



Universidad Nacional de Costa Rica

Sistema de Estudios de Posgrado

**Maestría en Administración de Tecnologías de la
Información (MATI)**

Énfasis en Administración de Proyectos

**Diseño de una propuesta de implementación del
departamento de SQA en una empresa de software**

Estudiante: Pamela Green Arias

Heredia, Costa Rica

Noviembre, 2023

DECLARACIÓN JURADA DE RESPETO AL DERECHO DE AUTOR

20 de noviembre del año 2023

Universidad Nacional

Facultad de Ciencias Exactas y Naturales

Escuela de Informática

Posgrado en Gestión de la Tecnología de Información y Comunicación (ProGestIC)

<p>FORMULARIO DE DEPÓSITO LEGAL, AUTORIZACIÓN DE USO DE DERECHOS PATRIMONIALES DE AUTOR E INCORPORACIÓN A REPOSITARIOS INSTITUCIONALES DE INFORMACIÓN DE ACCESO PÚBLICO</p>
--

La persona abajo firmante, en condición de estudiantado de la Maestría en Tecnologías de la Información (MATI) con Énfasis en Administración de Proyectos y autor del Trabajo final de graduación titulado:

Diseño de una propuesta de implementación del departamento de SQA en una empresa de software

Para optar al grado académico de Máster en:

Tecnologías de la Información con Énfasis en Administración de Proyectos

De conformidad con lo establecido en el documento de "Lineamientos generales para la realización del trabajo final de graduación" y demás normativa universitaria relacionada con estos trabajos de graduación, DECLARO BAJO FE DE JURAMENTO

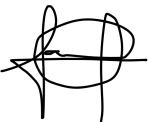
conociendo la responsabilidad civil, penal o administrativa en que podría incurrir al no decir la verdad, lo siguiente:

1. El documento, producto, obra audiovisual, software, resultado del trabajo final de graduación referido anteriormente es original, inédito y ha cumplido con todo el proceso de aprobación académico que confiere el grado académico postulado con esta obra.
2. El trabajo final de graduación referido anteriormente constituye una producción intelectual propia de la persona abajo firmante y a esta fecha no ha sido divulgado a terceros(as) de forma pública, por ningún medio de difusión impreso o digital.
3. Autorizo el depósito de un ejemplar en formato impreso y otro en formato digital (entregado en soporte de disco compacto), en la colección de trabajos finales de graduación del ProGesTIC de la Universidad Nacional, así como la realización de copias electrónicas adicionales para fines exclusivos de seguridad y conservación de la información.
4. En caso de que el trabajo final de graduación haya sido elaborado como obra en colaboración -bien se trate de obras en las que los autores(as) tienen el mismo grado de participación o aquellas en las que existe una persona autora principal y una o varias personas autoras secundarias-, todos(as) ellos(as) han contribuido intelectualmente en la elaboración del documento y en este acto, libero de responsabilidad a las autoridades del posgrado y al personal administrativos que custodian la colección del ProGesTIC, en relación con el reconocimiento que se realiza respecto de los niveles de participación asignados por el propio autor del TFG.
5. En caso de que el trabajo final de graduación haya sido elaborado como obras en colaboración (conforme a lo dispuesto en el punto 4), el autor abajo firmante designa a Pamela Green Arias como encargado(a) de recibir comunicaciones y representar con autoridad suficiente a los suscritos, en condición de agente autorizado(a) de los demás autores(as).
6. Reconozco que la colección de trabajos finales del ProGesTIC no emite criterios ni valoraciones académicas sobre lo planteado en el producto final del trabajo de graduación y autorizo a esta dependencia para que proceda a poner a

disposición de público la obra en mención, a través de los espacios físicos o virtuales que se posea, así como a través del Repositorio Institucional; a partir del cual los usuarios de dichas plataformas puedan acceder al documento y hacer uso de este en el marco de los fines académicos, no lucrativos y de respeto a la integridad del contenido del mismo, así como la mención del autor o poseedor de sus derechos.

7. Manifiesto que todos los datos de citas dentro de texto y sus respectivas referencias bibliográficas, así como las tablas y figuras (ilustraciones, fotografías, dibujos, mapas, esquemas u otros) tienen la fuente y el crédito debidamente identificados y se han respetado los derechos de autor.
8. Autorizo la licencia gratuita no exclusiva de los derechos patrimoniales de autor para reproducir, traducir, distribuir y poner a disposición pública en formato electrónico, el documento depositado, para fines académicos, no lucrativos y por plazo indefinido en favor de la Universidad Nacional, que incluye además los siguientes actos:
 - a. La publicación y reproducción íntegra de la obra o parte de esta, tanto por medios impresos como electrónicos, incluyendo Internet y cualquier otra tecnología conocida o por conocer.
 - b. La traducción a cualquier idioma o dialecto de la obra o parte de esta.
 - c. La adaptación de la obra a formatos de lectura, sonido, voz y cualquier otra representación o mecanismo técnico disponible, que posibilite su acceso para personas no videntes parcial o totalmente, o con alguna otra forma de capacidades especiales que le impida su acceso a la lectura convencional del TFG.
 - d. La distribución y puesta a disposición de la obra al público, de tal forma que el público pueda tener acceso a ella desde el momento y lugar que cada quien elija, a través de los mecanismos físicos o electrónicos de que disponga.
 - e. Cualquier otra forma de utilización, proceso o sistema conocido o por conocerse que se relacione con las actividades y fines académicos a los cuales se vincula la maestría, la colección de trabajos finales del ProGesTIC, la Escuela de Informática y la Universidad Nacional.

9. Reconozco que la colección de trabajos del ProGestIC manifiesta actuar con diligencia para evitar la existencia en su sitio web de contenidos ilícitos y en caso de que tenga conocimiento efectivo de la existencia de infracciones a los derechos de propiedad intelectual, se reserva el derecho de proceder a bloquear el acceso durante el trámite del debido proceso para comprobar el incumplimiento y en caso de verificarse la falta, retirar definitivamente el acceso al TFG depositado.
10. Acepto que la publicación y puesta a disposición del público del trabajo final de graduación, así como la presente autorización de uso de la obra, se regirá por la normativa institucional de la Universidad Nacional y la legislación de la República de Costa Rica. Adicionalmente, en caso de cualquier eventual diferencia de criterio o disputa futura, acepto que esta se dirimirá de acuerdo con los mecanismos de Resolución Alternativa de Conflictos y la Jurisdicción Costarricense.

Autor:	Pamela Green Arias
Fecha de entrega:	20 de noviembre del 2023
Correo:	Pamela.green.arias@gmail.com
Firma:	

Índice General

Índice de Figuras	3
Índice de Tablas	3
Resumen ejecutivo	4
Capítulo I. El problema y su importancia	5
1.1 Título del tema de investigación.....	6
1.2 Antecedentes.....	6
1.3 Justificación.....	7
1.4 Problema	7
1.5 Objetivo general	8
1.6 Objetivos específicos.....	8
1.7 Metas por alcanzar por objetivo	9
Capítulo II. Marco teórico o referencial	10
2.1 Conceptos de calidad de software.....	11
2.1.1 ¿Qué es calidad?.....	11
2.1.2 ¿Qué es aseguramiento de calidad y control de calidad?	12
2.1.3 ¿Qué son pruebas de software?	14
2.2 La organización y el departamento de SQA	15
2.3 El proceso del aseguramiento de la calidad de software	16
2.4 Roles y responsabilidades del departamento de SQA	19
2.5 Organizaciones, institutos, estándares, comités y guías.....	21
Capítulo III. Marco metodológico	24
3.1 Enfoque de la investigación.....	25
3.2 Tipo de la investigación	25
3.2.1 Revisión de literatura	25
3.2.2 Entrevistas	28
3.2.3 Observación.....	29
3.2.4 Evaluación par	31
3.3 Fuentes y sujetos de información	31
3.3.1 Fuentes.....	31
3.3.2 Sujetos.....	32
3.4 Población y muestra.....	32
Capítulo IV. Diagnóstico y análisis de resultados	33
4.1 Revisión de literatura	34
4.2 Entrevistas	35

4.2.1 Datos demográficos	35
4.2.2 Estructura del departamento de SQA.....	36
4.2.3 Perfil del personal del departamento de SQA	37
4.3 Observación.....	40
Capítulo V. Solución del problema	45
5.1 Propuesta de Solución.....	46
5.2 Proceso de prueba y resultados	47
Capítulo VI. Análisis financiero.....	50
6.1 Beneficios.....	52
6.2 Costos	53
6.2.1 Tiempo.....	53
6.2.2 Recurso humano.....	53
6.2.3 Hardware y software	54
6.2.4 Costos fijos	54
Capítulo VII. Conclusiones y recomendaciones	56
7.1 Conclusiones	57
7.2 Recomendaciones.....	59
Capítulo VIII. Análisis retrospectivo	61
Bibliografía.....	63
Apéndice	67
Apéndice 1: Preguntas para la entrevista	67
Apéndice 2: Instrumento de revisión par	69
Apéndice 3: Propuesta de implementación.....	71
Apéndice 4: Revisión de José Chacón	72
Apéndice 5: Revisión de Mario Cordero	74
Anexos.....	77
Anexo 1: Respuestas de las entrevistas.....	77
Anexo 2: Anotaciones de las observaciones	94

Índice de Figuras

Figura 1. Aseguramiento de la calidad.....	14
Figura 2. Ejemplo de ficha de trabajo llena.	27
Figura 3. Formato de anotación para la observación.	30
Figura 4. Análisis de patrones para la estructura del departamento de SQA.....	36
Figura 5. Análisis de patrones personal del departamento de SQA	37
Figura 6. Análisis de patrones de cómo implementar el departamento de SQA.	38
Figura 7. Análisis de patrones de cómo evaluar al departamento de SQA.....	38
Figura 8. Análisis de patrones de cómo desarrollar el departamento de SQA.....	39
Figura 9. Análisis de patrones de consideraciones adicionales	40

Índice de Tablas

Tabla 1. Metas por alcanzar por objetivo	9
Tabla 2. Subcategorías de la gestión de la calidad	13
Tabla 3. Responsabilidades.....	20
Tabla 4. Matriz de síntesis	26
Tabla 5. Costo de recurso humano.....	54
Tabla 6. Costos de hardware y software.....	54
Tabla 7. Detalle de los principales costos	55

Resumen ejecutivo

La creciente complejidad en el desarrollo de software ha incrementado la necesidad de asegurar la calidad de este. El desconocimiento y la poca relevancia que se le ha dado a este tema ha llevado a la ausencia de departamentos de Aseguramiento de la Calidad de Software conocido en inglés como Software Quality Assurance (SQA) en algunas empresas de desarrollo de software. Generando grandes desafíos tanto para la organización como para el usuario final. La implementación de un departamento de Aseguramiento de la Calidad de Software permite sentar las bases para un desarrollo de software orientado a la calidad. Dicho enfoque se traduce directamente en la mejora de la satisfacción del cliente y el posicionamiento competitivo de la empresa en el mercado.

El objetivo general de este trabajo final de graduación fue el diseño de una propuesta de implementación del departamento de SQA en una empresa de desarrollo de software. Además de este objetivo principal se desarrollaron cuatro objetivos específicos. Estos objetivos específicos estuvieron orientados a alcanzar el objetivo general, mediante la identificación de las formas en que las organizaciones de desarrollo de software estructuran los procesos de SQA, conociendo los aspectos claves relacionados con la organización operativa de los departamentos de SQA, estableciendo un marco de referencia válido y finalmente validando la idoneidad de la propuesta de solución mediante el criterio experto.

Por su parte, la conclusión principal de este trabajo final de graduación es que, según los expertos, la propuesta de solución diseñada puede ser implementada en cualquier tipo de organización, ya que proporciona una base sólida para garantizar la calidad del software.

Mientras que la recomendación principal se enfoca en que la calidad del software es un esfuerzo continuo y de toda la organización, por lo que se recomienda que esta se integre en todos los aspectos y a lo largo del ciclo de vida del desarrollo de software.

Capítulo I. El problema y su importancia

1.1 Título del tema de investigación

Diseño de una propuesta de implementación del departamento de SQA en una empresa de software.

1.2 Antecedentes

En Estados Unidos, un error en el sistema de recaudación de impuestos ha provocado que millones de personas usen el código incorrecto en el sitio web para pagar los impuestos sobre la renta. Este error provocó que muchas personas pagaran menos impuestos y otras pagaran mucho más de lo que debían. Desafortunadamente, tomó un año entero descubrir este error. Ahora el Departamento de Impuesto sobre la Renta cuenta con innumerables declaraciones de impuestos erróneas. Si se hubiese tenido un departamento de aseguramiento de la calidad del software (conocido en inglés como Software Quality Assurance y de ahí sus siglas SQA o solo QA) y los miembros de este hubieran llevado a cabo los procesos adecuados de SQA, este tipo de falla hubiese sido detectada, reportada y corregida a tiempo (Galin, D. 2018). Como se muestra en el ejemplo anterior, las fallas en el software pueden tener un gran impacto negativo y causar daños sustanciales. Según Sahni, I., & Anand, A. (2014), se estima que las fallas en el software han llegado a costarle a las organizaciones hasta \$175 mil millones. La mayoría de los errores que se encuentran durante el desarrollo de software se pueden evitar adoptando un proceso de desarrollo de software sólido y teniendo un estricto control de la calidad del software mediante el aseguramiento de la calidad.

La falta del aseguramiento de la calidad del software es un gran problema que afecta a la industria del software desde hace varios años (Hossain, M. 2018). El aseguramiento de la calidad es un factor importante en la industria del software. Para lograr una posición respetable en el mercado global de la industria de TI, una empresa debe hacer productos de muy alta calidad. La competencia es latente y las organizaciones no pueden darse el lujo de corregir errores después de enviar los productos al cliente. Una corrección después del envío es muy costosa y afecta la credibilidad de la compañía. Las organizaciones no pueden permitirse perder clientes debido a este tipo de problemas y para evitar esto, las organizaciones deben contar y seguir un plan de gestión de la calidad adecuado para eliminar los errores de los productos de software (Javed et al., 2012).

1.3 Justificación

Los sistemas de software son cada vez más difíciles de desarrollar, debido a la amplia aceptación que tienen, diariamente vemos como incrementa el uso de diferentes softwares y como estos nos afectan en casi todos los aspectos de nuestra vida diaria de una u otra manera. Por ende, existe la presión y la responsabilidad dentro de las organizaciones por crear software que cumpla con altos estándares de calidad. La mala calidad del software conduce a consecuencias graves tanto para las organizaciones como para los usuarios. Algunas de estas consecuencias pueden ser: daño a la reputación de la empresa o producto, pérdida de clientes, costos económicos adicionales y pérdida de vidas, entre otros. Las fallas de software son, en última instancia, el resultado de fallas en el aseguramiento de la calidad (Walkinshaw, N. 2017).

El aseguramiento de la calidad del software desempeña un papel fundamental en el ciclo de vida del desarrollo de software o SDLC (por las siglas en inglés de Software Development Life Cycle). Cuando el equipo de SQA, conoce desde etapas tempranas los requerimientos de los clientes, revisa el diseño desarrollado y luego aplica las técnicas de prueba correctas, esto permite aumentar la calidad del producto de software, ahorrar tiempo y costos (Laaraib et al., 2021).

Habiendo mencionado esto, es evidente que, con la creciente complejidad y escalabilidad de los sistemas de software, exige una demanda cada vez mayor del aseguramiento de la calidad del software y por ende la importancia y necesidad de que las empresas desarrolladoras de software cuenten con un departamento de SQA, que a través del uso de diversas estrategias y técnicas pueda transmitir un nivel de confianza en la calidad del sistema desarrollado por la empresa.

1.4 Problema

Muchas organizaciones de desarrollo de software operan sin un departamento de SQA que les ayude en la gestión de procedimientos y estándares de SQA, sin estos procedimientos y estándares, no se tiene una base sólida para juzgar el trabajo del equipo de desarrollo, el cual rara vez produce planes de calidad verificables.

