

| Roles del SQA | Responsabilidades | Competencias |
|---------------------------------|---|---|
| Líder de Factoría | Encargado de planear las evaluaciones e incluirlas en el cronograma del proyecto, analizar los resultados de las evaluaciones, elaborar las acciones correctivas para resolver las no conformidades detectadas durante las evaluaciones y comunicar las acciones correctivas a los revisores líderes de las evaluaciones. | Poseer habilidades de liderazgo, trabajo en equipo, comunicación y toma de decisiones. Conocer las tecnologías generales. |
| Administrador de Calidad | Encargado de confeccionar del plan de aseguramiento de la calidad y mantener actualizado el registro de evaluaciones del proyecto, además de monitorear el cumplimiento de las acciones correctivas y el estado de las no conformidades. | Ser buen comunicador. Dominar técnicas de recolección de información. Dominar el ciclo de desarrollo de software. Conocer el modelo CMMI. Conocer los principales estándares internacionales en la producción de software, así como los procedimientos y lineamientos que norman la producción en la UCI. |

| | | |
|----------------------------|---|---|
| Coordinador de SQA. | Es el encargado de redactar solicitudes de mejora de procesos y productos, recepcionar y enviar la notificación de solicitudes de escalamiento al nivel correspondiente y realizar los reportes de SQA e informar a la Alta Gerencia sobre las solicitudes de mejora. | Conocer la técnica de dirección. Poseer capacidad de análisis y sensibilidad para detectar e identificar problemas y tomar decisiones. |
| Revisor Líder | Es el encargado de efectuar la reunión de inicio evaluación, organizar el equipo de revisores y distribuir las tareas, realizar las evaluaciones de desempeño de los revisores y comunicar los resultados. | Detectar e identificar problemas, capacidad de análisis, actitud constructiva y compromiso institucional. |
| Revisor | Participar en las revisiones que se realicen. Además de participar en otras actividades que deba hacer el equipo de trabajo. | Conocer todos los elementos del proyecto técnico. Facilidad de lectura, interpretación y capacidad de resumen. Dominar las reglas ortográficas y de estilos de redacción. |
| Probador | Encargado de probar los componentes de software desarrollados. | Poseer habilidad en la lectura de planes y casos de prueba. Conocer de los enfoques y técnicas de las pruebas. Habilidad de diagnóstico y resolución de problemas. Conocer del sistema o aplicación que se somete a prueba. |

Tabla 4: Roles del SQA. (Elaboración Propia)

Responsable: Administrador de Calidad.

(Ver Anexo 3).

2.3.3 Estándares y Guías

| Estándar | Nombre | Descripción |
|-----------------|--|--|
| LC | Lineamientos de Calidad de Software (IPL-3100:2008). | Constituyen una guía a seguir por los proyectos de desarrollo de software atendiendo cuáles son los artefactos que deben generar y que procesos deben realizar para aspirar a un producto final con calidad. |
| CMMI | Modelo de Madurez de Capacidad Integrado. | Modelo que permite catalogar a las organizaciones con el nivel de capacidad de madurez de su proceso de desarrollo. |
| ISO 9000 | Organización Internacional para la Estandarización. | Proporciona una guía útil que sirve para detectar y corregir una serie de problemas de los productos software, consiguiendo tras su aplicación una mejora en la calidad de los mismos. |

Tabla 5: Estándares y Guías. (Elaboración Propia)

Responsable: Administrador de Calidad.

(Ver Anexo 3).

2.3.4 Herramientas, Técnicas y Metodologías

- Se utilizará como herramientas las Listas de Chequeo, para las revisiones y las Plantillas del Expediente de Proyecto para registrar la documentación del proyecto.
- Para la realización de las revisiones se utilizará la técnica Revisiones Técnicas Formales (RTF), para revisar periódicamente cualquier representación que tenga el producto en busca de errores.
- La metodología de desarrollo que se utilizará como apoyo para el establecimiento de las pruebas en la factoría es RUP.

Responsable: Administrador de Calidad.

(Ver Anexo 3).

2.3.5 Registro de Calidad

En el registro de calidad de la Factoría de Software Aplicativos SIG se guardarán todos los documentos relacionados con esto. Este registro será un repositorio que se mantendrá mientras exista la factoría. Los documentos que se guardaran en el registro son:

- ✓ Plan de Aseguramiento de la Calidad.
- ✓ Informe de No Conformidades.
- ✓ Registro de Revisiones y Auditorias.
- ✓ Registro de Pruebas.
- ✓ Registro de capacitaciones y certificaciones.
- ✓ Casos de Pruebas.
- ✓ Plan de Pruebas.
- ✓ Listas de Chequeos.
- ✓ Informe de Gestión de Configuración.
- ✓ Actas de Reunión efectuadas.
- ✓ Resúmenes de Resultados.

Responsable: Administrador de Calidad.

(Ver Anexo 3).

2.3.6 Gestión de la Configuración del Software

Cuando se construye un software se realizan cambios en este, ocasionando confusión entre los desarrolladores del software, para minimizar dicha confusión está la Gestión de Configuración del Software (GCS), siendo esta un proceso que se aplica para mantener la integridad de los productos de trabajo.

El Expediente de Proyecto (EP) y la GCS persiguen lograr la calidad tanto en los procesos como en los productos de trabajado. La diferencia está en que el EP es una herramienta que agrupa y organiza todos los artefactos que se generan durante el desarrollo de software para garantizar la calidad del proceso de producción del software como del producto en si, mientras que la GCS es un conjunto de actividades de protección que se aplica durante todo el

proceso de desarrollo de software, permitiendo mantener la calidad durante los procesos de cambios que se realicen en los productos.

Para la GCS se realizan las siguientes actividades: Identificación de la Configuración (IC), Control de Versiones (CV), Control de Cambios (CC), Auditorías de Configuración (AC) y Generación de Informes (GI). Para realizar las actividades se utiliza como herramienta el Subversión para llevar el control de versiones con autenticación estática. El sistema cuenta con contenedores tanto para el código fuente a realizar como para la documentación que se vaya generando en el transcurso. Las posibles ramificaciones del proyecto también tienen su espacio en el repositorio y serán versionadas bajo el mismo sistema. Por último el sistema versionará también los entregables que se generen. Otra herramienta que se utiliza en las Auditorías a la Configuración son las RTF. (Ver en Expediente de Proyecto/Soporte/Gestión de Configuración/Plan de Gestión de Configuración de Software)

Responsable: Jefe de la Factoría y Jefes de Grupo de las Factorías.

Artefactos: Plan de Gestión de Configuración del Software. Pedido de Cambio. Informe de Autoridad de Control de Cambio para los resultados de la evaluación. La Orden de Cambio de Ingeniería para cada cambio aprobado. Lista de Chequeo y Lista de Recomendaciones cuando se realiza las RTF en la Auditorías a la Configuración.

(Ver Anexo 3).

2.3.7 Revisiones y Auditorías

Las auditorías y revisiones pueden detectar los defectos y corregirlos en tiempo, permitiendo de esta manera que el producto final, tenga la menor cantidad de errores posibles.

Se realizarán revisiones al grupo de factoría Ingeniería y a los DCP:

- Revisiones al Expediente de Proyecto:

Responsable: Equipo Revisor y Revisor Líder.

Grupo en el que se realiza: Al finalizar el trabajo del grupo Ingeniería.

Artefactos: Documentos del Expediente de Proyecto de Aplicativos SIG. Acta de Reunión de Apertura. Lista de Chequeo de Revisión Expediente v2.0. Lista de Recomendaciones. Informe Final y Acta de Reunión de Cierre. (Ver Anexos)

- Revisión a DCP:

Responsable: Equipo Revisor y Revisor Líder.

Grupo en el que se realiza: En el grupo IC.

Artefactos: Acta de Reunión de Apertura. Lista de Chequeo de Caso de Prueba basado en CU. Lista de Recomendaciones. Informe Final y Acta de Reunión de Cierre.

(Ver Anexo 3).