Existen muchas razones por las que una empresa que se encargue de desarrollar software debe considerar implementar un departamento de SQA. Se trata de la supervivencia empresarial, un aseguramiento de la calidad del software (SQA)

insuficiente o su ausencia del todo, aumenta las complicaciones de los desarrolladores, desaprobación de las partes interesadas y malas experiencias de los consumidores. Algunos de los riesgos más comunes del lanzamiento de software sin aseguramiento y control de calidad incluyen gastos excesivos, lanzamientos de productos retrasados, usuarios insatisfechos y reputación de marca reducida. Los peligros de la implementación de software sin control de calidad superan con creces cualquier beneficio de un lanzamiento de software rápido, pero defectuoso (Manocha, V. 2014).

Para este trabajo de investigación se propone ¿cómo implementar un departamento de aseguramiento de la calidad en una empresa de desarrollo de software?

1.5 Objetivo general

Diseñar una propuesta para implementar un departamento de SQA en una empresa de desarrollo de software por medio de un análisis teórico práctico de algunas de las mejores prácticas disponibles.

1.6 Objetivos específicos

- Identificar las formas en cómo las organizaciones de desarrollo de software organizan los procesos de SQA, por medio de una revisión sistemática de literatura para identificar el uso de buenas prácticas dentro de la industria.
- Conocer algunos de los principales aspectos relacionados con la organización operativa de los departamentos de SQA en organizaciones de software por medio de entrevistas dirigidas a expertos y por observación.
- Diseñar una propuesta de cómo implementar un departamento de SQA, tomando como base la teoría y experiencias empíricas recopiladas para establecer un marco de referencia válido.
- Validar la propuesta para la implementación de un departamento de SQA, por medio de un proceso de evaluación pareada para someter la propuesta a la opinión y recomendaciones de mejora de expertos relacionados con la gestión de SQA en empresas de desarrollo de software.

1.7 Metas por alcanzar por objetivo

En la Tabla 1 se detallan las metas por alcanzar para cada uno de los objetivos específicos.

Tabla 1. Metas por alcanzar por objetivo. Elaboración propia

Objetivo General	Presentar una propuesta para implementar un departamento de SQA en una empresa de desarrollo de software.
Primer Objetivo Específico	Tomando como base la revisión de literatura documentar los procesos de SQA en organizaciones de software.
Segundo Objetivo Específico	Analizar los principales aspectos relacionados con la organización operativa de los departamentos de SQA en organizaciones de software.
Tercer Objetivo Específico	Diseñar una propuesta metodológica de cómo implementar un departamento de SQA.
Cuarto Objetivo Específico	Validar la propuesta metodológica por medio de un proceso de valuación pareada.

Capítulo II. Marco teórico o referencial

En este apartado se presentan los diversos elementos teóricos que permitieron estructurar los principales conceptos a utilizados en la investigación. Estos se estructuran en cinco subapartados, a saber: conceptos de calidad de software, la organización y el departamento de aseguramiento de la calidad del software, el proceso del aseguramiento de la calidad, roles y responsabilidades del departamento de aseguramiento de la calidad del software y finalmente estándares, guía, comités e institutos relacionados con SQA.

2.1 Conceptos de calidad de software

Todo marco teórico relacionado con temas como el abordado en este trabajo final de graduación tiene necesariamente que clarificar la diferencia entre calidad, aseguramiento de calidad (conocido por sus siglas en inglés como QA), control de calidad (conocido por sus siglas en inglés como QC) y pruebas de software (conocido en inglés como testing).

2.1.1 ¿Qué es calidad?

El término “Calidad” ha evolucionado a lo largo del tiempo. Algunas de las definiciones recopiladas más representativas son las siguientes:

ISO –9000:2000: Calidad significa “El grado en que un conjunto de características inherentes cumple con los requisitos”.

American Society for Quality (2023), define calidad como un término subjetivo para el cual cada persona o sector tiene su propia definición. En el uso técnico, la calidad puede tener dos significados: 1) las características de un producto o servicio que se relacionan con su capacidad para satisfacer necesidades establecidas o implícitas; 2) un producto o servicio libre de deficiencias.

Según Lewis (2017), al observar la literatura informática, se puede notar que hay dos definiciones generalmente aceptadas del término calidad. La primera es que la calidad es cumplir con los requisitos. Por ende, se dice que la calidad es un estado binario; considerando esto, un producto es o no de calidad. Siempre y cuando los requisitos sean medibles, se puede determinar si la calidad se ha cumplido o no. La segunda definición de calidad está relacionada con la opinión del cliente, ya que este define la calidad en cuanto a si el producto o servicio hace lo que el cliente necesita.

2.1.2 ¿Qué es aseguramiento de calidad y control de calidad?

Según Pressman (2010), los procesos para mejorar la calidad del software comenzaron en la década de 1990, luego de que las principales corporaciones reconocieran que cada año se desperdician miles de millones de dólares en software que no tenía las características ni la funcionalidad solicitadas por los usuarios. Esto escaló a tal grado que tanto el gobierno como la industria se preocupaban por la posibilidad de que alguna falla de software pudiera afectar la infraestructura importante, provocará la pérdida de vidas y de decenas de miles de millones de dólares. Considerando esto, se empezaron a aplicar procesos de control y del aseguramiento de la calidad del software. El control de calidad incluye un conjunto de acciones de ingeniería de software que principalmente ayudan a asegurar que todo producto de trabajo cumpla sus metas de calidad. Por otro lado, el aseguramiento de la calidad consiste en un conjunto de actividades que definen y evalúan la adecuación de los procesos de software para proporcionar evidencia que establezca la confianza de que los procesos de software son apropiados y producen productos de software de calidad adecuada para los fines previstos (IEEE Std 730-2014). Una de las principales metas del aseguramiento de la calidad es proveer al equipo administrativo y técnico los datos necesarios para mantenerlos informados sobre la calidad del producto (Pressman, 2010).

A menudo, las personas utilizan el término aseguramiento de la calidad (o QA por sus siglas en inglés) para referirse a las pruebas; sin embargo, el aseguramiento de la calidad y las pruebas no son lo mismo, pero están relacionados. A ambos términos los une un concepto mucho más amplio como es, la gestión de la calidad. Este último término incluye todas las actividades que dirigen y controlan una organización con respecto a la calidad. La gestión de la calidad incluye tanto el aseguramiento de la calidad como el control de la calidad. El aseguramiento de la calidad se centra, por lo general, en el cumplimiento de los procesos adecuados, a fin de proporcionar la confianza de que se alcanzarán los niveles de calidad adecuados. Cuando los procesos se llevan a cabo de forma correcta, los productos de trabajo creados por esos procesos son generalmente de mayor calidad, lo que contribuye a la prevención de defectos. Por su lado, el control de la calidad implica varias actividades de prueba, que apoyan el logro de niveles apropiados de calidad. Las actividades de prueba son parte del proceso general de desarrollo o mantenimiento del software. Dado que el aseguramiento de calidad se ocupa de la correcta ejecución de todo el proceso, el aseguramiento de calidad promueve la realización de la prueba adecuada (ISTQB, 2018).

SWEBOK (2014), menciona que la gestión de la calidad del software es el conjunto de todos los procesos que aseguran que los productos de software, los servicios y las implementaciones del proceso del ciclo de vida cumplan con los objetivos de calidad del software de la organización y logren la satisfacción de las partes interesadas. La gestión de la calidad del software comprende cuatro subcategorías, ver Tabla 2.

Tabla 2. Subcategorías de la gestión de la calidad. Elaboración propia.

Planificación de la Calidad del Software	Garantía de la Calidad del Software	Control de la Calidad del Software	Mejora del Proceso del Software
<p>Determina los estándares y procesos de calidad a utilizar.</p> <p>Define los objetivos en lo referente a la calidad de software.</p> <p>Estima el esfuerzo y el cronograma de las actividades de calidad del software.</p>	<p>Define y evalúa la idoneidad de los procesos de software.</p> <p>Proporciona evidencia de que los procesos de software son apropiados y producen productos de calidad.</p> <p>Facilita la mejora continua.</p>	<p>Examina artefactos específicos del proyecto (documentos y ejecutables) para determinar si se cumple con los estándares establecidos para el proyecto (incluidos los requerimientos, restricciones, diseños, contratos y planes).</p> <p>Se ejecutan las pruebas.</p>	<p>Busca mejorar la efectividad, eficiencia y otras características del proceso con el objetivo final de mejorar la calidad del software.</p> <p>Podría incluirse en cualquiera de las primeras tres categorías.</p>

2.1.3 ¿Qué son pruebas de software?

Las pruebas de software, conocido en inglés como “testing” de software, es una actividad desarrollada para evaluar la calidad del producto de software y para mejorarlo al identificar defectos y problemas que pudiera este tener. El testing de software consiste en la verificación dinámica del comportamiento de un programa sobre un conjunto finito de casos de prueba, apropiadamente seleccionados a partir del dominio de ejecución que usualmente es infinito, en relación con el comportamiento esperado (Dupuis et al.,1999). Otras definiciones clásicas de pruebas de software apuntan a que estas pueden ser usadas para mostrar la presencia de defectos, también conocidos en inglés como “bugs”, pero nunca su ausencia (Dijkstra, 1970). Finalmente, se considera a las pruebas de software como una actividad realizada para evaluar la calidad del producto y mejorarla, identificando defectos y problemas (Bourque, P., & Fairley, R. E. 2014).

Para fines de esta investigación se utilizará la definición por Owens & Khazanchi (2009), la cual indica que el aseguramiento de la calidad del software es un proceso repetible y bien definido que se integra con la gestión de proyectos y los ciclos de vida de desarrollo de software, para revisar los mecanismos de control interno, asegurar el cumplimiento de los estándares y procedimientos de software. El objetivo del proceso es asegurar la conformidad con los requisitos, ejecutar actividades de prueba para reducir el riesgo, evaluar los controles internos y mejorar la calidad mientras se ajusta al cronograma y presupuesto establecido. En la Figura 1, se ilustra la manera en que se agrupan algunos de los términos mencionados anteriormente.

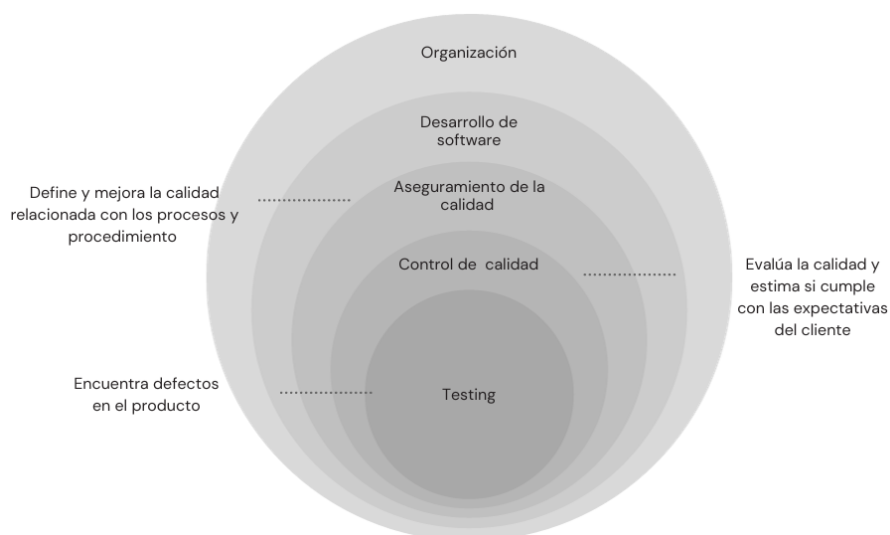


Figura 1. Aseguramiento de la calidad. Mejorado a partir de Manocha, V. (2014)

2.2 La organización y el departamento de SQA

El IEEE (2014), menciona que se requiere tanto el apoyo como el compromiso de la organización para que las operaciones y procesos del departamento de SQA sean eficaces. Según Galin (2018) y el IEEE (2014), Este apoyo incluye al menos:

- Que la gerencia esté familiarizada y comprenda las políticas, propósitos, conceptos, prácticas y necesidades del departamento de SQA.
- Desarrollar y establecer una política de calidad de software que se ajuste a las metas y al propósito de la organización, a través del cumplimiento de los objetivos de calidad relacionados con los productos de software.
- La gerencia proporciona al departamento de SQA un nivel apropiado de recursos calificados (personas, equipos, conocimientos, instalaciones y herramientas) para cumplir con sus responsabilidades, realizar las actividades de control y aseguramiento de la calidad del software con el fin de establecer la mejora continua.
- La gerencia transmite la importancia de que los productos y servicios de software cumplan con los requisitos reglamentarios y del cliente.

Por su parte, la versión del ISTQB Expert level (2011) menciona que las actividades de calidad no deben limitarse únicamente al equipo de SQA, cada grupo que contribuye a la definición, creación y soporte de un producto debe ser consciente de que también está contribuyendo a la calidad de ese producto. Por ejemplo:

- Las partes interesadas en el proyecto deben participar activamente en las revisiones o aclaración de consultas relacionadas con los requerimientos o la documentación.
- Los desarrolladores deben hacer uso de las buenas prácticas y aplicar pruebas unitarias, entre otras.

En resumen, la calidad de software es un esfuerzo colaborativo que implica a todos los miembros de la organización.

2.3 El proceso del aseguramiento de la calidad de software

Galin (2018), argumenta que los procesos de aseguramiento de la calidad del software (SQA), se crearon para servir como infraestructura y base para la gestión del desempeño de la calidad del software por parte de la organización. Por lo que estas actividades son de carácter estratégico. Los procesos de calidad del software consisten en tareas y técnicas para indicar cómo se implementan los planes de software (por ejemplo, planes de gestión de software, desarrollo, gestión de calidad o gestión de configuración) y qué tan bien los productos intermedios y finales cumplen con sus requisitos específicos (SWEBOOK,2014).

El proceso del aseguramiento de la calidad de software (SQA), es un mecanismo de evaluación continua implementado a lo largo de los ciclos de vida del proyecto y desarrollo con controles específicos y requisitos de documentación. El proceso debe usarse para controlar y asegurar el proceso de desarrollo de software, con el fin de obtener mejores resultados en lo que a calidad respecta (Thayer & Fairley, 1999).

El IEEE (2014), menciona que el propósito de los procesos SQA es especificar las actividades y tareas que le ayudan a la organización a producir, recopilar y validar evidencia que permita asentar la base para una declaración justificada de confianza de que el producto de software cumple con los requisitos establecidos. SQA asegura que los procesos sean establecidos, administrados, mantenidos y aplicados en los diferentes proyectos por personal capacitado y calificado y que las actividades y tareas realizadas sean proporcionales al riesgo del producto. Los procesos de SQA definen el rol, los conceptos, los métodos, los procedimientos y las prácticas de la función de SQA.

Este mismo estándar sugiere que las principales actividades para implementar un proceso de SQA en la organización, son las siguientes:

1. Implementar las actividades del proceso de SQA. Se deben definir, establecer y documentar los procesos y estándares de SQA para cada uno de los proyectos. Cuando estos procesos de SQA se aplican a un proyecto, permiten el desarrollo

de software que se ajusta a los requisitos establecidos. Estos procesos también brindan a los proyectos mediciones de la calidad del software para ayudar a realizar compensaciones de costo, cronograma, calidad y riesgo.

Implementar las actividades del proceso de SQA incluye las siguientes tareas:

- Establecer procesos de SQA.
- Crear y documentar el plan del departamento SQA.
- Ejecutar el plan desarrollado por el equipo de SQA
- Coordinar con los procesos de desarrollo de software.
- Administrar y registrar los registros del departamento de SQA, esto conducirá a la mejora continua en función de medidas objetivas y resultados reales del proyecto.