Las auditorías se realizan a nivel central, estas son conocidas como Auditorías a la actividad productiva y son coordinadas por Calisoft. El objetivo de estas es examinar y evaluar el grado de correspondencia entre los procedimientos, lineamientos y disposiciones establecidas y su aplicación a fin de determinar el grado de planificación, organización, dirección, control y si se han alcanzado las metas propuestas.

Resolución de problemas y actividades de corrección

A partir de las RTF se emitirán las No Conformidades (NC), el proceso para informar y manejar los problemas sería:

1. Una vez terminada la revisión el revisor debe emitir el documento de NC, si existe alguna.
2. Este documento debe ser enviado al jefe del grupo de la factoría bajo revisión.
3. El jefe de grupo y demás miembros del grupo procederán a resolver cada una de las NC encontradas.
4. Resuelta las NC, el jefe de grupo debe emitir un documento de respuestas a NC para el Líder de Proyecto.
5. El Líder de Proyecto decidirá si es necesaria una nueva revisión según la importancia que tengan las NC para la factoría, comunicándole al Administrador de Calidad el cual debe estar al tanto del cumplimiento de las acciones correctivas que se realicen.

6. Si el Líder de Proyecto, no considera que se puede realizar otra revisión, pero el Administrador de Calidad considera que las NC detectadas afectan el desarrollo del producto, se realizará otra revisión, en caso de no resolverse estas NC se realizará la solicitud de escalamiento de estas.
7. Los documentos de NC y respuesta a las NC deben almacenarse en el registro de calidad.

Niveles de Importancia de las NC para la factoría:

A – Alto: Impacta el cumplimiento de los objetivos, se pide que se detenga el desarrollo del producto y que se realice una acción correctiva.

M – Media: Puede impactar los objetivos del desarrollo del producto y se pide que se realice una acción correctiva.

B – Baja: No impacta los objetivos del desarrollo del producto, sólo se registra y se pide que se cumpla para la siguiente ocasión.

Niveles de escalamiento:

Nivel 1: Director General de la IP y Vicerrector de Producción.

Nivel 2: Directores de las direcciones y centros de servicio y soporte de la IP.

Nivel 3: Jefes de Grupos de las direcciones y centros de servicio y soporte de la IP.

Nivel 4: Decanos, directores de Áreas, jefes de centros de desarrollo.

Nivel 5: Gerentes, jefes de polos y líneas de producción.

2.3.8 Pruebas

Las pruebas son las verificaciones que se le realizan al producto de software, con el objetivo de encontrar y documentar defectos en la calidad de software.

La etapa de prueba es la más importante ya que en esta se ve la calidad con la que se ha desarrollado el producto. Para lograr esto se planifica y se ejecuta una serie de pasos que

definen cómo se efectuará el proceso de prueba, brindando la posibilidad de revisar todos los elementos del software.

Pasos para realizar las pruebas:

1. Descripción de las funcionalidades del sistema.
2. Una vez concluido las descripciones de cada una de las funcionalidades del sistema, diseñar los casos de pruebas.
3. Realizar pruebas de sistema, unidad e integración.
4. Revisar los diseños de casos de prueba y el modelo de sistema según listas chequeo.
5. Registrar los resultados de las pruebas mediante el documento “Lista de Recomendaciones R-XX-XXX” del expediente de proyecto.

(Ver Anexo 3).

Pruebas a realizar en la Factoría Aplicativos SIG

Pruebas de Unidad: Se realizarán con el objetivo de comprobar la calidad del código, partiendo de los diseños de caso de prueba. En esta se descubren la mayor cantidad de errores de programación. Para esta prueba se utilizará la técnica de:

- Pruebas de Caja Blanca (PCB): Durante la aplicación de esta técnica se trabaja con el código. Son realizadas con el objetivo de comprobar la calidad del código.

Responsable: Desarrollador.

Grupo en el que se realiza: Al terminar el trabajo del grupo IC.

Artefactos: Registro de No Conformidades.

Pruebas del Sistema: Estas pruebas comprueban la funcionalidad e integridad globalmente. Con la realización estas se podrán verificar el programa final después de que todos los componentes han sido integrados. Para su realización se utilizará la técnica:

- Pruebas de Caja Negra (PCN): Se centra en los requisitos funcionales del software, y se lleva a cabo sobre la interfaz del software. Con esta prueba se pretende comprobar si existen errores en la funcionalidad del producto, es decir se comprueban las interfaces del mismo para detectar defectos de función en ellas.

Responsable: Probador.

Grupo en el que se realiza: Esta prueba se aplica luego de realizada la prueba de CB y elaborado el Diseño de Caso de Prueba, en el grupo IC.

Artefactos: Diseño de Caso de Prueba, Registro de No Conformidades, Lista de Recomendaciones.

(Ver Anexo 3)

2.3.9 Métricas

Las métricas son una forma de entender, controlar y probar el desarrollo de software y son aplicadas con el objetivo de controlar qué es lo que ocurre en los proyectos, para poder mejorar procesos y productos.

Las métricas de calidad son un subconjunto de las métricas del software que se enfocan hacia los aspectos relacionados a la calidad del producto, proyecto o proceso. Las métricas de calidad están asociadas más con el proceso y el producto que con el proyecto. No obstante, los parámetros del proyecto (número de desarrolladores y el nivel de sus habilidades, el cronograma de trabajo, el tamaño y la estructura de la organización) afectan sensiblemente la calidad del software. Las métricas de calidad del software se dividen en dos categorías: métricas de calidad del producto terminado y métricas de calidad del proceso. (Kan, 2002).

Las métricas de calidad son las responsables de definir en el proceso de desarrollo de software el estado o progreso de este en cuanto a la calidad y proporcionar información sobre este.

Actualmente existen tres tipos de métricas, las cuales se dividen de acuerdo al producto, el proceso y el proyecto.

Métricas del Producto

Las métricas de productos se centran en las características del software, por lo que debe medirse cada una de las características del software. Se basan principalmente en predecir y controlar:

1. El tamaño (líneas de código, bytes de código, operadores y operando).
2. La estructura (control de flujo, relación entre componentes, cohesión y acoplamiento).

3. La complejidad (combinación de tamaño y estructura).
4. Los índices para controlar la documentación.
5. La calidad (independencia, completo, entendible, aumentado).
6. La estabilidad (los cambios aumentan el número de fallas, los cambios se pueden dar por definición de requerimientos o por cambios del entorno). (Departamento de Control de Calidad y Auditoría, 2000).

Métricas del Proceso.

Las métricas de procesos permiten obtener un conjunto de indicadores de proceso que conduzcan a la mejora de los procesos de software a largo plazo, las cuales se usan con fines estratégicos. (Puglla Remache, y otros, 2004). Estas se caracterizan por:

- El control y ejecución del proyecto.
- Medición de tiempos del análisis, diseño, implementación, implantación y pos implantación.
- Medición de las pruebas (errores, cubrimiento, resultado en número de defectos y número de éxito).
- Medición de la transformación o evaluación del producto.

Métricas del Proyecto

Las métricas de proyecto permiten valorar el estado de un proyecto en curso, así como también rastrear los riesgos potenciales y descubrir las áreas con problema antes que se vuelvan “críticas”. (Puglla Remache, y otros, 2004). Estas se caracterizan por:

- Permitir la evaluación del estado del proyecto.
- Permitir el seguimiento de las pistas de los riesgos.

Las Métricas de Calidad que se utilizarán para Aplicativos SIG son las Métricas del Producto (Definidas por la IEEE y utilizadas por la Dirección de Calidad de Software), ya que permiten considerar las características de los artefactos generados, además de controlar la estabilidad y calidad de los productos durante su desarrollo y evaluar el resultado final, estas métricas son:

1. Eficacia en la Eliminación de Defectos (EED): Medida de la actividad de filtrar de las actividades de aseguramiento de la calidad y de control. Se define de la forma siguiente:

$$EED = E \div (E+D)$$

Donde E es el número de defectos encontrados antes de entregar al usuario final y D es el número de defectos encontrados después de la entrega. El valor ideal de EED es 1, demostrando que no se han encontrado defectos en el software. De forma realista, D será mayor que cero, pero el valor de EED todavía se puede aproximar a 1 cuando E aumenta. En consecuencia cuando E aumenta es probable que el valor final de D disminuya (los errores se filtran antes de que se conviertan en defectos). EED será utilizada dentro de la factoría como un indicador de la destreza de filtrar las actividades de la garantía de la calidad.

Responsable: Administrador de Calidad.

Grupo en el que se realiza: Luego de entregado el producto al cliente. (Ver Anexos)

2. Idoneidad (I): Esta métrica se propone medir cuan adecuada es la función evaluada. Se calcula como:

$$I = 1 - A \div B$$

Donde:

A: Número de funciones en las cuales se detectaron problemas en la evaluación.

B: Número de funciones evaluadas.

El valor obtenido debe estar entre 0 y 1 ($0 \leq X \leq 1$). A mayor cercanía del 1 resultará más adecuada.

Responsable: Administrador de Calidad.

Grupo en el que se realiza: Al terminar la PCN. (Ver Anexo 3)

3. Detección de Errores (DE): Cuantos fallos fueron detectados en las revisiones al producto. Se calcula de la siguiente forma:

$$DE = A \div B$$

Donde:

A: Es el número absoluto de defectos encontrados en las revisiones.

B: Es el número de errores estimados para esta revisión.

El valor de esta métrica debe oscilar entre [0; 1]. De 0 a 0.2 la evaluación será de Satisfactoria, de 0.3 a 0.6 la evaluación será de Aceptable y de 0.7 a 1 la evaluación será de Deficiente.

Responsable: Revisor Líder.

Grupo en el que se realiza: Al terminar las revisiones al Expediente de proyecto.

Artefactos: Lista de Chequeo de la revisión anterior. Lista de Chequeo de la revisión actual.

(Ver Anexo 3).

2.4 Conclusiones

En este capítulo se definió la estrategia de SQA para la Factoría de Software Aplicativos SIG, explicando todas las actividades que se realizarán, como es el Plan de Aseguramiento de la Calidad, el plan de entrenamiento para los miembros de la factoría y la organización de los recursos humanos de la Factoría.

Capítulo 3: Validación de la Propuesta de Solución de la Estrategia de Aseguramiento de la Calidad para la Factoría de Software Aplicativos SIG

3.1 Introducción

En este capítulo se validará la Estrategia de Aseguramiento de la Calidad propuesta para la Factoría de Software Aplicativos SIG, realizando las actividades que se describen en ella y valorando los resultados que arrojaron cada una de estas.

3.2 Estrategia de SQA

Las actividades que se han realizado en la factoría son las siguientes:

- Entrenar el equipo de desarrollo y el grupo de calidad de la Factoría.
- Organizar el trabajo de los Recursos Humanos de la factoría.
- Aplicar el Plan de SQA.

Para aplicar PSQA a la factoría, se escogió entre sus productos, el SIG-Rutas. A continuación se muestra una gráfica donde se representa la medida en que se realizaron las actividades de la estrategia:

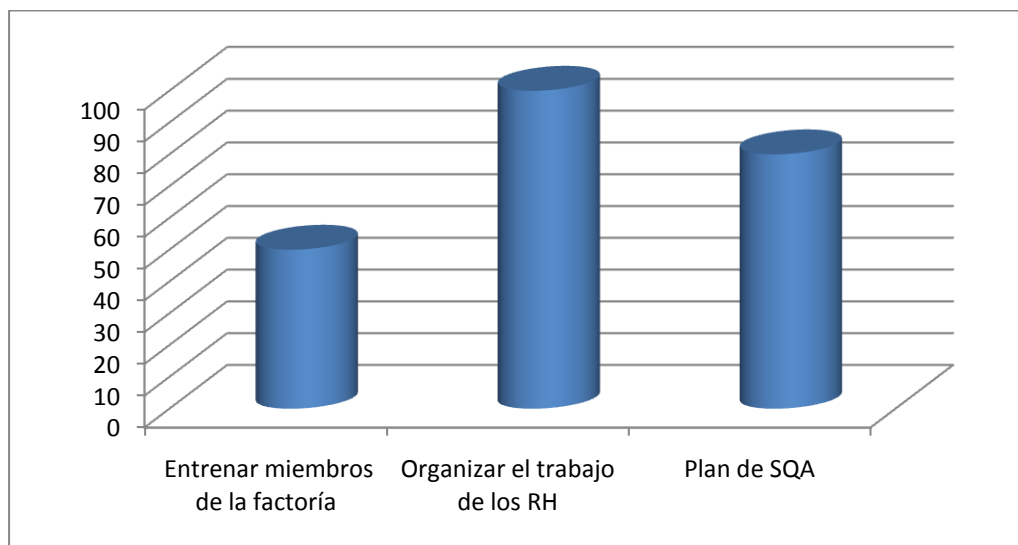


Figura 5: Porcentaje de cumplimiento de la Estrategia de SQA. (Elaboración Propia)

Las actividades de SQA propuestas están siendo aplicadas de acuerdo al grupo que está desarrollando el producto:

- ✓ SIG-Salud: El desarrollo de este producto se está realizando en el grupo de Ingeniería y BD. Las actividades de SQA realizadas hasta el momento son las revisiones al EP y la aplicación de la métrica de DE.
- ✓ SIG-INSMET: Este producto se está desarrollando en el grupo de IC, realizándose las revisiones al EP y a los DCP.
- ✓ SIG-Verano: El desarrollo de este producto está iniciando en el grupo de Ingeniería, por lo que no se le ha realizado ninguna actividad propuesta para el SQA.
- ✓ SIG-INRH: Este producto ya está terminado, pero se está realizando otra versión, por lo que es necesario aplicar la estrategia desde que se comience a desarrollar.
- ✓ SIG-MIC: Se realizó en corto tiempo y aunque se realizaron todas las actividades propuestas por el SQA al estar terminado, ya está siendo liberado por Calisoft.
- ✓ SIG-UCI: Después de realizada la primera versión, se desarrolló una segunda versión, a la que se le están aplicando las actividades de SQA, por las deficiencias que han sido detectadas en esta.
- ✓ SIG-Rutas: Desde sus inicios se le aplicaron todas las actividades propuestas en la estrategia, detectándose y corrigiéndose las deficiencias existentes. La realización de estas actividades permitió que el producto llegase a Calisoft con la menor cantidad de errores posibles, posibilitando de esta manera su liberación. Este producto se escogió como piloto para validar la Estrategia de SQA propuesta.

3.2.1 Organizar y Entrenar el trabajo de los Recursos Humanos

Las actividades de entrenamiento se propusieron con el objetivo de elevar el conocimiento de todos los involucrados en el proceso de desarrollo, y de esta manera contribuir con la excelencia de los productos que se desarrollan en la factoría. Estas actividades fueron realizadas según necesidad de la factoría, entre las actividades cumplidas se encuentran:

- Trabajos Investigativos

Resultados: De los trabajos investigativos propuestos, se les dio cumplimiento a dos, estos fueron el de Roles y Responsabilidades de los miembros de la factoría y del grupo de SQA.

Para la evaluación de esta tarea los Jefes de Grupo de Factoría y el Administrador de Calidad, asignaron a cada uno de sus miembros, los roles que debían investigar, enfocándose en el rol que desempeñaba cada uno. Cada grupo realizó un taller de Roles y Responsabilidades, permitiendo que de esta manera el equipo de trabajo se apoderara del conocimiento necesario para desempeñar su rol. Aunque se puede afirmar que no todos obtuvieron el conocimiento o no se sienten motivados con el rol que desempeñan, ya que algunas de las tareas que deben realizar no tienen calidad o no son entregadas en tiempo.

- Vinculación de Estudiantes

Resultados: Los estudiantes de 3er año que fueron seleccionados para el trabajo en la factoría, fueron incorporados a la producción, con 4 horas de tiempo de máquina, todos los lunes, aunque no están rotando por todos los grupos, sino que fueron integrados a las diferentes factorías, de esta manera ya están vinculados a los grupos que pertenecerán cuando sean vinculados por completo a la producción. Puede que los estudiantes tengan un buen desenvolvimiento trabajando en un único grupo, pero alguno de ellos que no tenga la capacidad necesaria como para formar parte de ese grupo, tendrá problemas en el desarrollo de las actividades que se realicen en el grupo y por tanto el grupo también tendrá problemas con el cumplimiento de las actividades que les son asignadas.