2. Aseguramiento del Producto. Los productos son el software y la documentación relacionada. La documentación relacionada puede incluir todos aquellos documentos asociados con el desarrollo, el soporte, el mantenimiento, la instalación y la administración. Un producto también puede ser un servicio de software proporcionado.

Los resultados de esta actividad proporcionan evidencia de que los servicios de software, los productos y cualquier documentación relacionada se identifican en el contrato y cumplen con el, además que se identifican y abordan las no conformidades.

El aseguramiento del producto incluye las siguientes tareas:

- Evaluar los planes y requerimientos del usuario para llevar a cabo el cumplimiento de los contratos, normas y reglamentos.
- Evaluar el producto para garantizar la conformidad y aceptabilidad del producto por parte del usuario.
- Evaluar y asegurar el ciclo de vida del producto para garantizar que siguen las metodologías y se cumplen con los requisitos del diseño.
- Medir la calidad del producto a través de los diferentes tipos de pruebas.
- Registrar y evidenciar los problemas identificados.

3. Aseguramiento de procesos. Los resultados de esta actividad permiten asegurar que los procesos utilizados para desarrollar, instalar, operar y mantener el software se ajusten al contrato, cumplen con las reglamentaciones impuestas y son adecuados, eficientes y efectivos, por lo que el software se ajusta a los requisitos establecidos.

El aseguramiento de procesos incluye las siguientes tareas:

- Evaluar los procesos del ciclo de vida y los planes para la conformidad.
- Evaluar los entornos, para determinar si los entornos de ingeniería de software y los entornos de prueba de software se ajustan a los procesos y planes del proyecto.
- Medir la efectividad de los procesos y sugerir mejoras.
- Registrar y evidenciar los problemas identificados
- Evaluar la habilidad y el conocimiento del personal, para determinar si el personal asignado al proyecto tiene los conocimientos, habilidades, competencias necesarias para realizar las tareas requeridas para sus funciones.

En conclusión, los procesos de SQA tienen que ser definidos y desarrollados independientemente de los proyectos de software de la organización para lograr la calidad adecuada de los productos y servicios de software. Los datos recopilados sobre el rendimiento de calidad de los proyectos de software, incluidos los problemas, permiten que la función SQA identifique oportunidades de mejora de procesos y planifique su implementación. En organizaciones nuevas o en las cuales no se cuenta con un departamento de SQA lo recomendable es completar estos procesos antes de firmar nuevos contratos de software (Galín, D. 2018).

El proceso de SQA consta de una variedad de fases con actividades específicas. Los desarrolladores se encargan de crear el software mientras el equipo de SQA es responsable del aseguramiento de la calidad del software, lo cual incluye diferentes actividades como planificación, revisiones formales, pruebas, control de la documentación, análisis e informes (Presman, 2005). El proceso del SQA es más efectivo cuando se lleva a cabo a través de un equipo de profesionales adecuadamente

cualificados (Godbole, 2004). Los roles y responsabilidades del equipo de SQA incluye diferentes actividades, las cuales se cubrirán en el siguiente subapartado.

2.4 Roles y responsabilidades del departamento de SQA

Existen varios roles dentro del aseguramiento de la calidad del software, pero ISTQB (2018) indica, que los principales roles con que debe contar todo departamento de SQA dentro de una empresa de desarrollo de software son, el jefe de pruebas (QA Lead, en inglés) y los probadores o ingenieros de pruebas (conocidos en inglés como QA o testers). Las actividades y tareas desarrolladas por estos dos roles dependen del contexto del proyecto, del producto y de las competencias de las personas en los roles y dentro de la organización, de manera general los podemos detallar de esta manera:

- El jefe de pruebas asume la responsabilidad y el liderazgo de las actividades relacionadas con las pruebas. Este rol se encarga de la gestión de las pruebas. Puede ser desempeñado por un jefe de pruebas profesional, por un jefe de proyecto, un jefe de desarrollo o un responsable de aseguramiento de la calidad. En proyectos u organizaciones más grandes, varios equipos de prueba pueden informar a un jefe de prueba.
- Un ingeniero de pruebas es un profesional de tecnología de la información (TI) que está a cargo de una o más actividades de prueba técnica, incluido el diseño de entradas de pruebas, la implementación de casos de prueba, la ejecución de scripts de prueba, el análisis e informe de resultados y la configuración del entorno de pruebas (Graham et al., 2008). Según Everett & McLeod (2007), el papel del ingeniero de pruebas de software es el de un verificador. El ingeniero de pruebas analiza objetivamente el software en curso, independientemente de los autores de los documentos de desarrollo y del código del programa, y determina a través de pruebas repetidas si el software cumple con los requisitos y especificaciones solicitadas.

En la Tabla 3, se muestran las responsabilidades habituales tanto para el jefe de pruebas como para el ingeniero de pruebas.

Tabla 3. Responsabilidades. Elaborado a partir de ISTQB, 2018.

Jefe de prueba	Ingeniero de pruebas
<ul style="list-style-type: none"> ● Desarrollar o revisar la política y la estrategia de prueba de la organización. ● Planificar las actividades de las pruebas. ● Redactar y actualizar los planes de prueba. ● Coordinar los planes de prueba con los interesados. ● Preparar y entregar informes sobre las pruebas. ● Adaptar la planificación en función de los resultados y avance de la prueba. ● Introducir métricas adecuadas para medir el avance de la prueba y evaluar la calidad de la prueba y del producto. ● Apoyar la selección e implementación de herramientas para dar soporte al proceso de prueba. ● Decidir sobre la implementación de los entornos de prueba. ● Promover y defender al equipo de prueba. ● Desarrollar las competencias y las carreras profesionales de los probadores. 	<ul style="list-style-type: none"> ● Revisar y aportar al plan de pruebas. ● Analizar, revisar y evaluar toda la documentación para la prueba. ● Identificar y documentar las condiciones de prueba. ● Diseñar, configurar y verificar los entornos de prueba. ● Diseñar e implementar casos de prueba. ● Preparar y obtener datos de prueba. ● Crear el calendario de pruebas. ● Ejecutar pruebas. ● Evaluar los resultados. ● Documentar las desviaciones de los resultados. ● Usar herramientas apropiadas para facilitar el proceso de prueba. ● Automatizar las pruebas según sea necesario. ● Evaluar características no funcionales como el desempeño, usabilidad, seguridad y compatibilidad, entre otras. ● Revisar pruebas desarrolladas por otros.

Es importante mencionar que se puede diferenciar entre dos tipos de ingeniero de pruebas. Los ingenieros de pruebas funcionales, los cuales están más orientados al diseño y la ejecución de pruebas para los productos durante el proceso de desarrollo y los ingenieros de pruebas técnicos, estos últimos orientados a la automatización de pruebas, rendimiento y la optimización de que se cumplan los estándares de calidad en los productos de software (Huerta, 2017).

McCaffrey (2019), menciona que un buen probador de software debe poseer las siguientes cualidades:

- Pasión por el análisis y las pruebas.
- Habilidades técnicas.
- Capacidad para priorizar y organizar.
- Capacidad de adaptación y aprendizaje.
- Capacidad para trabajar sin supervisión directa.
- Capacidad para comunicarse de manera efectiva.
- Capacidad para comprender la estrategia de negocios.

2.5 Organizaciones, institutos, estándares, comités y guías

Temas como el abordado en este trabajo final de graduación tienen necesariamente que mencionar y detallar algunas de las organizaciones, institutos, estándares, comités y guías relacionados con el aseguramiento de la calidad del software. Con el fin de exponer más sobre cada uno de ellos, comprender su importancia, principales elementos y aportes en el ámbito del SQA.

Según Galin (2018), son varias las organizaciones e institutos reconocidos que se encargan de redactar y desarrollar estándares de calidad para unificar las buenas prácticas en el aseguramiento de la calidad del software. Algunos de estos son:

- **IEEE:** Institute of Electrical and Electronics Engineers, en español “Instituto de Ingenieros Eléctricos y Electrónicos”. Se enfoca en promover la investigación, la publicación de revistas, conferencias científicas y la creación de estándares técnicos en diversos campos, incluidos la informática y el desarrollo de software entre otros.
- **ISO:** International Organization for Standardization, en español “Organización Internacional de Normalización”. Sus normas especifican requerimientos para

garantizar que los productos y/o servicios cumplen con su objetivo de garantizar calidad, seguridad y eficiencia. Los estándares ISO abarcan una gran variedad de temas, entre ellos la gestión de la calidad, la seguridad de la información y tecnologías de la información, entre otros.

- **ANSI:** American National Standards Institute, en español Instituto Nacional de Estándares de los Estados Unidos. Es una organización sin fines de lucro que supervisa el desarrollo de estándares para productos, servicios, procesos y sistemas en los Estados Unidos.

Galín (2018), menciona, que los estándares se pueden clasificar en dos grupos principales:

- **Estándares del proceso del proyecto:** Los estándares de este grupo se enfocan en metodologías para llevar a cabo proyectos de desarrollo y mantenimiento de software y asegurar su calidad. Este tipo de estándares definen: los pasos a seguir, los requisitos de documentación, las pruebas de software a realizar, entre otros.
- **Estándares de gestión de la calidad.** Los estándares de esta clase se centran en la infraestructura de la organización, la organización y la gestión del departamento de SQA y requisitos.

Hay una gran variedad de estándares de calidad disponibles que ayudan a controlar y mantener la gestión de la calidad. A continuación, se muestran los más relacionados con el tema de investigación:

- **IEEE Std. 730-2014:** Estándar para el aseguramiento de la calidad del software, establece los requisitos para iniciar, planificar, controlar y ejecutar los procesos de Aseguramiento de la Calidad del Software de un proyecto de desarrollo o mantenimiento de software (Heimann, D. 2014).
- **IEEE Std 1061-1992:** Estándar para una metodología de métricas de calidad de software, proporciona una metodología para establecer requisitos de calidad e identificar, implementar, analizar y validar el proceso y el producto de las métricas de calidad del software (IEEE, 1993).

- ISO 9001: Estándar ampliamente reconocido para sistemas de gestión de calidad, que describe los conceptos y principios fundamentales de la gestión de la calidad que son universalmente aplicables a las organizaciones de todos los tipos y tamaños, que buscan el éxito a través de la implementación de un sistema de gestión de calidad. También proporciona un marco para desarrollar e implementar un sistema integral de gestión de calidad propio (Sánchez, J. M. C. 2017).
- ISO 25000: Proporciona una nueva serie de estándares internacionales sobre la medición de la calidad del producto de software, llamados Requisitos y Evaluación de la Calidad de Productos de Software o SQuaRE por sus siglas en inglés Software Product Quality Requirements and Evaluation (Yang, H. 2012).

Comités, marcos y guías

- ISTQB® International Software Testing Qualifications Board, por sus siglas en inglés o Comité Internacional de Certificaciones de Pruebas de Software en español. Es la certificación líder en el campo de las pruebas de software. Las principales razones para tomar en cuenta estas publicaciones y certificaciones, es que en sus silabarios incorporan normas IEEE e ISO. Este comité ha creado diferentes niveles de certificaciones que van desde su nivel básico hasta el experto y otras especialidades relacionadas siempre con el área de calidad de software (ISTQB, 2014).
- CMMI por sus siglas en inglés Capability Maturity Model Integration o Modelo Integrado de Madurez y Capacidad en español: Abarca temas de información en la producción de software, por lo cual se utiliza para mejorar procesos, aumentar la producción y aumentar la competitividad de una organización (Khraiwesh, M. 2014).
- La Guía del Cuerpo de Conocimientos de Ingeniería de Software (SWEBOK,2014), documento promovido por el IEEE, que proporciona conocimientos generalmente aceptados para la profesión de ingeniería de software. El contenido de la Guía SWEBOK se deriva de fuentes académicas y las mejores prácticas de la profesión (Bourque, P., & Fairley, R. E. 2014).

Capítulo III. Marco metodológico

En este apartado se describe la metodología utilizada para realizar la investigación, se incluyen aspectos como el enfoque, el tipo de la investigación, las fuentes y sujetos de información, la población y la descripción de los instrumentos a utilizar.

3.1 Enfoque de la investigación

Se detalla el marco metodológico utilizado en el proyecto, además de las técnicas y métodos que se utilizaron en la investigación. Esta investigación se desarrolla dentro de un enfoque fundamentalmente cualitativo, ya que no se manejan grandes volúmenes de datos numéricos, solamente los pocos generados por el instrumento a desarrollar. Sin embargo, puede ser sustentado en algunos puntos por el enfoque cuantitativo. El enfoque cuantitativo es aquel en el cual se registra y verifica cuidadosamente información, casi siempre en forma de números. Mientras, el enfoque cualitativo busca utilizar una amplia gama de técnicas de recolección y análisis de datos que utilizan datos no tan estructurados como los manejados en el enfoque anterior (Choy, 2014).

3.2 Tipo de la investigación

Se realizó un trabajo de revisión, análisis, extracción e interpretación de diferentes fuentes bibliográficas de alta calidad y la utilización de diversos tipos de técnicas, como, entrevistas, observaciones y evaluación par. Cada una de estas técnicas permitió recopilar información de diferente tipo, por este motivo, es importante conocer sus características. A continuación, se detalla cada una de las técnicas:

3.2.1 Revisión de literatura

Cortés & León (2004), indican que la revisión de la literatura es el proceso mediante el cual se investiga, consulta, analiza, evalúa, extrae y recopila la información relevante para descubrir lo que ya se sabe sobre el problema a investigar. Las principales ventajas de la revisión de literatura son:

- Proporciona una visión general sobre los conceptos relacionados con el tema a investigar.
- Se dispone de una gran cantidad de información.
- Permite la comparación continua con bibliografías anteriores.

Una revisión de la literatura le permite al autor a aprender sobre la historia y la naturaleza de su tema, al identificar problemas y lagunas en la investigación. Se sugiere que la elección y la cantidad de documentos debe ser guiada por la pregunta de investigación (Paradis et al., 2016).

Las principales debilidades de la revisión de literatura son:

- La búsqueda bibliográfica puede ser exhaustiva y consumir mucho tiempo.
- Riesgo de encontrar información desactualizada.

La utilización de esta técnica permitió tener insumos que generaran el desarrollo del proyecto de investigación.

La selección de los temas se hizo a través del uso de una matriz de síntesis. Este método es especialmente provechoso para organizar revisiones de literatura. La matriz de síntesis es una tabla que permite al investigador clasificar y ordenar los diferentes argumentos presentados de un tema. En la fila de arriba se muestran las fuentes de información, mientras que en las filas de la tabla se muestran los principales temas, como se muestra en la Tabla 4. Matriz de síntesis (Ingram et al., 2006).

Tabla 4. Matriz de síntesis. Elaboración propia.

Temas	ISTQB Fundamentals	ISTQB Test Management	ISTQB Test Manager	PMBOK V6	ISO	SWEBOK	IEEE
Conceptos de calidad del software	✓				✓	✓	
La organización y el departamento de SQA	✓	✓	✓	✓			✓
El proceso del aseguramiento de calidad					✓	✓	✓
Roles y responsabilidades del departamento de SQA	✓	✓					
Organizaciones, institutos, estándares, comités y guías	✓	✓	✓		✓	✓	✓

Para esta matriz de síntesis se consideraron documentos relevantes, como diferentes estándares y normas de instituciones emblemáticas tales como la Junta Internacional de Calificaciones de Pruebas de Software (ISTQB), el Cuerpo de Conocimiento en Dirección de Proyectos (PMBOK), el Instituto de ingenieros eléctricos y electrónicos (IEEE) y la Organización Internacional de Normalización (ISO), entre otros. Todos estos han sido descritos a grandes rasgos en el capítulo II del marco teórico.

Para la revisión de literatura, también se utilizaron fichas de síntesis, como la mostrada en la Figura 2, como instrumento. Una ficha por cada elemento de literatura. Esto con el objetivo de llevar a cabo el análisis, la síntesis y la facilitación de la clasificación de los elementos de literatura.

FICHA DE SÍNTESIS		
Tema: Turismo	Subtema: Oportunidades de ^{turismo} trabajo	Nº 3
<p>A causa de la epidemia de la influenza y la crisis económica del año 2009, muchos empleos se unieron a pesar. Los agencias de viajes son una de ellas y se ofrecen varias alternativas para hacer crecer su negocio. La música atrae a los turistas, en los conciertos de sus artistas favoritos, es una gran alternativa para que los agencias de viajes abran su campo y organicen viajes para ese tipo de turistas. La gastronomía y el turismo "negro" hacen que los turistas quieran a países y conozcan su cultura o temas de su comida o tal vez su historia a través de sus lugares de eventos o actividades que ocurren en esos lugares.</p>		
Referencia: Abbott, R. I. (2010). <u>Unos vacaciones diferentes</u> . <i>Emprepreneur</i> , Vol. 18 Número 7, 10.		
Observaciones: Actividad en clase	Tipo: Síntesis	
Id. Investigador: Miguel Ángel Arellano Fraga	Fecha: 11. Octubre. 2011	

Figura 2. Ejemplo de ficha de trabajo llena.

3.2.2 Entrevistas

Griffiee (2005), describe la entrevista como un método de investigación cualitativa que se basa en hacer preguntas para recopilar datos. Las principales fortalezas de la entrevista son:

- Alta tasa de respuesta.
- Es una técnica muy flexible.
- Proporcionan una muestra más representativa.

Las entrevistas se utilizan para recopilar información de personas 1 a 1, utilizando una serie de preguntas predeterminadas. Estas pueden ser estructuradas o no estructuradas; Pueden seguir un guion estrictamente escrito que imite una encuesta o inspirarse en un conjunto suelto de preguntas que invitan a los entrevistados a expresarse con mayor libertad. Los entrevistadores deben escuchar y cuestionar activamente, sondear e incitar más para recopilar datos más completos. Las entrevistas son ideales cuando se utilizan para documentar los relatos, percepciones o historias de los participantes sobre las actitudes y las respuestas a ciertas situaciones o fenómenos (Paradis et al., 2016).

Las principales debilidades de la entrevista son:

- Son costosas, ya que se requiere de más tiempo para llevarlas a cabo y analizarlas.
- Como necesita más tiempo, el tamaño de la muestra generalmente se minimiza.
- El sesgo del sujeto es siempre una amenaza para la validez de los hallazgos y su consistencia en la recopilación de datos de un sujeto a otro.

El uso de esta técnica y de un guion con preguntas previamente diseñadas, evitó el sesgo de hacer preguntas diferentes en cada entrevista a los diferentes profesionales del área de informática.

Los criterios utilizados para definir esta selección son de oportunidad. Esto obedece a las limitaciones de tiempo de la investigación y conveniencia para la investigadora. Los entrevistados seleccionados fueron únicamente sujetos relacionados directamente con la organización operativa de los departamentos de SQA en diferentes empresas de desarrollo de software y cumplían con los siguientes criterios de selección:

- Contar con estudios universitarios completos, relacionados con el área de ingeniería en sistemas o a fin.

- Desenvolverse en alguno de los siguientes puestos: ingeniero de pruebas, jefe de pruebas, gerente de pruebas, gerente de proyectos o desarrollador.
- Tener más de 5 años de experiencia laboral en alguno de los puestos mencionados en el punto anterior.

El número de sujetos a entrevistar es de 12 participantes. En el apéndice 1, se muestra el guion de preguntas integro a utilizar para la entrevista.

3.2.3 Observación

Paradis et al. (2016), describen la observación, como su nombre lo indica, es una forma de recopilar datos a través de la observación. Observando el comportamiento, los eventos o notando las características físicas en su entorno natural. Las principales fortalezas de la observación son:

- Es un método simple, no depende de la disposición de las personas para proporcionar información.
- Permite altos niveles de flexibilidad.
- Útil para enmarcar hipótesis.

Las observaciones se utilizan para recopilar información en sitio utilizando sentidos como la vista, oído, tacto y olfato. La técnica de observación se utiliza para registrar comportamientos específicos. La observación puede ser estructurada o no estructurada. Las observaciones estructuradas le permiten al investigador decidir cuidadosamente qué observar, cómo observar, durante cuánto tiempo y cómo registrar los datos observados. Mientras que, las observaciones no estructuradas se realizan espontáneamente y se registran como lo que se ve en palabras.

Las principales debilidades de la observación son:

- Altos niveles de sesgo del observador.
- Es necesario escribir notas de campo detalladas, para no olvidar datos importantes.
- Se requiere autodisciplina para el registro oportuno.

Se hizo uso de esta técnica, con el fin de observar, comprender y conocer algunos de los principales aspectos relacionados con la organización operativa de los departamentos de SQA en diferentes organizaciones de software. Se utilizó una bitácora como instrumento con el fin de recopilar toda aquella información observada útil para la investigación. El

proceso de observación se llevó a cabo a través de la visita a dos organizaciones a las cuales fui invitada sin intervenir ni alterar el ambiente, con el fin de identificar elementos y datos relevantes relacionados con el tema de SQA. Cabe resaltar que, por motivos de confidencialidad, los nombres de estas organizaciones ni de los clientes de esta son mencionados a lo largo de este trabajo final de graduación, pero sí los datos pertinentes a cómo se lleva a cabo la gestión de SQA.

En la Figura 3, se muestra el formato que fue utilizado para anotar los datos durante el proceso de observación.

Fecha	Hora
Observación	Role
Comentarios adicionales	

Figura 3. Formato de anotación para la observación. Elaboración propia.

3.2.4 Evaluación par

Según los múltiples aportes que hace Tanner (2018) en su libro, podemos definir la revisión por pares como una medida de control de calidad que aplica para varias áreas. Es un proceso en el que los profesionales revisan el trabajo de los demás para asegurarse de que sea preciso, relevante y significativo. Las principales ventajas de la revisión por pares son:

- Brinda comentarios, retroalimentación y permite identificar áreas de mejora al estar expuesto a diferentes perspectivas.
- Llevar a cabo revisiones por pares anónimas permite reducir el sesgo. Ya que el revisor se centra en evaluar el instrumento y no al autor.

La revisión por pares se considera por la gran mayoría de los investigadores como el mecanismo más efectivo y eficaz para garantizar la calidad, confiabilidad, integridad y consistencia de la literatura académica (Mora-Campos, 2015).

Las principales debilidades de la revisión por pares son:

- Puede incurrir en retrasos, ya que se requiere de un segundo evaluador para validarla.
- Al conocerse el autor se puede prestar al sesgo.

Esta técnica permitió evaluar la propuesta de implementación de un departamento de SQA. La propuesta de implementación fue sometida a la validación y recomendación de expertos relacionados con la gestión de SQA en empresas de desarrollo de software. Esto se llevó a cabo a través del uso de un instrumento de evaluación con el fin de sistematizar uniformemente la evaluación de la propuesta. Dicho instrumento se puede ver de manera íntegra en el apéndice 2.

3.3 Fuentes y sujetos de información

3.3.1 Fuentes

Las fuentes de información son todos aquellos medios de información que permiten satisfacer la necesidad de conocimiento en una situación o problema dado y que luego se utilizarán para lograr los objetivos establecidos. Según su origen, se dividen en fuentes primarias y fuentes secundarias (Soberón, U. E. M., & Acosta, Z. 2009). Para uso

de este documento, estas fueron descritas en los puntos: 3.2.1 Revisión de literatura, 3.2.2 Entrevistas, 3.2.3 Observación y 3.2.4 Evaluación par.

- Fuentes primarias

Soberón (2009), menciona que las fuentes primarias, son todos aquellos usuarios y acompañantes a quienes se les aplicó un instrumento de investigación. Para esta investigación las fuentes primarias se obtienen de los datos resultantes de los instrumentos de investigación: entrevistas, observación y revisión par. Adicionalmente, son fuentes primarias los artículos de revistas o conferencias resultantes de investigaciones realizadas por los autores de dichos artículos.

- Fuentes secundarias

Son las que contienen información primaria, sintetizada y reorganizada. Estas fuentes de información se caracterizan porque el investigador las recoge de estudios realizados anteriormente. Es información que existe de antemano (Abril, V. 2008). Para esta investigación las fuentes secundarias se obtuvieron de documentación bibliográfica como libros, estándares de SQA, guías, mejores prácticas y tesis, entre otros.

3.3.2 Sujetos

Los sujetos de estudio que participaron en la entrevista y en la revisión par, son profesionales del área de la informática, que ejercen diversos roles como, gerente de proyecto, ingenieros de pruebas de diferentes niveles de experiencia, gerentes de departamentos de SQA y jefes de equipos de desarrollo.

3.4 Población y muestra

Por la naturaleza de este trabajo de investigación no existe una población y muestra.

Capítulo IV. Diagnóstico y análisis de resultados

En este apartado se presenta el diagnóstico y análisis obtenidos al aplicar 3 de las 4 técnicas utilizadas en esta investigación (revisión de literatura, entrevistas y observación). Estas tres primeras técnicas permitieron desarrollar la propuesta de implementación del departamento de SQA. Esta propuesta se presenta en el capítulo V. Como parte de ese capítulo V, se desarrollará la cuarta técnica utilizada en este trabajo final de graduación (revisión par). Consecuentemente, el diagnóstico y análisis desarrollado a continuación, se deriva de los resultados obtenidos en las 3 primeras técnicas antes citadas.

4.1 Revisión de literatura

Se realizó una revisión de la literatura con el fin de crear un marco teórico que permitiera presentar los conceptos claves acerca del estudio realizado. El marco teórico íntegro fue presentado en el capítulo II. Aquí se muestra una síntesis de los resultados de esa revisión de literatura.

Para esta investigación se utilizaron fuentes bibliográficas de alta calidad, en total 26 referencias bibliográficas. Antes de continuar, se debe aclarar que, desde el punto de vista de tipo de referencias bibliográficas, existen dos tipos. Las referencias de fuentes primarias y las de secundarias. Las primarias hacen referencia a elementos de literatura que relatan resultados de investigaciones u opiniones vertidas por sus autores. Ejemplos típicos de este tipo de elementos de literatura de fuente primaria son los artículos de revistas o conferencias. Por otro lado, las referencias secundarias son aquellas que recogen o reportan fuentes primarias. Este tipo de bibliografía lo que hace es presentar o resumir resultados hechos por otros autores, de ahí que sean secundarias. Ejemplos de este tipo de fuentes son los libros, sitios de internet que relatan investigaciones de otras personas, etc.

Hecha esta aclaración, debemos indicar que en este trabajo de investigación el 61.54 % (16) fueron fuentes primarias y el 38.46 % (10) fuentes secundarias. El medio de publicación utilizado fue de revistas en un 34.6% (9), referencias de libros un 26.92% (7), estándares, guías y silabarios 26.92% (7) mientras que otros tipos de literatura en un 11.54% (3). El análisis de la antigüedad de estas referencias bibliográficas indica que solo el 50% (13) fueron inferiores al año 2013, pero se incluyen en este trabajo por la relevancia que aún aportan al tema en investigación.

4.2 Entrevistas

La segunda técnica utilizada fue la de entrevistas. Se aplicaron entrevistas a 12 participantes (las respuestas completas de dichas entrevistas se muestran en el Anexo 1: Respuestas de las entrevistas).

La presentación de los resultados de esta técnica se estructuran en subsecciones que contienen:

- Datos demográficos.
- Estructura del departamento de SQA.
- Perfil del personal del departamento de SQA.
- Implementación del departamento de SQA.
- Evaluación del departamento de SQA.
- Desarrollo del departamento de SQA.
- Consideraciones adicionales en la implementación del departamento de SQA.

Para el análisis de la subsección con los datos demográficos, simplemente se utilizaron elementos básicos de estadística descriptiva como cálculo de porcentajes y cuantificaciones. Para el resto de subsecciones, el análisis fue hecho utilizando los principios básicos del enfoque de la teoría fundamentada de Strauss y Corbin (Strauss y Corbin 1998). Esta teoría permitió identificar patrones derivados del análisis del texto (respuestas a las preguntas en las entrevistas). El investigador compara contenidos de diversas entrevistas u observaciones con el objetivo de notar las similitudes y diferencias de los datos, conduciendo así a la derivación de patrones que ayudan a comprender el fenómeno en estudio (Páramo Morales, D. 2015).

4.2.1 Datos demográficos

La entrevista incluyó algunas preguntas demográficas tendientes a establecer un marco de referencia de las cualidades de los participantes. Primeramente, los participantes reportaron contar con estudios universitarios completos y más de 5 años de experiencia laboral en roles tales como: ingeniero de pruebas, jefe de pruebas, gerente de pruebas, gerentes de proyectos o desarrolladores. El 66.7% (8 participantes) fueron de género masculino, mientras que el 33.3% (4 participantes) fueron de género femenino. A los participantes se les preguntó acerca de su nivel de educación, obteniéndose que el 66.7% (8 participantes) indicaron tener un bachillerato universitario completo, 16.7% (2

participantes) licenciatura y el 16.7% (2 participantes) maestría. El 75% (9 participantes) cuentan con algún tipo de certificación. Los participantes también reportaron su experiencia laboral. En este sentido, el 50 % (6 participantes) cuenta con más de 10 años de experiencia, el 33.3% (4 participantes) cuenta con 5 años de experiencia y el 16.7 % (2 participantes) indicaron tener entre 6 y 9 años de experiencia. Por otro lado, en cuanto al rol de trabajo, el 33.3% (4 participantes) indicaron ser ingenieros e pruebas, el 25% (3 participantes) refirieron que son líderes del departamento de calidad, el 25% (3 participantes) reportaron ser programadores y el 16.7% (2 participantes) mencionaron que son gerentes de proyectos. Para finalizar esta caracterización demográfica, el 66.7% de los entrevistados (8 participantes) comentaron que trabajan en una organización que cuenta con un departamento de SQA mientras que un 33.3% (4 participantes) no cuentan con dicho departamento en la empresa para la que laboran.

4.2.2 Estructura del departamento de SQA

En la entrevista se indagó sobre la estructura que debería tener un departamento de SQA. Hecho el análisis de identificación de patrones, los resultados obtenidos se muestran en la Figura 4. La organización, la mejora continua y los procedimientos son los elementos que más están presentes en las respuestas de los entrevistados. Por su parte, la estrategia organizacional y el involucramiento desde etapas tempranas son los elementos que menos están presentes para los entrevistados.

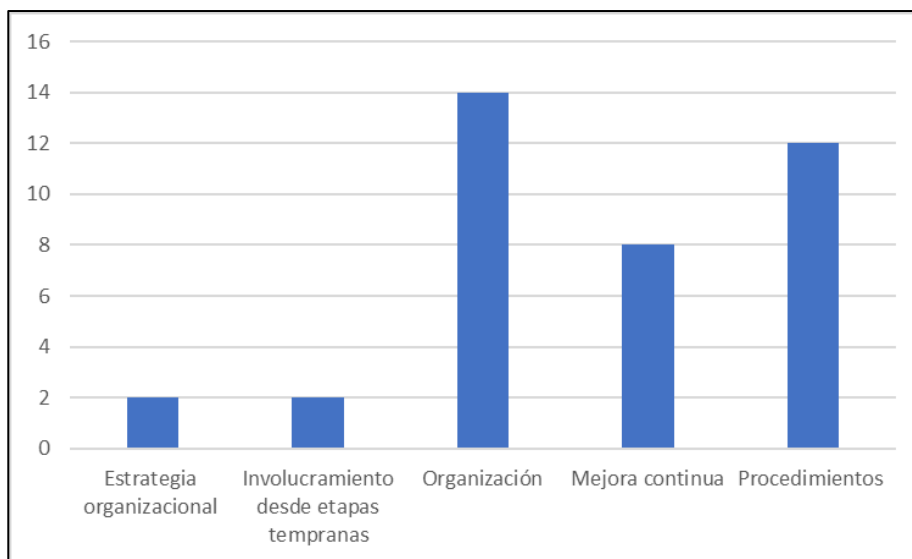


Figura 4. Análisis de patrones para la estructura del departamento de SQA.