- Talleres

Resultados: Entre los talleres planificados en la factoría, solamente se realizó el de las herramientas del grupo de BD, permitiendo de esta manera que los miembros del grupo estuviesen mejor preparados para el trabajo con estas herramientas. El taller de PHP a los grupos de IU e IC, es fundamental para los miembros, ya que este es el lenguaje de

programación que utiliza la factoría, por lo que su realización permitió un mejor desenvolvimiento de los miembros de estos grupos. Los 2 talleres de Revisiones al grupo de SQA, en el que se trataron todas las actividades que se deben realizar para lograr una buena revisión, permitiendo de esta manera que las revisiones realizadas al grupo de Ingeniería y a las pruebas, estuviesen correctamente realizadas.

Organizar el Trabajo de Recursos Humanos

Luego de organizado el equipo de trabajo, la factoría cuenta con 8 roles. Para lograr una mejor organización, se le dio cumplimiento a la actividad de entrenamiento, que permitió que cada miembro de la factoría conociera exactamente qué actividades debía desempeñar su rol. Además de distribuir los roles según el trabajo que se realiza en cada Grupo de Factoría.

| Grupos de Factoría | Roles | Estudiantes | Profesores |
|---------------------------------|---|-------------|------------|
| Ingeniería | Analista. Planificador. Jefe de Grupo. | 14 | 3 |
| BD | Diseñador de BD. Planificador. Jefe de Grupo. | 13 | 3 |
| IU | Diseñador de IU. Planificador. Jefe de Grupo. | 15 | 3 |
| IC | Desarrollador. Planificador. Jefe de Grupo. Líder de Factoría. | 16 | 4 |
| Gestión | Planificador. Jefe de Grupo. | 1 | 5 |
| Estudiantes 3er año | | 11 | |
| Administrador de Calidad | | 1 | |

Tabla 6: Organización de los Recursos Humanos. (Elaboración Propia)

Actualmente el trabajo en la factoría está organizado y las tareas están orientadas hacia el rol que desempeña cada miembro del equipo de trabajo, siendo este el responsable de que sean

realizadas correctamente y en el tiempo requerido. Aunque muchos de los miembros de la factoría no están cumpliendo con sus responsabilidades, lo que trae como consecuencia que muchas de las actividades no se estén realizando correctamente y demostrando que no todas las actividades de entrenamiento fueron cumplidas o realizadas correctamente.

3.2.2 Actividades del PSQA

Lo propuesto en el PSQA y realizado en la factoría y en el SIG-Rutas, han permitido el perfeccionamiento de los productos que se desarrollan en la factoría:

- La Gestión del SQA, realizada para organizar los roles y las responsabilidades del equipo de calidad, permitieron aplicar la estrategia correctamente.
- Los Estándares, Guías, Herramientas, Técnicas y Metodologías, fortalecieron el cumplimiento de las actividades ya que guiaron todo el trabajo del grupo de calidad.
- Todos los documentos generados en las actividades de SQA se guardaron en el Registro de Calidad.
- Para realizar las actividades de GCS en la factoría, se utiliza la herramienta Subversión, permitiendo mantener la calidad de los procesos de cambios que se realizaron al producto.

3.2.2.1 Revisiones

Revisiones al Expediente de Proyecto

Al SIG-Rutas se le realizaron 3 revisiones a la documentación del Expediente de Proyecto que confecciona el Grupo de Ingeniería, estas revisiones permitieron obtener las NC que existían en estos documentos, además de facilitar a los analistas, según las recomendaciones hechas, la solución de las NC.

Comparando las revisiones realizadas al Expediente de Proyecto se notó que la primera y la segunda iteración tenían la misma cantidad de NC (17), la tercera iteración arrojó mejores resultados que las anteriores, con 5 NC. Lo que demuestra que el equipo de trabajo de la factoría no solucionó las NC detectadas en la primera revisión. En la tercera revisión se

comprobó que se esforzaron para eliminar las deficiencias detectadas, aunque la documentación aún no está libre de errores.

Revisiones a los DCP

En las revisiones realizadas a los DCP se detectó una sola NC. En esta actividad la evaluación fue de Satisfactoria, demostrando de esta manera que el equipo de trabajo está bien preparado para realizar estas actividades. Con estas revisiones se pudo constatar que hubo un gran avance en el trabajo que se realiza en la factoría con lo relacionado a la documentación del Expediente de Proyecto.

3.2.2.2 Pruebas

Las pruebas propuestas a realizar son la de Unidad, con la técnica de PCB y la de Sistema con la técnica de PCN. En la primera iteración de técnica de PCN se detectaron 11 NC, en la 2da se detectaron 4 NC, con una evaluación de Aceptable y en la tercera se detectó 1 sola deficiencia.

Esta prueba permitió detectar las diferentes deficiencias que tenía el producto SIG-Rutas en su interfaz. Aunque en la primera y la segunda iteración se detectaron varias NC, la tercera y última iteración se demostró que el trabajo realizado para eliminar las deficiencias encontradas en el producto fue muy bueno.

3.2.2.3 Métricas

Métrica de DE para las revisiones efectuadas al Expediente de Proyecto

Segunda iteración de la revisión:

$$DE = A \div B$$

$$DE = 17 \div 17$$

$$DE = 1$$

La métrica se aplicó a la segunda iteración de la revisión y el valor obtenido es 1, lo que demuestra que el trabajo de la factoría en esta revisión estuvo mal, trayendo como consecuencia que en esas dos revisiones la factoría adquiriese Deficiente en las evaluaciones finales de las revisiones.

Tercera iteración de la revisión:

$$DE = A \div B$$

$$DE = 17 \div 17$$

$$DE = 0.2$$

La métrica aplicada a la tercera iteración arrojó un mejor resultado, teniendo una evaluación de Satisfactorio. Demostrando de esta manera que el trabajo en la factoría fue mucho mejor que en las primeras dos revisiones.

Métrica de Idoneidad para medir cuan adecuada es la función evaluada en la PCN

Primera iteración:

$$I = 1 - A \div B$$

$$I = 1 - 11 \div 13$$

$$I = 0.15$$

Segunda iteración:

$$I = 1 - A \div B$$

$$I = 1 - 11 \div 13$$

$$I = 0.69$$

Tercera iteración:

$$I = 1 - A \div B$$

$$I = 1 - 11 \div 13$$

$$I = 0.92$$

Las mediciones a las pruebas arrojaron diferentes resultados. En la primera iteración se pudo constatar que las funcionalidades evaluadas no eran adecuadas, ya que el resultado de la medición fue de 0.15. Para la segunda iteración el resultado fue de 0.69 y para la tercera

iteración el resultado fue de 0.92 y aunque el producto no era completamente adecuado el resultado en esta iteración demostró que el trabajo realizado para eliminar los problemas de las funcionalidades fue satisfactorio.

3.3 Conclusiones

Durante la aplicación de la estrategia, se pudo constatar que la estrategia de SQA sí puede ser aplicada a los productos de la factoría. Con la correcta aplicación de cada una de las actividades propuestas para el SQA se logrará que los productos de Aplicativos SIG sean liberados satisfactoriamente.

Conclusiones Generales

Durante el desarrollo de la investigación se le dio cumplimiento al objetivo general ya que:

- ✓ Para la definición de la estrategia se realizó un estudio de los diferentes enfoques del Aseguramiento de la Calidad según los modelos, normas y estándares de calidad.
- ✓ Se definió la Estrategia de SQA según las características de la Factoría de Software Aplicativos SIG.
- ✓ Se validó la estrategia propuesta aplicándola al producto SIG-Rutas de la Factoría de Software Aplicativos SIG.
- ✓ Se registraron los resultados obtenidos, valorando la importancia de la estrategia en este trabajo, ya que contribuirá a elevar la eficiencia del proceso de desarrollo del software empleado en la Factoría de Software Aplicativos SIG.
- ✓ El Aseguramiento de la Calidad en la factoría de software representa un factor imprescindible dentro del proceso de desarrollo del producto, asegurando que este cumpla con todas las expectativas del cliente.