4.2.3 Perfil del personal del departamento de SQA

En la entrevista se preguntó acerca del perfil que deben tener las personas que trabajan en un departamento de SQA. Hecho el análisis de identificación de patrones, los resultados obtenidos se muestran en la Figura 5. La educación y las habilidades blandas son las características que más están presentes en las respuestas de los entrevistados. Por su parte, la experiencia es bien valorada, pero no tanto en comparación con las otras dos características.

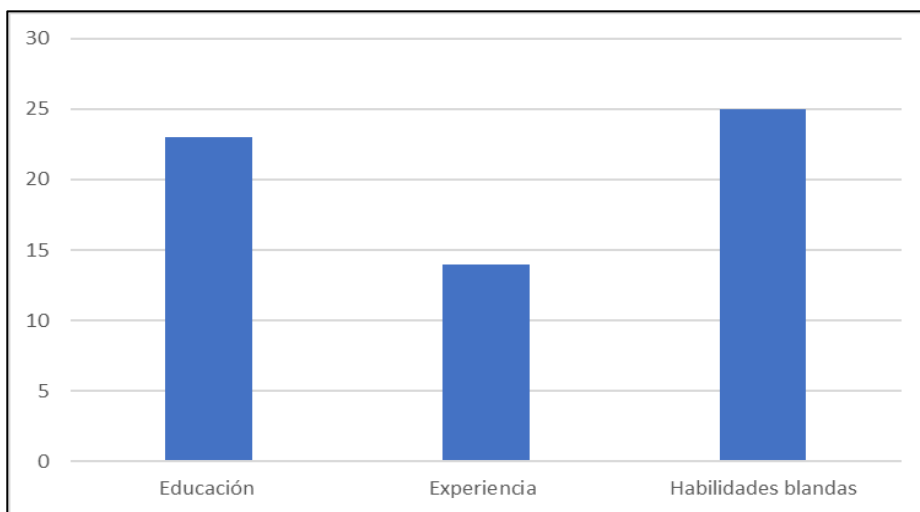


Figura 5. Análisis de patrones para el perfil del personal del departamento de SQA.

4.2.4 Implementación del departamento de SQA

Se le preguntó al entrevistado acerca de cómo considera que se debería de implementar un departamento de SQA. Hecho el análisis de identificación de patrones, los resultados obtenidos se muestran en la Figura 6. La organización es el elemento que más está presente en las respuestas de los entrevistados, mientras que la mejora continua y la estrategia organizacional son los elementos que menos están presentes para los entrevistados.

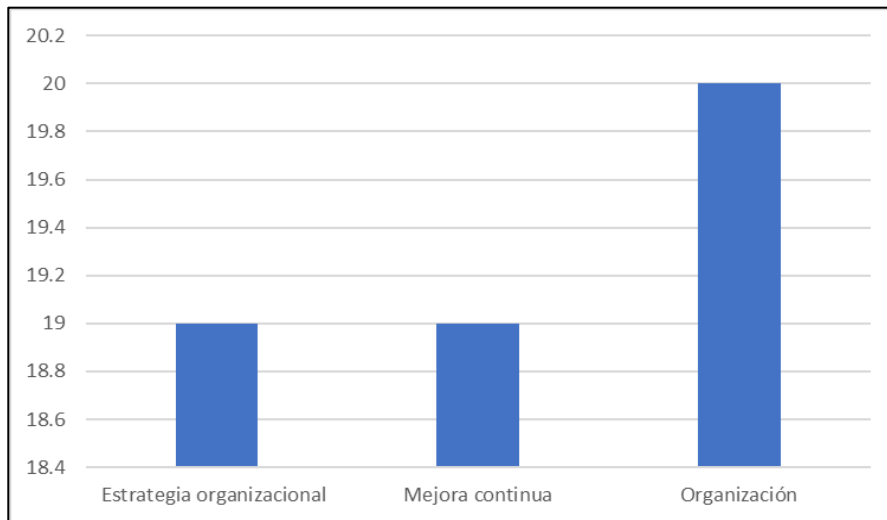


Figura 6. Análisis de patrones de cómo implementar el departamento de SQA.

4.2.5 Evaluación del departamento de SQA

En la entrevista se indagó sobre cómo se debe evaluar el departamento de SQA. Hecho el análisis de identificación de patrones, los resultados obtenidos se muestran en la Figura 7. Los procedimientos y métricas son el elemento que más está presente en las respuestas de los entrevistados. Por su parte, la mejora continua es el elemento que menos está presente para los entrevistados.

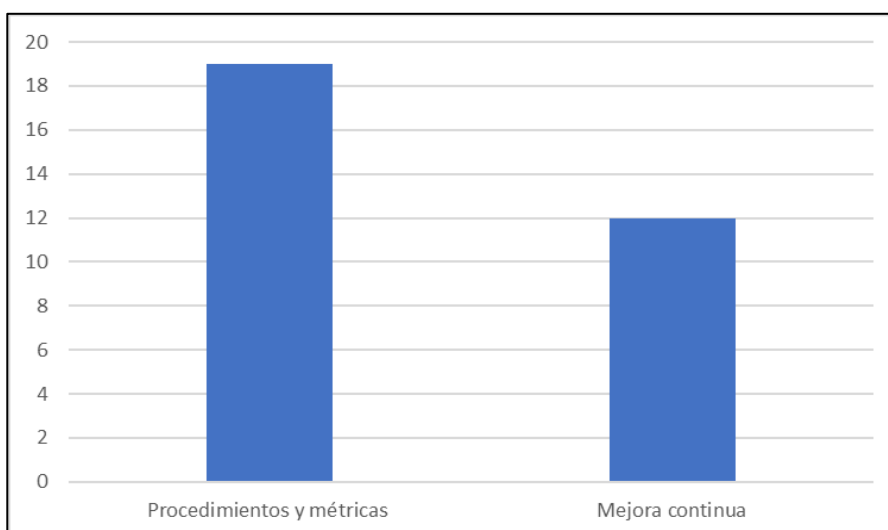


Figura 7. Análisis de patrones de cómo evaluar el departamento de SQA.

4.2.6 Desarrollo del departamento de SQA.

En la entrevista se preguntó acerca de cómo desarrollar un departamento de SQA. Hecho el análisis de identificación de patrones, los resultados obtenidos se muestran en la Figura 8. La mejora continua es el elemento que más está presente en las respuestas de los entrevistados. Por su parte, la estrategia y la organización son los elementos menos presentes.

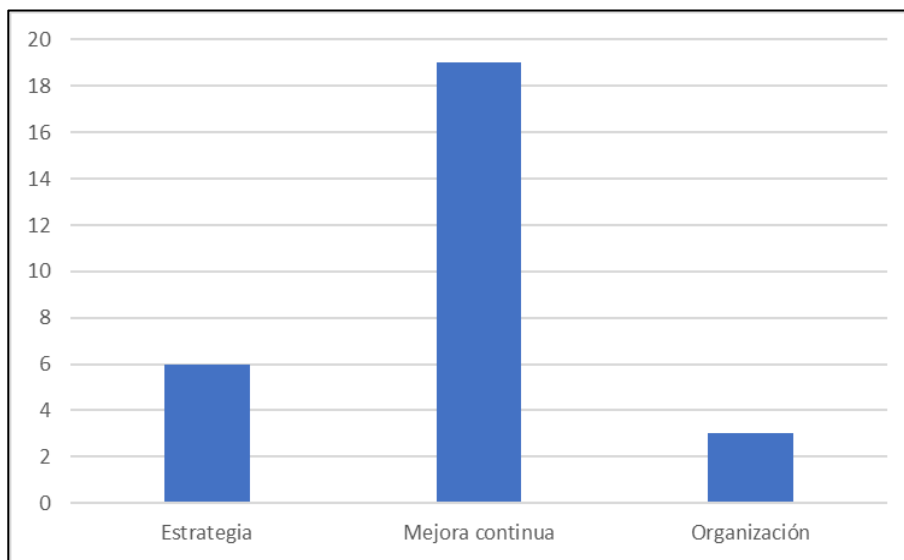


Figura 8. Análisis de patrones de cómo desarrollar el departamento de SQA.

4.2.7 Consideraciones adicionales en la implementación del departamento de SQA

Se les preguntó a los entrevistados acerca de recomendaciones adicionales para implementar un departamento de SQA. Hecho el análisis de identificación de patrones, los resultados obtenidos se muestran en la Figura 9. La creatividad e innovación y la promoción de una cultura de calidad son los elementos que más están presente en las respuestas de los entrevistados. Por su parte, la estrategia, la mejora continua y el plan de reconocimientos son los elementos que menos está presente para los entrevistados.

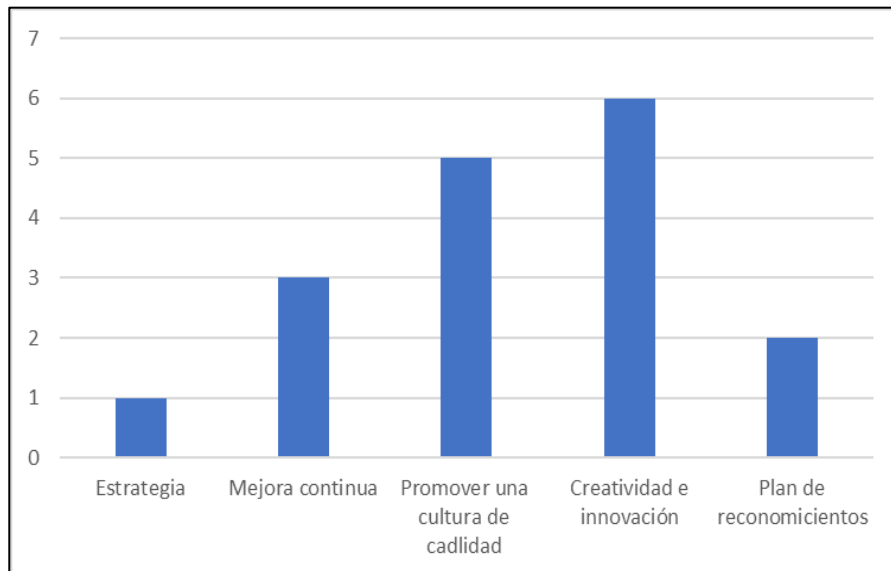


Figura 9. Análisis de patrones de consideraciones adicionales.

4.3 Observación

La tercera técnica utilizada fue la observación. Se llevaron a cabo 10 observaciones del departamento de SQA en dos empresas de software diferentes. (Las anotaciones completas de dichas observaciones se muestran en el Anexo 2: Observaciones). A continuación, presentamos un resumen de cada observación y su correspondiente análisis.

- Observación 1: Esta observación está relacionada con la jerarquía y organización del departamento de calidad.

El departamento de QA cuenta con una jerarquía bien definida, donde existe la figura del jefe de pruebas, pero no con un gerente de pruebas como en otras organizaciones en las que he trabajado anteriormente. El líder del departamento está a cargo de seis ingenieros de pruebas (QAs), los cuales están involucrados en todos los proyectos de la organización. Hay un QA por proyecto. Coincidentemente, en el capítulo II de este trabajo final de graduación, se hace mención de la existencia de varios roles dentro del proceso del aseguramiento de la calidad del software y los principales roles que debe contar todo departamento de SQA dentro de una empresa de desarrollo de software son el jefe de pruebas y los ingenieros de pruebas (conocidos en inglés como QA o testers). Las

actividades y tareas desarrolladas por estos dos roles dependen del contexto del proyecto, del producto y de las competencias de las personas en los roles y dentro de la organización (ISTQB, 2018).

- Observación 2: Esta observación está relacionada con las métricas del departamento de SQA.

Se lleva a cabo el proceso de registro de una de las métricas internas, la cual es el registro de horas trabajadas en una determinada asignación. Esto se hace a través de la herramienta de Jira¹. Dicha herramienta es utilizada por el equipo completo (QA's, desarrolladores y demás involucrados), registran el tiempo que les toma resolver cada una de las tareas asignadas. Coincidentemente en las entrevistas realizadas se indagó sobre cómo se debe evaluar un departamento de calidad (ver apartado 4.2.5). En las entrevistas se identificó que las métricas son el elemento que más está presente en las respuestas de los entrevistados (ver Figura 7).

- Observación 3: Esta observación está relacionada con la configuración de ambientes.

Configuración del ambiente de pruebas. Los encargados de tener los ambientes de prueba actualizados son el equipo de QA, los desarrolladores no les facilitan estos ambientes. Coincidentemente, en el capítulo II de este trabajo final de graduación, en la Tabla 3 responsabilidades, se menciona que una de las tareas que debe llevar a cabo un QA es el de diseñar, configurar y verificar los entornos de prueba (ISTQB, 2018).

- Observación 4: Esta observación está relacionada con el proceso de la creación de la estrategia de control de calidad.

El líder de QA y el QA se reúnen para tomar decisiones y planear la estrategia de SQA que van a aplicar al nuevo proyecto, analizan diversos temas como: el tamaño del proyecto, las métricas, los requerimientos, la documentación, el

¹ Software para la gestión de proyectos, seguimiento de errores e incidencias
<https://www.atlassian.com/es/software/jira>.

ambiente, el tipo y el alcance de las pruebas. Coincidentemente, en el capítulo II de este trabajo final de graduación, se hace referencia a que el proceso de SQA consta de una variedad de fases con actividades específicas, las cuales incluyen diferentes actividades como planificación, revisiones formales, pruebas, control de la documentación, análisis e informes (Presman, 2005).

- Observación 5: Esta observación está relacionada con el proceso de entrevistas de contratación de personal para el departamento de SQA.

Se requiere contratar a otro QA. En esta entrevista están presentes varios miembros con bastante experiencia del equipo. Al entrevistado se le hacen preguntas relacionadas con sus estudios, habilidades blandas, experiencia y por qué desea cambiar de trabajo. Nuevamente en sincronía con lo planteado en el capítulo II, se muestra como el proceso de SQA es más efectivo cuando se lleva a cabo a través de un equipo de profesionales adecuadamente cualificados (Godbole, 2004). Por ende, la importancia de contratar personal con las cualificaciones adecuadas para llevar a cabo dicho trabajo.

- Observación 6: Esta observación está relacionada con las reuniones para brindar el estatus de los proyectos.

Reunión semanal del departamento de SQA para dar estatus de sus proyectos. En esta reunión participa cada uno de los miembros del equipo respondiendo a preguntas relacionadas con la cantidad de pruebas que están pendientes de ejecutar, la cantidad de errores abiertos y los que se han tenido que reabrir de nuevo, el nivel de gravedad de estos, en que etapa se encuentran del proyecto, comentarios u observaciones adicionales. En el capítulo II de este trabajo final de graduación, se hace una pequeña referencia a esto, ya que los datos recopilados sobre el rendimiento de calidad de los proyectos de software, incluidos los problemas, permiten que la función SQA identifique oportunidades de mejora de procesos y planifique su implementación. Sin embargo, la manera en que esta organización está llevando esto a cabo es por el uso de una metodología ágil, tema que no está referenciado a lo largo de este trabajo (Galín, D. 2018).

- Observación 7: Esta observación está relacionada con la mejora continua del departamento de calidad.

Están en clases, se están preparando para la certificación de dispositivos móviles de ISTQB, reciben clases una vez a la semana en horario laboral. La empresa les provee el profesor y si pasan el examen se les hace un reembolso por el monto total del costo de la certificación, en caso de perderlo el empleado se debe hacer responsable de pagarla. A diferencia de otras observaciones, esta no ha sido mencionada o está directamente relacionada con el capítulo II de este trabajo final de graduación, pero sí en algunas de las respuestas de las entrevistas realizadas en el apartado 4.2.6, en las entrevistas realizadas se indagó sobre cómo desarrollar un departamento de calidad (ver apartado 4.2.7). En dichas entrevistas se identificó la mejora continua como el elemento que está más presente en las respuestas de los entrevistados (ver Figura 9).

- Observación 8: Esta observación está relacionada con la motivación.

La empresa motiva a los empleados a nominar a los compañeros que hacen bien el trabajo, generando un ambiente de reconocimiento y recompensa. No se permite que los empleados se auto nominen. Este tipo de observación no ha sido mencionada o está directamente relacionada con el capítulo II de este trabajo final de graduación, sin embargo, en las entrevistas realizadas se indagó sobre recomendaciones a tomar en cuenta para la implementación de un departamento de calidad (ver apartado 4.2.7). En las respuestas se menciona el plan de reconocimientos como un elemento a tomar en cuenta por parte de los entrevistados (ver Figura 9).

- Observación 9: Esta observación está relacionada con los requerimientos del proyecto.

El equipo completo que va a llevar a cabo un nuevo proyecto se reúne con el business analyst (BA) o en español analista de negocios para revisar los requerimientos completos del proyecto. Coincidentemente, en el capítulo II de este trabajo final de graduación, se menciona que todas las partes interesadas en el proyecto deben participar activamente en las revisiones o aclaración de

consultas relacionadas con los requerimientos o la documentación (ISTQB Expert level, 2011).

- Observación 10: Esta observación está relacionada con la documentación que realiza el departamento de calidad.

Se puede ver la importancia que tiene la documentación en la organización, utilizan la herramienta Confluence² para llevar un registro de los procesos que deben ser aplicados en el departamento de SQA. Coincidentemente, en el capítulo II de este trabajo final de graduación, donde se hace mención que una de las principales actividades que debe tener un proceso de SQA es la de definir, establecer y documentar los procesos y estándares de SQA para cada uno de los proyectos (IEEE 2014).

² Software de colaboración en equipo <https://support.atlassian.com/confluence-cloud/>.

Capítulo V. Solución del problema

En este apartado se describe la propuesta de solución planteada, la cual se encuentra detallada de forma íntegra en el apéndice 3. Dicha propuesta es el resultado de la aplicación de las técnicas de investigación utilizadas en este trabajo, es decir, de la revisión bibliográfica, el proceso de observación hecho a departamentos de SQA en varias organizaciones, además de experiencias profesionales y la aplicación de entrevistas. En el caso específico de la revisión bibliográfica se consideraron, entre otros, documentos relevantes de referencia para la propuesta de solución, tales como estándares y normas de instituciones emblemáticas como lo son el Instituto de ingenieros eléctricos y electrónicos (IEEE), la Organización Internacional de Normalización (ISO) y la Junta Internacional de Calificaciones de Pruebas de Software (ISTQB). A continuación, se presenta un resumen de la propuesta de solución y el proceso de prueba y resultados. Por su naturaleza, en este capítulo se prescinde del desarrollo de la solución y el procedimiento de implementación, puesto que este trabajo final de graduación consiste en una propuesta específica que no considera, dentro de su alcance, su desarrollo o implementación.

5.1 Propuesta de Solución

La primera sección de la propuesta desarrollada es la introducción. En esta se describe el propósito del documento, el cual es servir como un documento guía que permita implementar un departamento de Aseguramiento de la Calidad del Software. Este tipo de departamentos es usualmente conocido como SQA (por las siglas en inglés de Software Quality Assurance). El objetivo principal de este departamento es garantizar a la organización que sus procesos y productos de software cumplan con los estándares de calidad, confiabilidad y funcionalidad requeridos por los usuarios.

Luego de la introducción, la propuesta de solución presenta el marco organizacional. Este contiene siete aspectos puntuales. El primero de esto son los objetivos de SQA, los cuales se recomiendan que estén alineados con los objetivos de la organización. Se debe tomar en consideración que los objetivos son declaraciones específicas y cuantificables que describen lo que se desea lograr en términos de calidad de software, esto permitirá continuar con la calidad del proceso y del producto. Seguidamente, dentro de este mismo marco organizacional, se presenta como segundo aspecto lo relativo a la gestión. Se describen las principales actividades y elementos involucrados en el proceso y cómo se relacionan estos con el perfil, las tareas y responsabilidades de los miembros del departamento en relación con el aseguramiento y el control de la calidad del software.

El tercer aspecto del marco organizacional es la documentación. La cual se refiere a la creación, gestión y mantenimiento de documentos relacionados con los procesos de calidad de software dentro de la organización. En el mismo apartado se identifica y describe la documentación mínima que debe regir en el departamento SQA.

Seguidamente, como cuarto aspecto se encuentran los estándares, normas, guías y comités que se aplican para garantizar un departamento de SQA. Estos permiten que la organización establezca pautas y criterios que les ayuden a cumplir con las especificaciones y requerimientos del software.

Por su parte, las métricas son el quinto aspecto. Estas describen las principales medidas cuantitativas o cualitativas que se utilizan para evaluar y mejorar los diferentes aspectos relacionados con la calidad de software.

El sexto aspecto de este marco es el de las pruebas. Las cuales describen las principales pruebas para evaluar y verificar la calidad del software.

Finalmente, el séptimo y último aspecto de este marco organizacional aborda el tema de las herramientas. Las cuales describen la importancia y tipos de herramientas que ayudan tanto al equipo de desarrollo como al equipo de pruebas en las actividades relacionadas con el aseguramiento y control de calidad del software.

Otro aspecto muy importante de la propuesta es el de los costos y beneficios.

La conclusión de la propuesta incluye aspectos como la similitud que existe al implementar un departamento de SQA independientemente de factores como el tamaño, el tipo de negocio o industria y la importancia de este para la organización en sí.

La propuesta finaliza con la bibliografía, en la cual se enlistan las fuentes bibliográficas consultadas que ayudaron a complementar dicha propuesta de solución.

5.2 Proceso de prueba y resultados

Para comprobar la viabilidad de la propuesta de solución planteada, esta fue sometida al criterio de expertos mediante el uso de un instrumento de evaluación par, el cual se detalla de forma íntegra en el apéndice 2.

Como se menciona en el capítulo III, se diseñó este instrumento con el fin de sistematizar uniformemente la evaluación de la propuesta, además de obtener retroalimentación y recomendaciones de mejora de parte de los expertos relacionados con la gestión de SQA.

El instrumento de revisión par y la propuesta fueron enviadas a los evaluadores el lunes 9 de octubre del presente año por medio de correo, los mismos tenían una semana de tiempo para realizar dicha evaluación. La medición de los resultados presentados en el instrumento de revisión se hizo utilizando la escala de Likert, esta escala de medición es comúnmente utilizada en investigaciones y encuestas, ya que permiten determinar el nivel de acuerdo o desacuerdo de evaluadores (Da Silva, 2023).

La propuesta se envió a dos evaluadores. El primer evaluador, José Chacón egresado de la maestría en tecnologías de información con énfasis en productos y servicios TIC de la Universidad Nacional, cuenta con más de 15 años de experiencia como ingeniero de pruebas. El segundo evaluador, Mario Cordero egresado de la maestría en tecnologías de información con énfasis en administración de proyectos de la Universidad Nacional, cuenta con más de 10 años de experiencia como ingeniero de pruebas.

Los resultados obtenidos de parte de los expertos para la propuesta de solución son los siguientes. El primer evaluador José Chacón, clasificó la mayoría de los atributos de la propuesta con 5 que según la escala de Likert esto hace referencia a que se encuentra muy de acuerdo con que la propuesta realmente establece un plan para crear un departamento estándar de SQA, está puede usarse para controlar y asegurar el proceso de la gestión de calidad, ya que esta establece los procesos de SQA, además de que la propuesta permite medir la efectividad de los procesos, sugerir mejoras y garantizar la conformidad y aceptabilidad del producto por parte del usuario final. Finalmente, él considera que si implementara dicha propuesta en su lugar de trabajo. Sin embargo, él se encuentra en desacuerdo con que la propuesta define los roles, responsabilidades y las prácticas de la función de SQA, ya que un departamento de SQA no necesariamente tiene que ser liderado por un QA manager o un QA lead, que existen organizaciones donde el jefe de desarrollo es el mismo jefe para los ingenieros de pruebas. Además de que en las metodologías ágiles los ingenieros de pruebas son parte del equipo de desarrollo y no tienen una estructura de managers o líderes de QA. Por otro lado, considera que, en la sección de herramientas, se debería de abordar el tema de las herramientas automatización.

El segundo evaluador, Mario Cordero calificó todos los atributos de la propuesta con 5 que según la escala de Likert él se encuentra muy de acuerdo con que la propuesta realmente establece un plan para crear un departamento estándar de SQA, que esta puede usarse para controlar y asegurar el proceso de la gestión de calidad, ya que esta establece los procesos de SQA, además de que la propuesta permite medir la efectividad de los procesos, sugerir mejoras y garantizar la conformidad y aceptabilidad del producto por parte del usuario final. Y al igual que José, él también implementaría dicha propuesta en su lugar de trabajo. Por su parte Mario considera que se debe contemplar el correcto uso de los términos establecer e implementar dentro de la propuesta, que sería buena idea asociar los entregables a cada una de las diferentes etapas dentro del proceso y que es importante agregar una etapa de cierre de pruebas, donde se finiquiten resultados finales.

A pesar de que ambos evaluadores aplicarían la propuesta en sus lugares de trabajo, ambos recomendaron aprobar la propuesta con correcciones. Las evaluaciones íntegras se pueden observar en los apéndices 3 y 4.

Capítulo VI. Análisis financiero

En este apartado se detallan los principales costos y beneficios (cualitativos) relacionados con la propuesta de implementación del departamento de SQA.

Realizar un adecuado análisis financiero en lo que respecta a la calidad del software puede ser un tema complejo, más como en este caso que, por razones de limitación de tiempo y recursos, se decidió desde el inicio no contemplar el proceso de desarrollo e implementación de la propuesta en una organización en particular. Existen muchísimos factores claves que influyen en este tipo de análisis. Algunos de estos son: la gran variedad de pruebas de software posibles, los entornos de pruebas y el impacto que causa la calidad de software según el tipo de industria, solo por mencionar algunos. En un trabajo futuro se podría implementar la propuesta de solución en una organización y en base en este futuro desarrollo evaluar adecuadamente aspectos como el análisis de retorno de la inversión (ROI), el valor actual neto (VAN) o la tasa interna de retorno (TIR), entre otros.

No obstante, con el fin de brindar una idea desde el punto financiero sobre la propuesta de implementación del departamento de SQA en empresas desarrolladoras de software, se podrían desarrollar aspectos financieros tales como incorporar un análisis de costos/beneficios desde el Project Charter utilizando el ROI como un insumo para los inversionistas. Esto permitiría determinar si una implementación de la propuesta de solución es financieramente viable para la organización. Un Project Charter proporciona a las partes interesadas una idea clara de los objetivos, alcance y responsabilidades del proyecto (Farag, 2021). El Project Management Institute (PMI) define el retorno de la inversión (ROI) como una métrica financiera utilizada para indicar la rentabilidad y eficiencia de un proyecto, a través de comparar la ganancia o pérdida de un proyecto con respecto a su costo, la fórmula del ROI es:

$$ROI = \frac{\text{Beneficios Netos} - \text{Costos Totales}}{\text{Costos Totales}} * 100$$

6.1 Beneficios

Nguyen (2023), menciona que algunos de los principales beneficios de contar con un departamento de SQA en una empresa de desarrollo de software son:

- Permite ahorrar dinero: Sin duda alguna los errores son costosos. Si una empresa lanza un software defectuoso, dependiendo de la gravedad de esta, tendrá que desechar el software o crear correcciones.
- Reduce el retrabajo: Esto permite eliminar los costos asociados al desarrollo requerido para corregir los defectos y el costo de volver a probar estos.
- Mejora el proceso de desarrollo de software: Se toma más tiempo corregir un error en producción que corregirlo en una etapa temprana.
- Promueve la mejora continua dentro de la organización: Al monitorear y evaluar periódicamente el desempeño, las organizaciones pueden identificar áreas de mejora e implementar acciones correctivas. Esto permite a las organizaciones adaptarse tanto a las necesidades, como a las tendencias y a los avances tecnológicos.
- Facilita la estandarización de procesos: Los procesos estandarizados reducen la variabilidad, minimizan los riesgos de defectos. También simplifican la capacitación, la incorporación y la transferencia de conocimientos, lo que resulta en una mayor eficiencia de los procesos y una menor dependencia de las habilidades o experiencia individuales.
- Mejora la experiencia del usuario: A los usuarios les satisface interactuar con productos de software libre de defectos, seguros, usables y que superen sus expectativas.
- Posicionamiento en el mercado: El software de alta calidad coloca a las empresas en una posición de mercado más fuerte y dominante.
- Reduce los costos de mantenimiento: Al obtener un software de alta calidad a la primera, así la organización no tendrá que preocuparse más de este.

- Disminuye gastos importantes: Al contar con un departamento de SQA le permite a la organización disminuir la probabilidad de causar daños graves al usuario, la pérdida de confianza, la mala reputación y las posibles sanciones hacia la organización.

6.2 Costos

Existen varios tipos de costos asociados al departamento de SQA, los principales son:

6.2.1 Tiempo

El período de tiempo de implementación de un departamento de SQA puede variar significativamente de una organización a otra y este es un factor importante que se debe tomar en cuenta, a este punto le afectan características como el tamaño, la madurez y el tipo de software, entre otros. Según diferentes fuentes consultadas en una empresa mediana, se dice que en un periodo de 6 a 8 meses se puede llevar a cabo esta implementación. Para llevar a cabo este ejercicio, se partirá del supuesto de que se tardan 6 meses en implementar el departamento de SQA en una organización de desarrollo de software de tamaño mediano.

6.2.2 Recurso humano

El personal necesario para probar un software depende de muchos factores como el tipo de software, la complejidad, lo que se necesita probar de este y la cantidad de desarrolladores. Se habla de un ingeniero de pruebas por cada 5 o 6 desarrolladores, esto partiendo del principio de que los desarrolladores realicen pruebas unitarias y de integración.

Dentro de este costo se incluyen los salarios, capacitaciones y cualquier otro gasto relacionado con el recurso humano del departamento de SQA.

Se estima que los ingenieros de pruebas asignados a este proyecto hagan uso de 40 horas semanales para un total de 160 horas mensuales. Mientras que el gerente de proyectos y el jefe de desarrollo dedicaran 15 horas semanales para un total de 60 horas mensuales cada uno. Dentro de estos costos se incluyen tanto los salarios mensuales como las respectivas cargas patronales, según el sitio web de la Caja Costarricense del

Seguro Social (CCSS) este último corresponde al 1.26% para el patrono. En la Tabla 5, se muestra la estimación del costo del recurso humano para este proyecto.

Tabla 5. Costo de recurso humano. Elaborado a partir de ERI Economic Research Institute y el sitio web de CCSS, 2023

ítem	Costo por hora
Recurso humano:	
• 3 Ingenieros de pruebas	\$31.50 c/u
• 1 Líder de QA	\$37.80
• 1 Gerente de proyectos	\$37.80
• 1 Jefe técnico	\$45.36

6.2.3 Hardware y software

Los equipos de hardware y software, como las herramientas de prueba, de gestión y otros recursos de software utilizados por el departamento de SQA, deben ser costos tomados en consideración al momento de llevar a cabo la propuesta de implementación. Como ejemplo para este ejercicio se toma en consideración la herramienta de gestión de Jira y se parte del principio que se otorgue una computadora nueva para cada uno de los 3 ingenieros de prueba y el líder de pruebas. En la Tabla 6, se muestra la estimación de costos del hardware y software.