Recomendaciones

Luego de aplicada la estrategia se recomienda:

- ✓ Darle seguimiento a la todas las actividades que plantea la estrategia de SQA mientras exista la Factoría de Software Aplicativos SIG.
- ✓ Aplicar la estrategia desde el inicio a todos los productos que se desarrollan en la factoría.
- ✓ Estudiar y Analizar otras métricas que sean de posible aplicación en la factoría.
- ✓ Aplicar la métrica de Eficacia en la Eliminación de Defectos.
- ✓ Estudiar y Analizar otras pruebas que sean de posible aplicación en la factoría.
- ✓ Realizar nuevas actividades de entrenamiento, según las necesidades que vayan surgiendo en la factoría o en los grupos de factoría.

Bibliografía

- Ana M. López, Cesar Cabrera, Luz E. Valencia Ayala. 2008.** *Introducción a la Calidad de Software*. Colombia : Universidad Tecnológica de Pereira., 2008. ISSN 0122-1701.
- Asociación de Técnicos de Informática. 2009.** Asociación de Técnicos de Informática. [En línea] 9 de Marzo de 2009. [Citado el: 15 de Diciembre de 2010.] <http://www.ati.es/spip.php?article1135>.
- 2010-2011.** Bill Gates y el juicio. *Entorno Virtual de Aprendizaje*. [En línea] 2010-2011. [Citado el: 2011 de Enero de 16.] <http://eva.uci.cu/mod/resource/view.php?id=9288>.
- Calisoft. 2009.** Calisoft. *PROGRAMA DE MEJORA*. [En línea] 2009. [Citado el: 18 de Enero de 2011.] http://calisoft.uci.cu/index.php?option=com_content&view=article&id=46&Itemid=27.
- Casal, Julio. 2005.** *Desarrollo de Software Basado en Componentes*. s.l. : Colabora .Net, 2005.
- Coba Santiesteban, Yanet. 2008.** *Propuesta de Plan de Aseguramiento de la Calidad del Proyecto Servicios Comunitarios*. La Habana. Cuba : s.n., 2008.
- Corporation, Ideal Network. 2010.** IdealNetworkCorp. [En línea] 2010. [Citado el: 07 de Febrero de 2011.] <http://idealnetcorp.com/index.php/es/portafolio-de-servicios/aseguramiento-de-calidad>.
- CRS IT CONSULTING. 2009.** CRS IT CONSULTING. *CRS IT CONSULTING*. [En línea] 2009. [Citado el: 25 de Mayo de 2011.] <http://www.crs-itconsulting.com>.
- Cueva Lovell, Juan Manuel. 1999.** *Calidad del Software*. España : s.n., 1999.
- . 1999. *Calidad del Software*. España : Departamento de Informática Universidad de Oviedo, 1999.
- Departamento de Control de Calidad y Auditoría. 2000.** *Control de Calidad en los Sistemas*. 2000.
- Díaz, Cisneros Luis R y Candeaux, Rafael. 1994.** *Los Sistemas de información Geográfica SIG*. Madrid : Asociación Española para la Enseñanza de las Ciencias de la Tierra (AEPECT), 1994. ISSN: 1.131.9.100.
- Díaz, Oscar y Trujillo, Salva. 2010.** *LÍNEAS DE PRODUCTO SOFTWARE*. País Vasco : Ra-Ma, 2010.
- Fábrica Académica de Software. Avalos, Zalatiel Carranza. 2008.* 3, Lima : Revista digital de la Facultad de Ingeniería de Sistemas, 2008.
- FÁBRICAACADÉMICA DE SOFTWARE. Carranza Avalos, Zalatiel. 2008.* 3, Lima : Revista digital de la Facultad de Ingeniería de Sistemas, 2008, Vols. 7-22.
- García, Joaquin. 2005.** IngenieroSoftware. [En línea] 14 de Agosto de 2005. [Citado el: 21 de Enero de 2011.] <http://www.ingenierosoftware.com/calidad/cmm-cmmi.php>.

Guillermo Domínguez, Luz Lozano. 2003. El concepto de calidad y su evolución. [aut. libro] Colectivo de Autores. *Calidad y Formación: Binomio inseparable*. 2003.

2010-2011. Historia de Microsoft. *Entorno Virtual de Aprendizaje*. [En línea] 2010-2011. [Citado el: 15 de Enero de 2011.] http://eva.uci.cu/mod/resource/view.php?id=9287&subdir=/Temas_Generales.

Hoz, Yanosky Rios La. 2005. *Modelo Funcional de la Fábrica de Software de la UCI para la línea Carrefour*. La Habana. Cuba : s.n., 2005.

Informáticas, universidad de las Ciencias. 2009. Calisoft. *PROGRAMA DE MEJORA*. [En línea] 2009. [Citado el: 18 de Enero de 2011.] http://calisoft.uci.cu/index.php?option=com_content&view=article&id=46&Itemid=27.

ISO. 2002. *ISO 19011: 2002 Auditoría de Gestión de Calidad y/o Ambiental*. 2002.

Kan, Stephen H. 2002. *Metrics and Models in Software Quality Engineering, Second Edition*. s.l. : s.l. Adison Wesley, 2002, 2002. 0-201-72915-6.

Lazo Ochoa, Rene y Muro Fumero, Dorisbel. 2010. *“Polos productivos y los modelos de desarrollo basados en enfoques industriales. Propuesta de marco de Fábricas de Software”*. 2010.

Los Sistemas de información Geográfica SIG. Díaz Cisneros, Luís R. y Candeaux Duffatt, Rafael. 1994. Madrid : Asociación Española para la Enseñanza de las Ciencias de la Tierra (AEPECT), 1994. ISSN: 1.131-9.100.

Malevski, Yoram. 1995. *Manual de gestión de la calidad total a la medida*. Guatemala : Piedra Santa, 1995.

Minguet Melián, Jesus M y Hernández Ballesteros, Juan Francisco. 2003. *La calidad del software y su medida*. España : CENTRO DE ESTUDIOS RAMON ARECES, S.A., 2003. M. 44.635-2003.

Montilva, Jonás A. 2006. *Desarrollo de Software Basado en Líneas de Productos de Software*. Mérida-Venezuela : s.n., 2006.

Padilla, Gabriel. 2002. GestioPolis. [En línea] 02 de 2002. [Citado el: 05 de 11 de 2010.] <http://www.gestiopolis.com/recursos/documentos/fulldocs/ger/aucaid.htm>.

Pantoja Zaldívar, Yoenis. 2010. *Modelo para la concesión de una línea de productos de software para el desarrollo de Sistemas de Información Geográfica sobre la base de la Plataforma GeneSIG*. La Habana. Cuba : s.n., 2010.

Pello, Javier. 2010. SoftQaNetwork. *SoftQaNetwork*. [En línea] 3 de 7 de 2010. [Citado el: 1 de 12 de 2010.] <http://www.softqanetwork.com/philip-crosby-y-la-calidad-del-software>.

Perry, William E. 1983. *“Effective methods of EDP quality”*. Englewood Cliffs, N.J : Prentice_Hall, 1983.

Pressman, Roger S. 2005. *Ingeniería de Software. Un enfoque práctico. Quinta Edición*. 2005.

2010. Profesional.com. [En línea] 20 de Octubre de 2010. [Citado el: 16 de Enero de 2011.]
<http://management.iprofesional.com/notas/>.

2009. PROGRAMA DE MEJORA. Libro de Proceso para PPQA. *Calisoft*. [En línea] 2009. [Citado el: 18 de Enero de 2011.] http://calisoft.uci.cu/index.php?option=com_content&view=article&id=46&Itemid=27.

Puglla Remache, Gabriela Noemí y León Quiñones, Lorena del Cisne. 2004. *Métricas de Proceso y Proyecto*. Ecuador : Métricas de Proceso y Proyecto, 2004.

Rivero Canino, Lanny y Rodríguez Martínez, Meydi Elena. 2088. *Estrategia de Aseguramiento de la calidad para el Simulador Quirúrgico*. Ciudad de la Habana. Cuba : UCI, 2088.