Tabla 6. Costos de hardware y software. Elaborado a partir del sitio web de JIRA y Apple, 2023.

ítem	Costo por hora
Herramienta gestión de pruebas de software (\$ 8.17 mensuales x 6 m x 6 usuarios)	\$0.05
Computadoras	\$ 14.58

6.2.4 Costos fijos

Estos costos no son tomados en consideración. Se parte de la hipótesis que se continúe trabajando de manera remota y la organización no cuente con una oficina.

La Tabla 7 ejemplifica el detalle de los principales costos relacionados con la propuesta de implementación del departamento de SQA en una empresa de software durante los 6 meses que podría tomar llevar a cabo dicha implementación. Cabe mencionar nuevamente que esto es solamente un ejemplo para ilustrar el tema del análisis financiero, ya que este trabajo final de graduación es una propuesta y no está siendo implementado por ninguna organización en este momento.

Tabla 7. Detalle de los principales costos. Elaboración propia.

Mes	1	2	3	4	5	6
Costos	\$ 26,245.38	\$ 26,245.38	\$ 26,245.38	\$ 26,245.38	\$ 26,245.38	\$ 26,245.38
Recurso humano						
Ingeniero de pruebas (1)	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00
Ingeniero de pruebas (2)	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00
Ingeniero de pruebas (3)	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00	\$ 5,040.00
Lider de QA	\$ 6,048.00	\$ 6,048.00	\$ 6,048.00	\$ 6,048.00	\$ 6,048.00	\$ 6,048.00
Gerente de proyectos	\$ 2,268.00	\$ 2,268.00	\$ 2,268.00	\$ 2,268.00	\$ 2,268.00	\$ 2,268.00
Jefe técnico	\$ 2,721.60	\$ 2,721.60	\$ 2,721.60	\$ 2,721.60	\$ 2,721.60	\$ 2,721.60
Hardware	\$ 87.48	\$ 87.48	\$ 87.48	\$ 87.48	\$ 87.48	\$ 87.48
Software	\$ 0.31	\$ 0.31	\$ 0.31	\$ 0.31	\$ 0.31	\$ 0.31

Capítulo VII. Conclusiones y recomendaciones

7.1 Conclusiones

En este trabajo final de graduación se desarrolló una propuesta para implementar el departamento de SQA en una empresa de software. Esta propuesta fue sometida a un proceso de evaluación por expertos que proporcionaron retroalimentación sobre la misma para garantizar la calidad, la precisión y la viabilidad de poner la misma en práctica. A continuación, se detallan las conclusiones asociadas a cada objetivo de la investigación.

1. El objetivo principal planteado al inicio de este trabajo final de graduación consistía en diseñar una propuesta que permitiera implementar un departamento de SQA en una empresa de desarrollo de software. En este sentido, se diseñó una propuesta por medio de un análisis teórico - práctico con algunas de las mejores prácticas disponibles.
2. En cuanto a los objetivos específicos. El primero de estos objetivos se enfoca en identificar las formas en cómo las organizaciones de desarrollo de software organizan los procesos de SQA. A través de una exhaustiva revisión de literatura en temas relacionados con la identificación de los procesos de SQA, se pudo establecer conceptos de calidad de software tales como qué es aseguramiento de calidad, control de calidad y pruebas de software. También se mostró la relación existente entre la organización y el departamento de aseguramiento de la calidad de software, incluyendo los roles y las responsabilidades de dicho departamento. El proceso del aseguramiento de la calidad de software en relación con la implementación de las actividades del proceso de SQA, el aseguramiento del producto y del proceso. Esto es importante porque sirvió como insumo para desarrollar la propuesta de implementación del departamento de SQA en una empresa de software. En conclusión, esto permitió resumir los principales conceptos utilizados, proporcionar orientación para la investigación, las tendencias y prácticas actuales.
3. El segundo objetivo específico, se enfoca en el análisis sobre los principales aspectos relacionados con la organización operativa del departamento de SQA. Para esto se desarrollaron entrevistas y un proceso de observación. En el caso de las entrevistas, se creó un guion con preguntas previamente diseñadas con el fin de evitar el sesgo de hacer preguntas diferentes en cada entrevista. En total se

realizaron 12 entrevistas, las cuales fueron aplicadas a sujetos que cumplían con los parámetros definidos previamente. Respecto a la observación, esta se llevó a cabo en los departamentos de SQA de dos organizaciones. El uso de ambas técnicas es importante porque permitieron obtener una perspectiva más completa y detallada de cómo está organizado y opera el departamento de SQA dentro de una organización. Al hacer una validación cruzada de datos entre las entrevistas y las observaciones, se concluye que muchas de las respuestas dadas por los entrevistados estaban directamente relacionadas con lo descrito por varios de los estándares, organizaciones y marcos de trabajo consultados durante esta investigación. Lo cual sugiere que existe un estrecho vínculo entre la práctica actual de la organización operativa de SQA y las pautas establecidas por los estándares de calidad de software consultados.

4. El tercer objetivo específico se enfoca en la propuesta metodológica de cómo implementar un departamento de SQA. Por medio de una revisión sistemática de literatura y la experiencia práctica de entrevistas y observaciones, se sintetizó la información obtenida de estas tres técnicas. Esto es importante porque permitió desarrollar una guía detallada y fundamentada que proporciona directrices y prácticas de cómo implementar un departamento de SQA en empresas desarrolladoras de software. En conclusión, el uso de esta guía puede tener un impacto significativo en la eficacia y éxito de las empresas desarrolladoras de software. Ya que este departamento es esencial para garantizar la calidad de los productos de software, reducir defectos, mejorar la eficiencia y la satisfacción del cliente.

5. El cuarto objetivo específico, se enfoca en la validación y retroalimentación de la propuesta metodológica por parte de expertos en el tema. Después de desarrollar una propuesta detallada para la implementación del departamento de SQA, la cual proporciona de forma detallada y fundamentada las principales directrices y prácticas de como implementar un departamento de SQA. Se presentó la propuesta de solución a los expertos con el fin de obtener sus opiniones y recomendaciones. Esto es importante, ya que esta propuesta está siendo validada a través de la perspectiva de expertos, permitiendo identificar posibles mejoras, aumenta la credibilidad y calidad de la misma. En conclusión, ambos evaluadores comentan que aplicarían la propuesta de solución presentada en sus lugares de trabajo, puesto que es una propuesta estándar que sin importar el tipo

de software que se desarrolle puede ser implementada. Sin embargo, se debe tomar en consideración que ambos expertos recomendaron aprobar la propuesta con correcciones, las cuales se basan en ampliar más ciertas secciones de la propuesta como tal, como se explica en el capítulo V, en la sección de pruebas y resultados.

6. Finalmente, la última conclusión de este trabajo final de graduación está relacionada con la importancia de contratar personal que posea amplia experiencia en el tema investigado para dirigir el departamento de SQA.

7.2 Recomendaciones

En este subapartado se detallan las recomendaciones finales después de aplicar todos los instrumentos y técnicas previstas.

1. El aseguramiento de la calidad se refiere a un proceso repetible y bien definido que permite revisar los mecanismos de control interno, asegurar el cumplimiento de los estándares y procedimientos de software. El objetivo de este proceso es asegurar la conformidad con los requerimientos, ejecutar actividades de prueba para reducir el riesgo, evaluar los controles internos y mejorar la calidad mientras se ajusta al cronograma y presupuesto establecido (Owens & Khazanchi, 2009). La calidad del software es un esfuerzo continuo y de toda la organización. Dicho esto, se recomienda a las organizaciones de desarrollo de software integrar este proceso en todos los aspectos y a lo largo del ciclo de vida del desarrollo de software.
2. Se recomienda a las organizaciones que deseen implementar dicha propuesta que incluyan dentro de su presupuesto elementos relacionados con la capacitación continua y certificaciones, con el fin de garantizar profesionales actualizados con las últimas tendencias y mejores prácticas en el campo de la calidad de software. Esto conduce a un mejor desempeño en los roles, lo que a la vez permite aumentar la confianza de los clientes y cumplir con diferentes estándares y regulaciones.

3. Se recomienda capacitar e incentivar a la organización de manera holística acerca de la importancia de la calidad de software, ya que este desempeña un papel fundamental en el producto final.

Capítulo VIII. Análisis retrospectivo

El análisis retrospectivo de este trabajo final de graduación toma la forma de recapitulación sobre las limitaciones que se presentaron durante el desarrollo de este. Además de una observación personal planteada para la universidad.

La primera limitación de esta investigación es que no se contaba con una organización para llevar a cabo dicha implementación. En este sentido, para gestionar esta limitación se procedió a utilizar el criterio de expertos.

La segunda limitación estuvo relacionada con el análisis financiero, al no contar con una organización para implementar dicha propuesta, es difícil determinar si es financieramente viable para la organización. Al respecto se procedió a consultar con expertos, los cuales mencionaron que la única manera era haciendo un ejercicio basado en supuestos.

La tercera limitación presentada en esta investigación está relacionada con las entrevistas y la observación, ya que al utilizar ambas técnicas están condicionadas con el tiempo, el acceso a diferentes organizaciones, personas y las pocas organizaciones tomadas como referencia. Dichas limitaciones fueron subsanadas al triangular las otras técnicas e instrumentos utilizados a través de este trabajo final de graduación.

Se invita a la Universidad Nacional, considerar los resultados de este trabajo final de graduación como referencia para futuras investigaciones en las cuales se pueda llevar a cabo la implementación de la propuesta del departamento de SQA en empresas desarrolladoras de software.

Bibliografía

American Society for Quality (2023). What Is A Quality?. Consultado el 29 de mayo 2023. <https://asq.org/quality-resources/quality-glossary>

Axelsson, J., & Skoglund, M. (2016). Quality assurance in software ecosystems: A systematic literature mapping and research agenda. *Journal of Systems and Software*, 114, 69-81.

Bohgard, M., Karlsson, S., Lovén, E., Mikaelsson, L., Mårtensson, L., Osvalder, A. Ulvengren, P. (2015). *Arbete och teknik på människans villkor (Vol. 3)*. Stockholm: Prevent.

Bourque, P., & Fairley, R. E. (2014). *SWEBOK v3. 0: Guide to the software engineering body of knowledge*. IEEE Computer Society, 1-335.

Cardozzo, D. R. (2016). *Desarrollo de software: requisitos, estimaciones y análisis*. IT Campus Academy.

CCSS | Calculadora Patronal. (2023). <https://www.ccss.sa.cr/calculadora>

Choy, L. T. (2014). The strengths and weaknesses of research methodology: Comparison and complimentary between qualitative and quantitative approaches. *IOSR Journal of Humanities and Social Science*, 19(4), 99-104.

Chouhan, Rekha, and Rajeev Mathur. "Role of Software Quality Assurance in Capability Maturity Model Integration." *International Journal of Advanced Research in Computer Engineering & Technology* 1.6 (2012): 2278-1323.

Cortés Cortés, M. E., & Iglesias León, M. (2004). *Generalidades sobre Metodología de la Investigación*. Ciudad del Carmen: Universidad Autónoma del Carmen.

Da Silva, D. (2023). Qué es escala de Likert y cómo aplicarla. Zendesk MX. <https://www.zendesk.com.mx/blog/que-es-escala-de-likert/>

Everett, G. D., & McLeod Jr, R. (2007). *Software testing. Testing Across the Entire*.

E. W. Dijkstra, "Notes on Structures Programming", Technical University Eindhoven, The Netherlands, departamento de Matemáticas, Technical Report 70-WSK-03 1970, <https://www.cs.utexas.edu/users/EWD/ewd02xx/EWD249.PDF>

Farag, A. (2021). 4.4. Project Charter. *Essentials of Project Management*.

Flora, H. K., & Chande, S. V. (2014). A systematic study on agile software development methodologies and practices. *International Journal of Computer Science and Information Technologies*, 5(3), 3626-3637.

Galín, D. (2018). Establishing SQA Processes and Their Coordination with Relevant Software Processes.

Godbole, N. S. (2004). *Software quality assurance: Principles and practice*. Pragbourne, U.K.: Alpha Science.

Graham, D., van Veenendaal, E., Evans, I., & Black, R. (2008). *Foundations of Software Testing ISTQB Certification*. Cengage Learning Emea.

Hossain, M. (2018). Challenges of software quality assurance and testing. *International Journal of Software Engineering and Computer Systems*, 4(1), 133-144.

Griffiee, D. T. (2005). Research Tips: Interview Data Collection. *Journal of Developmental Education*, 28(3), 36-37.

Heimann, D. (2014). IEEE standard 730-2014 software quality assurance processes.

IEEE Computer Society, New York, NY, USA, IEEE Std, 730, 2014.

IEEE Std 730 (2014): *Software Quality Assurance*, The IEEE Computer Society, IEEE, Nueva York.

IEEE Std 1061. (1992): *for a Software Quality Metrics Methodology*, The IEEE Computer Society, IEEE, Nueva York.

Ingram, L., Hussey, J., Tigani, M., & Hemmelgarn, M. (2006). *Writing a literature review and using a synthesis matrix*. NC State University: NC State University Writing and Speaking Tutorial Service.

ISTQB, Expert Level. (2011). *Test Management (Managing Testing, Testers, and Test Stakeholders) version 2011*. International Software Testing Qualifications Board.

ISTQB, Foundation Level. (2018). *Foundation level syllabus version 2018*. International Software Testing Qualifications Board.

ISTQB. (2014). *Standard Glossary of Terms used in Software Testing*. Recuperado de <http://www.istqb.org/downloads/glossary.html>

Javed, A., Maqsood, M., Qazi, K. A., & Shah, K. A. (2012). How to improve software quality assurance in developing countries. *Advanced Computing*, 3(2), 17.

- Khraiwesh, M. (2014). Process and product quality assurance measures in CMMI. *International Journal of Computer Science and Engineering Survey*, 5(3), 1.
- Laaraib, E., Maher, Z. A., Solangi, Z. A., Koondhar, M. Y., Memon, M., Depar, M., & Shah, A. (2021). A Methodology for Incorporating Quality Assurance Practices during Software Development Life Cycle. *Int. J.*, 10, 2296-2301.
- Lewis, W. E. (2017). *Software testing and continuous quality improvement*. CRC press.
- Manocha, V. (2014) *Software Quality Assurance*.
- McCaffrey, J. (2019). What Makes A Good Software Tester?. 22/03/22
<https://docs.microsoft.com/en-us/archive/msdn-magazine/2008/december/%7Bend-bracket-%7D-what-makes-a-good-software-tester>
- Mora-Campos, A. (2015). Nuevas formas de revisión por pares en revistas científicas: revisión abierta/open review/open review. *PENSAR EN MOVIMIENTO: Revista de Ciencias del Ejercicio y la Salud*, 13(1), 1-4.
- Morales, F. (2010). *Tipos de investigación*. Bogotá DC.
- Nguyen, T. (2023). Quality assurance | Benefits & Importance - Rikkeisoft - Trusted IT solutions provider. Rikkeisoft - Trusted IT Solutions Provider.
<https://rikkeisoft.com/blog/importance-of-quality-assurance/>
- Owens, D. M., & Khazanchi, D. (2009). Software quality assurance. In *Handbook of research on technology project management, planning, and operations* (pp. 242-260). IGI Global.
- Paradis, E., O'Brien, B., Nimmon, L., Bandiera, G., & Martimianakis, M. A. (2016). Design: Selection of data collection methods. *Journal of graduate medical education*, 8(2), 263-264.
- Páramo Morales, D. (2015). La teoría fundamentada (Grounded Theory), metodología cualitativa de investigación científica. *Pensamiento & gestión*, (39), 1-7.
- Pressman, R. S. (6th Ed.) (2005). *Software engineering: A practitioner's approach*, Boston: McGraw Hill.
- Pressman, R. S (7th Ed.) (2010). *Ingeniería del Software. "Un Enfoque Práctico"*, México: McGraw-Hill.
- Dupuis, R., Bourque, P., Abran, A., Moore, J. W., & Tripp, L. (1999). *The SWEBOK Project: Guide to the software engineering body of knowledge*. Stone Man Trial Version, 1.
- Sahni, I., & Anand, A. (2014). Software quality assurance. *International Journal*, 2(6), 738-741.