Ruilova Rojas, María Esther. 2008. *Informe Ejecutivo. Métricas del Producto para el Software*. 2008.

Sánchez Fornaris, Maité. 2009. *Propuesta de Guia de métricas para evaluar el desarrollo de los Sistemas de Información Geográfica*. 2009.

Scalone, Fernanda. 2006. *Estudio Comparativo de los Modelos y Estándares de Calidad del Software*. Buenos Aires : s.n., 2006.

Sené, María Luisa. 1997. *PRESENCIA DE LAS NORMAS ISO 9000 EN LA CALIDAD DE LOS SERVICIOS*. 1997.

2010. SHVOONG.com. [En línea] 20 de Diciembre de 2010. [Citado el: 19 de Enero de 2011.]
<http://es.shvoong.com/humanities/>.

Trujillo, Casañola Yaimí. 2009. *Modelo de Factoría de Software Aplicando inteligencia*. La Habana. Cuba : s.n., 2009.


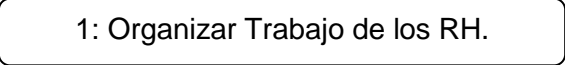
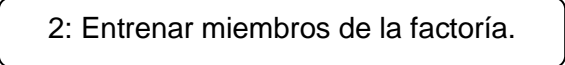

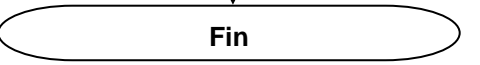
Una Propuesta de introducción de las Revisiones en el Proceso de desarrollo de Software. **Delgado Dapena, Martha Dunia. 2005.** 2, Ciudad de La Habana, Cuba : s.n., 2005, Vol. 26.

Universidad Católica de Chile. Departamento de Ciencia de la Computación. 2002. *Fábrica de Software. Documento de Proceso de la Gerencia de SQA*. Chile : s.n., 2002.

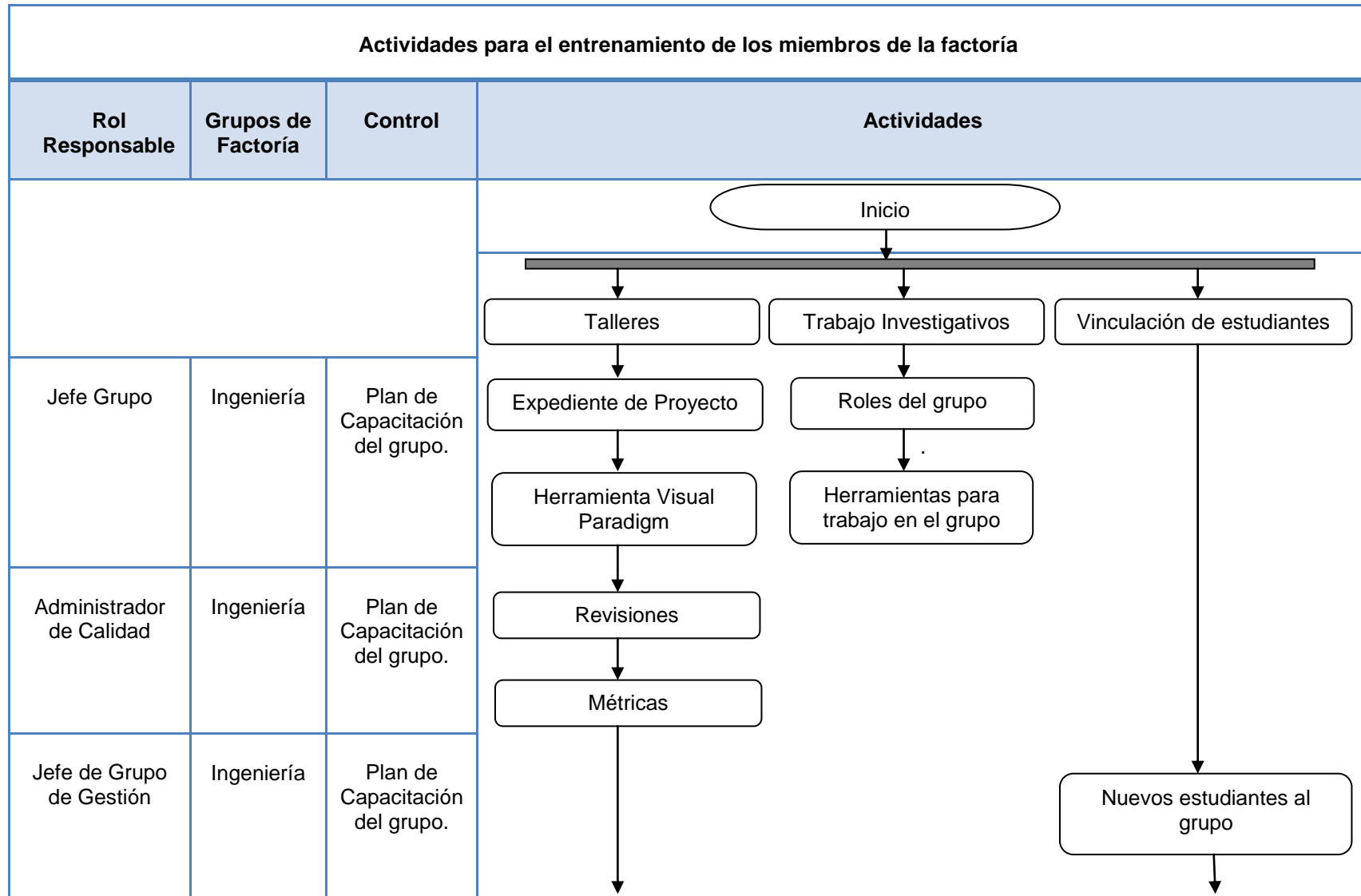
Vega Lebrúm, Carlos, Rivera, Laura Susana y García Santillán, Arturo. 2008. *MEJORES PRÁCTICAS PARA EL ESTABLECIMIENTO Y ASEGURAMIENTO DE LA CALIDAD DE SOFTWARE*. Veracruz : Universidad Cristobal Colón, 2008. 02.

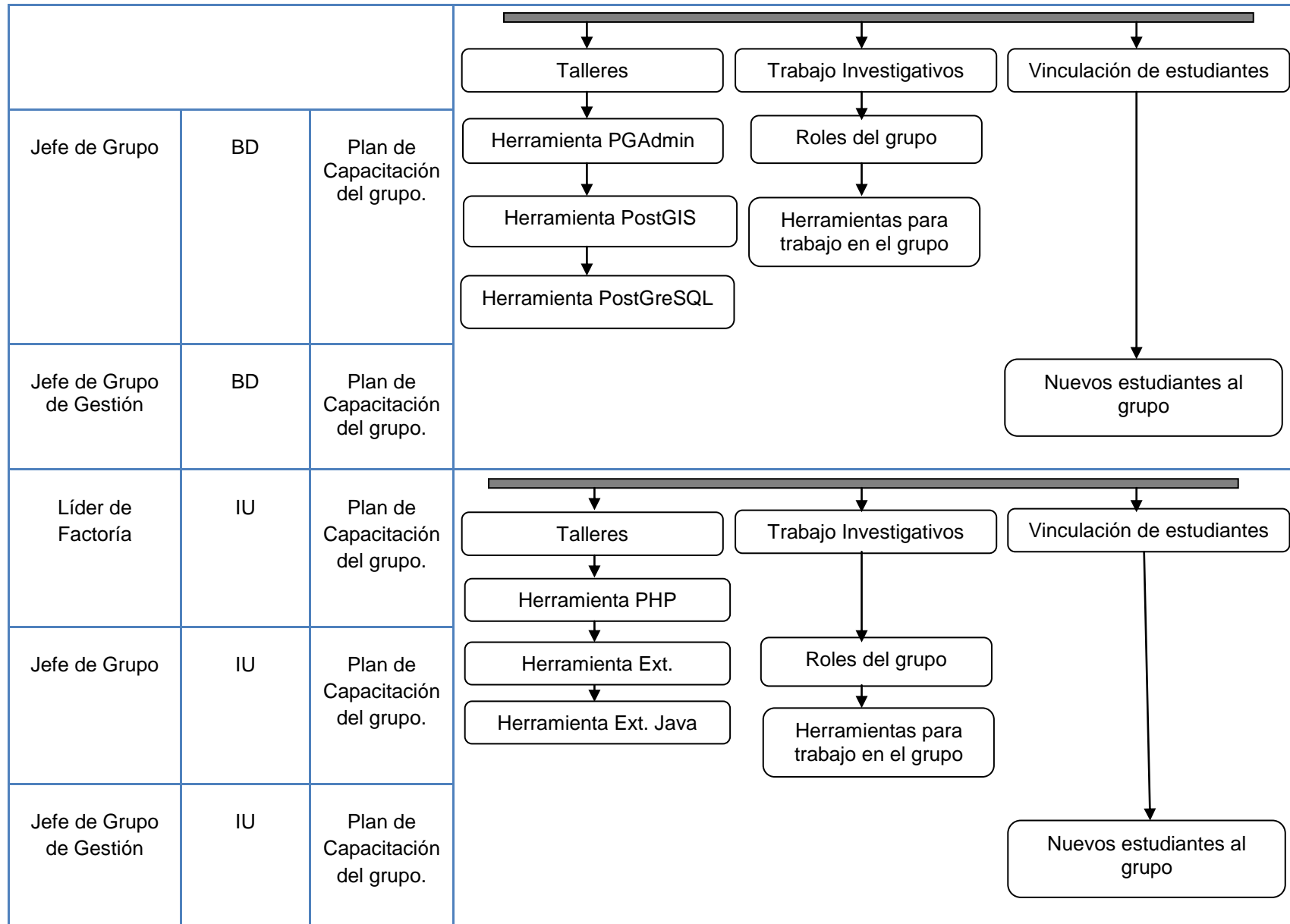
Anexos

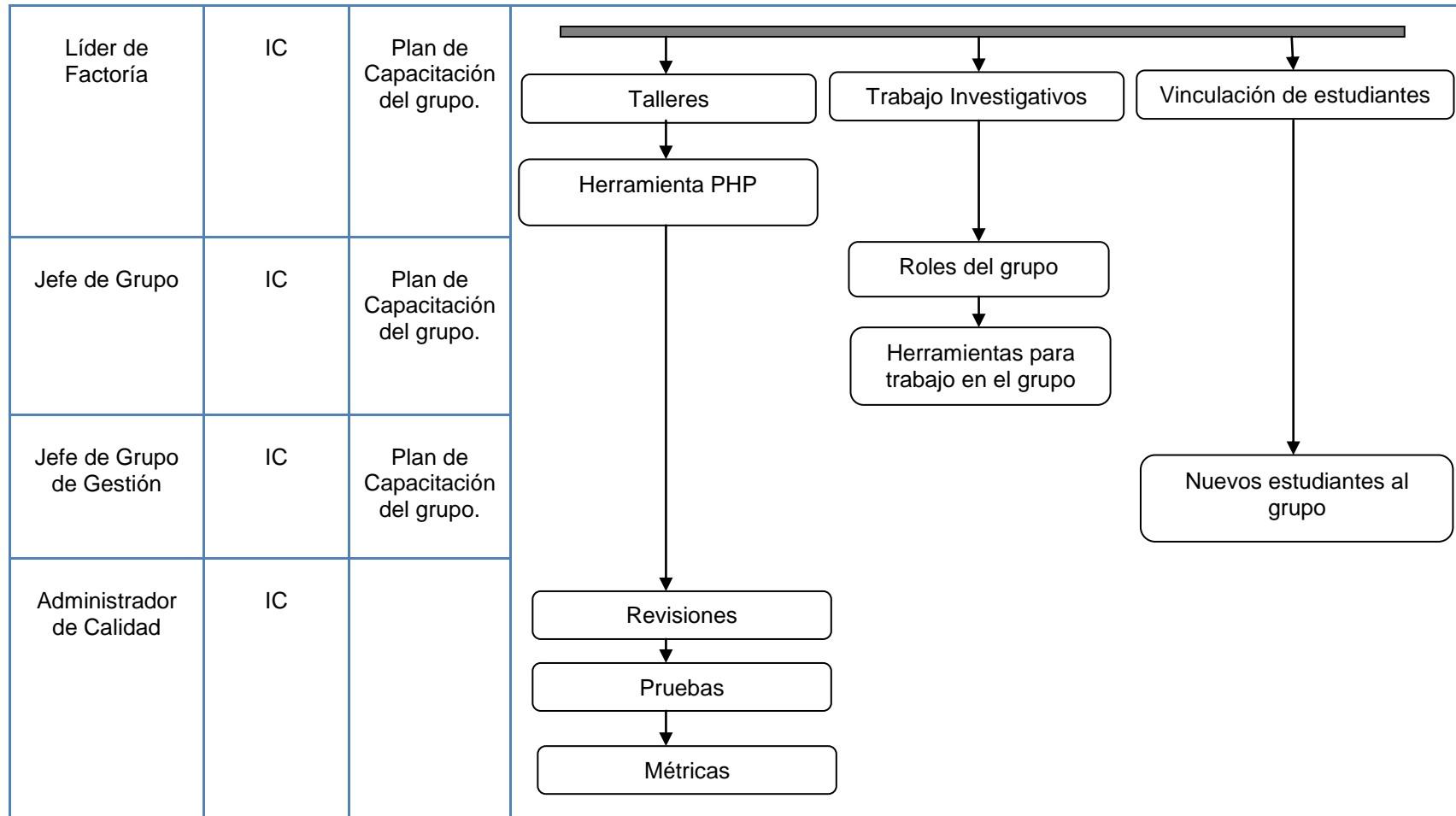
Anexo 1

| Estrategia de SQA | | | |
|--------------------------|----------|--|---------------------------------|
| Rol Responsable | Grupo | Actividad | Salida |
| | |  <p>Inicio</p> | |
| Administrador de Calidad | Factoría |  <p>1: Organizar Trabajo de los RH.</p> | Doc. Roles y Responsabilidades. |
| Administrador de Calidad | Factoría |  <p>2: Entrenar miembros de la factoría.</p> | Doc. Plan de Capacitación. |
| Administrador de Calidad | Factoría |  <p>3: Realizar Plan de SQA.</p> | Doc. Plan de SQA |
| | |  <p>Fin</p> | |

Anexo 2



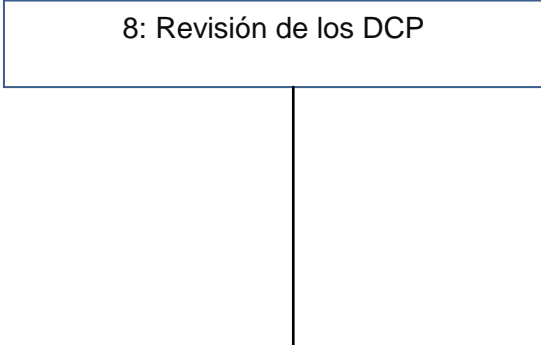
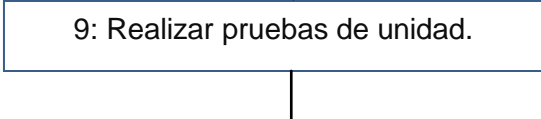
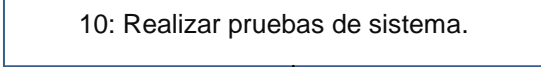
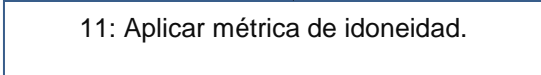
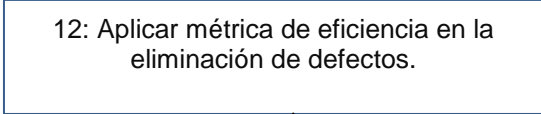





Anexo 3

| Actividades del Plan de SQA | | | | |
|-----------------------------|--------------------|---|--|---|
| Rol Responsable | Grupos de Factoría | Entrada | Actividades | Salida |
| | | | <p>Inicio</p> | |
| Administrador de Calidad. | Factoría | | <p>1: Definir objetivos de calidad.</p> | |
| Administrador de Calidad. | Factoría | | <p>2: Gestionar el SQA.</p> | Doc. Roles y Responsabilidades del SQA. |
| Administrador de Calidad. | Factoría | Lineamientos de Calidad. Modelo CMMI. Norma ISO 9000. | <p>3: Definir estándares y guías.</p> | |
| Administrador de Calidad. | Factoría | Lista de Chequeo. Ayuda de RUP. Expediente de Proyecto. | <p>3: Seleccionar herramientas, técnicas y metodologías.</p> | |

| | | | | |
|---|---|--|--|--|
| Administrador de Calidad. | Factoría | | <div style="border: 1px solid black; padding: 5px; text-align: center;">4: Crear registro de calidad.</div> | |
| Líder de Factoría. Jefes de Grupo. Administrador de Calidad. | Factoría | Plan de GC. Pedido de Cambio. Orden de Cambio. Lista de Chequeo. | <div style="border: 1px solid black; padding: 5px; text-align: center;">5: Gestión de Configuración de Software (GCS).</div> | Doc. Control de Versiones. Doc. Control de cambios. Informe de autoridad de Control de Cambio. Lista de Chequeo. Lista de Recomendaciones. |
| Revisor Líder. Equipo Revisor. Administrador de Calidad. | Ingeniería | Expediente de Proyecto. Lista de Chequeo. | <div style="border: 1px solid black; padding: 5px; text-align: center;">6: Revisar el expediente de proyecto.</div> | Acta de reunión Apertura. Lista de Chequeo. Lista de Recomendaciones. Informe Final. Acta de Reunión de Cierre. |
| Revisor Líder. | Al terminar la revisión a la documentación. | Lista de Chequeo de la revisión anterior. Lista de Chequeo de la revisión actual. | <div style="border: 1px solid black; padding: 5px; text-align: center;">7: Aplicar métrica de detección de errores.</div> | Informe con el resultado de la métrica y la valoración del resultado. |

| | | | | |
|--|---|--|---|---|
| <p>Revisor Líder.</p> <p>Equipo Revisor.</p> <p>Administrador de Calidad.</p> | <p>IC</p> | <p>Diseño de de Caso de Prueba.</p> <p>Lista de de Chequeo de Casos de Prueba.</p> |  <pre> graph TD A[8: Revisión de los DCP] --> B[9: Realizar pruebas de unidad.] B --> C[10: Realizar pruebas de sistema.] C --> D[11: Aplicar métrica de idoneidad.] D --> E[12: Aplicar métrica de eficiencia en la eliminación de defectos.] E --> F([Fin]) </pre> | <p>Acta de reunión Apertura.</p> <p>Lista de Chequeo de Casos de Prueba.</p> <p>Lista de Recomendaciones.</p> <p>Informe Final.</p> <p>Acta de Reunión de Cierre.</p> |
| <p>Desarrollador.</p> | <p>IC</p> | <p>Código de la Aplicación SIG.</p> |  | <p>Informe Final de evaluación a las Revisiones.</p> |
| <p>Probador.</p> | <p>IC</p> | <p>DCP</p> |  | <p>Registro de No Conformidad.</p> |
| <p>Administrador de Calidad.</p> | <p>Al terminar la PCN</p> | |  | <p>Informe con el resultado de la métrica y la valoración del resultado.</p> |
| <p>Administrador de Calidad.</p> | <p>Luego de entregado el producto al cliente.</p> | |  | <p>Informe sobre los errores encontrados en el software.</p> |
| | | |  | |

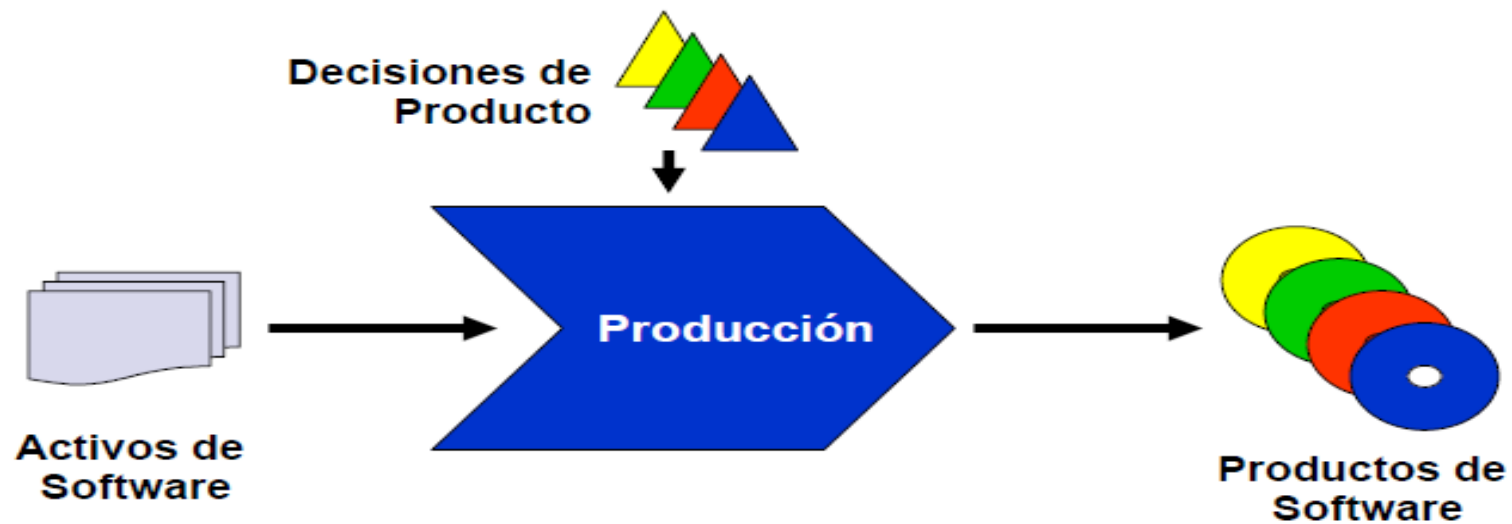
Anexo 4

| Entrevista a miembros de la Factoría Aplicativos SIG | | | | | | |
|---|--|-----------------------------|---------------------|---------------------|---------------------|--------------------------|
| Preguntas | Roles seleccionados para la entrevista | | | | | |
| | Líder de Factoría | Jefe de Grupo de Ingeniería | Jefe de Grupo de BD | Jefe de Grupo de IU | Jefe de Grupo de IC | Jefe de Grupo de Gestión |
| ¿Qué metodología de desarrollo utiliza la factoría? | X | | | | | |
| ¿Cómo está organizado el trabajo en la factoría? | X | | | | | |
| ¿Porque dividir la factoría en grupos? | X | | | | | |
| ¿Cuántos productos están en desarrollo en estos momentos? | X | | | | | |
| ¿Quiénes son los Jefes de cada grupo? | X | | | | | |
| ¿Cuántos roles hay definidos? | X | | | | | |
| ¿Cuántos estudiantes hay en la | | | | | | X |

| | | | | | | |
|---|--|---|---|---|---|---|
| factoría? | | | | | | |
| ¿Cuántos profesores hay en la factoría? | | | | | | X |
| ¿Cómo están distribuidos los estudiantes y los profesores en los grupos? | | | | | | X |
| ¿Tienen definido un Plan de Capacitación para los miembros de la factoría? | | | | | | X |
| ¿Qué gestiona el grupo? | | | | | | X |
| ¿Qué otras actividades realizan? | | | | | | X |
| ¿Cuántos estudiantes y profesores hay en su grupo? | | X | X | X | X | X |
| ¿Cuáles son las responsabilidades de los roles definidos en su grupo? | | X | X | X | X | X |
| ¿Qué función debe realizar el grupo para el desarrollo de un producto en la factoría? | | X | X | X | X | X |

| | | | | | | |
|---|--|---|---|---|---|---|
| ¿Qué herramientas utilizan para el trabajo del grupo? | | X | X | X | X | X |
| ¿Qué son las actividades cartográficas? | | | X | | | |

Anexo 5



Modelo Básico de una LPS. (Díaz, y otros, 2010)