Soberón, U. E. M., & Acosta, Z. (2009). Fuentes de información para la recolección de información cuantitativa y cualitativa.

Strauss AL, Corbin J. (1998). Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory, 2nd edn. Thousand Oaks, CA: Sage.

Tanner, K. (2018). Research Methods: Information, Systems, and Context.

Thayer, R. H., & Fairley, R. E. (2nd Ed.) (1999). software engineering project management: The silver bullets of software engineering. Software Engineering Project Management, (pp. 503-504). Los Alamitos, CA: IEEE Computer Society.

Walkinshaw, N. (2017). What Is Software Quality, and Why Does it Matter?. In: Software Quality Assurance. Undergraduate Topics in Computer Science. Springer, Cham. https://doi.org/10.1007/978-3-319-64822-4_2

Yang, H. (2012). Measuring software product quality with ISO standards base on fuzzy logic technique. Affective Computing and Intelligent Interaction, 59-67.

Apéndice

Apéndice 1: Preguntas para la entrevista

Parte 1: Datos demográficos

1.1 ¿Cuál es su nombre?

1.2 ¿Con cuál género se identifica?

Mujer

Hombre

Prefiero no contestar

1.3 ¿Cuál es su nivel de educación?

Bachiller Universitario

Licenciatura

Maestría

Doctorado

1.4 ¿Cuál es su área de trabajo?

Líder del departamento de calidad

Probador de software (tester)

Programador

Gerente de proyectos

1.5 ¿Tiene personal a cargo? Si es así indique ¿cuántas personas? En caso de que no agregue cero.

1.6 ¿Posee certificaciones? En caso de que así sea ¿cuáles? En caso de que no agregar n/a

1.7 ¿Cuántos años de experiencia tiene en TI?

5 años

6 - 9 años

10- 15 años

Más de 15 años

1.7 ¿Actualmente trabaja en una empresa que cuenta con un departamento de Calidad de Software (QA)?

Si

No

Parte 2: Departamento de calidad – implementación y desarrollo

2.1 ¿Cuál es la estructura que debe tener un departamento de calidad, en referencia a cuáles son las áreas que debe cubrir?

2.2. ¿Cuál considera usted que es el perfil que deben tener las personas que trabajan en esas áreas

2.3 ¿Cómo considera usted que se debe de implementar el departamento de SQA?

2.4 ¿Cómo se debe evaluar este departamento, para saber que lo que se está haciendo, está bien?

2.5 Basado en su experiencia, ¿cómo desarrollaría ese departamento, para hacer que se mantenga dentro de la organización, generar un crecimiento a nivel profesional y educacional de los profesionales que lo conforman?

2.6. ¿Algún comentario adicional que desee hacer respecto a cómo implementar un departamento de SQA en una empresa de software?

Apéndice 2: Instrumento de revisión par

Instrumento de revisión par

Instrucciones para completar el instrumento de revisión par:

1. Leer la propuesta de implementación adjunta.
2. Completar la información solicitada en el cuadro de dictamen general.
3. Completar el cuadro enunciados a considerar y clasificarlos utilizando la escala de Likert donde:
 1. Muy en desacuerdo.
 2. En desacuerdo.
 3. Ni de acuerdo ni en desacuerdo.
 4. De acuerdo.
 5. Muy de acuerdo.
4. Si así lo desea puede hacer comentarios dentro del documento, pero esto no es requisito.

Este instrumento está dirigido a los evaluadores de la propuesta creada para implementar un departamento de calidad en una empresa de software. Dicha propuesta fue enviada el mismo día que este instrumento a los evaluadores.

Se debe evidenciar si existen debilidades a subsanar con el fin de que se mejore la calidad de la propuesta presentada y dictaminar bajo el criterio de experto si dicha propuesta es útil, viable y realista.

DICTAMEN GENERAL LUEGO DE LA REVISIÓN (MARQUE UNA X):

1. Se recomienda aprobar la propuesta sin correcciones.
2. Se recomienda aprobar la propuesta con correcciones.
3. Se recomienda no aprobar la propuesta.

Califique según corresponda, los diferentes atributos de la propuesta revisada. Indique con un número entre 1 y 5, siendo 1 la nota más baja y 5 la más alta de la escala

Enunciados a considerar	Valor (de 1 a 5)
La propuesta realmente establece un plan para crear un departamento estándar de SQA.	
La propuesta puede usarse para controlar y asegurar el proceso de la gestión de calidad.	
La propuesta establece los procesos de SQA.	
La propuesta define los roles, responsabilidades y las prácticas de la función de SQA.	
La propuesta permite medir la efectividad de los procesos y sugerir mejoras.	
La propuesta permite garantizar la conformidad y aceptabilidad del producto por parte del usuario final.	
Usted implementaría dicha propuesta en su lugar de trabajo	

Si lo desea escriba aquí las observaciones o comentarios que tenga al respecto de la propuesta.

Nombre completo del revisor: _____

Muchas gracias

Apéndice 3: Propuesta de implementación

**Propuesta de implementación
del departamento de Software Quality Assurance
(SQA)
en una empresa de software**

Plantilla - Versión 1.1

Octubre - 2023

1. Historial de revisión

Versión	Autor	Descripción	Fecha
1.0	Pamela Green	Versión inicial	2 de agosto 2023
1.1	Pamela Green	Versión final	5 de octubre 2023

Tabla de contenido

1. Historial de revisión.....	2
2. Introducción	4
3. Definiciones y acrónimos	5
4. Marco organizacional.....	6
4.1. Objetivos.....	7
4.2. Gestión	8
4.3. Documentación.....	11
4.4. Estándares, normas, guías y comités	13
4.5. Métricas	14
4.6. Pruebas	16
4.7. Herramientas	19
5. Costos y beneficios.....	20
6. Conclusión	23
7. Bibliografía.....	24

2. Introducción

El propósito de este documento es servir como guía para implementar un departamento de Aseguramiento de la Calidad del Software, conocido en inglés como Software Quality Assurance (SQA), en una empresa de desarrollo de software. Esta guía es el resultado de una amplia revisión bibliográfica, varias observaciones, experiencias profesionales, recopilación y análisis de las respuestas a las entrevistas previamente realizadas y conocidos estándares, normas y comités de calidad como el del Instituto de ingenieros eléctricos y electrónicos (IEEE), Organización Internacional de Normalización (ISO), la Junta Internacional de Calificaciones de Pruebas de Software (ISTQB) y la Guía del Cuerpo de Conocimientos de Ingeniería de Software (SWEBOK) entre otros.

El departamento de SQA permitirá proporcionar una estructura organizada para gestionar, definir, medir y mejorar todos los aspectos relacionados con la calidad del producto (software) a lo largo de su ciclo de vida.

3. Definiciones y acrónimos

- Aseguramiento de la calidad (QA): Actividades que se realizan con el objetivo de asegurar que los productos y procesos posean el nivel de calidad requerido (Quiroa, 2022).
- Aseguramiento de la calidad del software (SQA): Proceso repetible y bien definido que se integra con la gestión de proyectos y los ciclos de vida de desarrollo de software para revisar los mecanismos de control interno, asegurar el cumplimiento de los estándares y procedimientos de software (Owens & Khazanchi, 2009).
- Calidad: Grado en que un producto o proceso cumple con los requisitos establecidos y con la satisfacción del cliente (Lewis, 2017).
- Control de calidad (QC): Proceso mediante el cual se busca garantizar que se mantenga o mejore la calidad del producto o proceso (Hayes, 2023).
- Caso de prueba (TC): Conjunto de instrucciones específicamente desarrolladas para determinar si se satisfacen los requisitos del software y este funciona correctamente (ISO 29119-3, 2013).
- Defecto: Error o falla en el software que hace que se produzca un resultado incorrecto, inesperado o que se comporte de manera no deseada (ISTQB, 2023).
- Pruebas de software: O “testing” de software en inglés, conjunto de actividades para descubrir defectos y evaluar la calidad del producto de software (ISTQB, 2023).
- Requerimiento: Capacidad que debe cumplir o poseer un producto o proceso para satisfacer un contrato, estándar, especificación u otro documento impuesto formalmente (Guía PMBOK 2017).
- Validación: Proceso de evaluación para comprobar si el software cumple con las necesidades y expectativas de los usuarios (ISTQB, 2023).
- Verificación: Proceso de evaluación para comprobar si el software se ha creado de acuerdo con los requisitos especificados previamente (ISTQB, 2023).

4. Marco organizacional

Según el IEEE Std 730-2014, la propuesta de implementación del departamento de Software Quality Assurance (SQA) en una empresa de software se debe de organizar y seguir de la siguiente manera:

- 4.1. Objetivos: Describe los objetivos en relación con la calidad tanto del proceso como del producto (software).
- 4.2. Gestión: Describe las principales actividades y elementos involucrados en el proceso y cómo se relacionan estos con el perfil, las tareas y responsabilidades de los miembros del departamento en relación con el aseguramiento y control de calidad.
- 4.3. Documentación: Describe la documentación producida para respaldar las actividades del aseguramiento y control de calidad del software.
- 4.4. Estándares, normas, guías y comités: Identifica los principales estándares, normas, guías y comités que permiten garantizar la calidad de software.
- 4.5. Métricas: Describe las principales medidas cuantitativas o cualitativas que se utilizan para evaluar diferentes aspectos relacionados con la calidad de software.
- 4.6. Prueba: Describe las principales pruebas a realizar para evaluar y verificar la calidad del software y garantizar que los productos cumplan con los estándares de calidad solicitados.
- 4.7. Herramientas: Describe la importancia, y tipos de herramientas que ayudan tanto al equipo de desarrollo como al de pruebas en las actividades relacionadas con el aseguramiento y control de calidad del software.

4.1. Objetivos

En este apartado se definen los objetivos de SQA. La importancia de estos radica en que le permitirá al departamento alcanzar las metas que se han propuesto previamente. Dichos objetivos son declaraciones específicas y cuantificables que describen lo que se desea lograr en términos de calidad del software. Es primordial que estos estén alineados con los objetivos generales de la organización.

Los objetivos de SQA deben establecerse durante la etapa de planificación. De esta manera servirán como guía durante las actividades del equipo SQA. Según ISTQB (2018) y el IEEE (2014), los principales objetivos de las actividades de SQA son los siguientes:

- Garantizar un nivel aceptable de confianza de que el software se ajusta a los requisitos técnicos solicitados.
- Gestionar e incentivar actividades relacionadas con la mejora del desarrollo de software.
- Identificar, prevenir, encontrar y gestionar los riesgos y defectos que pueden afectar la calidad del software.
- Incrementar y promover la formación y desarrollo profesional del personal relacionado con el departamento de calidad.

4.2. Gestión

En este apartado se describen los diferentes roles y responsabilidades del equipo de SQA, esto permite tener una visión clara y realista de las funciones que puede desempeñar dicho departamento dentro de la organización.

Perfil general de los miembros del equipo:

- Ingeniero en sistemas de información o carrera a fin.
- Habilidades blandas, como la atención al detalle, actitud de aprendizaje continuo, capacidad de priorización y resolución de problemas.
- Certificación similar o igual a la de ISTQB.
- Conocimientos básicos en programación.
- Experiencia o conocimiento en diferentes tipos de pruebas software.
- Experiencia en diseño, ejecución y documentación de casos de prueba.

Para gestionar adecuadamente un equipo de SQA, se necesita tener roles y responsabilidades bien definidas para cada uno de los miembros del equipo. Los principales miembros del equipo de SQA son, el gerente de pruebas (dependiendo del tamaño de la organización), el jefe de pruebas y el ingeniero de pruebas. En la Figura 1, se muestra el organigrama del equipo.

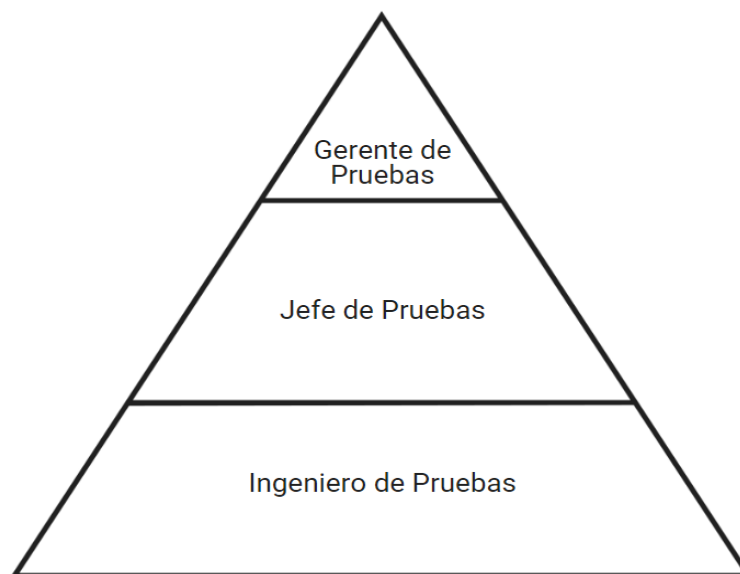


Figura 1. Organigrama del equipo de SQA. Elaborado a partir de Burnstein, 2006.

- Gerente de Pruebas: También conocido en inglés como QA Manager. Ocupa el nivel superior de la jerarquía, generalmente esta figura solo se encuentra en organizaciones bastante grandes que tienen varios equipos de SQA. Su principal objetivo es tener un panorama completo de los todos los proyectos que se manejan en el departamento de SQA, al tener dicho panorama, se convierte en un comunicador o enlace, entre la organización, la alta dirección, la gestión de proyectos, el equipo de desarrollo, el equipo de SQA y los interesados, identificando posibles cuellos de botella y gestionando las comunicaciones en todos los canales (Babaian, 2019). Según ISTQB, Expert Level (2011), algunas de las principales funciones y responsabilidades del gerente de pruebas son:
 - Responsable de definir las reglas y la cultura de todo el departamento de SQA, para garantizar que se logren los objetivos planteados.
 - Proporcionar orientación y dirección a la organización acerca de las actividades relacionadas con la calidad del software.
 - Desarrollar, implementar y establecer la estrategia y los procesos de SQA.
 - Dirigir el proceso de reclutamiento del personal de calidad de software.
 - Promover y defender al equipo de SQA.

- Jefe de pruebas: También conocido en inglés como QA Lead. Es el encargado de ayudar al gerente de pruebas a cumplir con los objetivos planteados y trabajar con el equipo de ingenieros de pruebas en proyectos individuales (Burnstein, 2006). Según Burnstein, (2006), algunas de las principales funciones y responsabilidades del jefe de pruebas son:
 - Responsable de supervisar el personal a su cargo.
 - Presentar distintos informes.
 - Ayudar a desarrollar las competencias de los ingenieros de pruebas.
 - Desarrollar, aplicar y mantener métricas para medir diferentes factores como el impacto de las pruebas, el avance del proyecto, etc.
 - Seleccionar, implementar e incentivar el uso de herramientas para dar soporte al proceso de calidad de software.
 - Decidir sobre la implementación de los entornos de prueba.
 - Asignar tareas a los ingenieros de pruebas de software.

- Encargado de mejorar la estrategia de prueba.
- Ingeniero de pruebas: También conocido en inglés como QA o Tester. Su objetivo principal es el de diseñar, desarrollar y ejecutar pruebas. Según el ISTQB (2018), algunas de las principales funciones y responsabilidades del ingeniero de pruebas son:
 - Aplicar las mejores prácticas, estrategias, estándares y procesos de QA en la organización.
 - Crear el plan de pruebas y los casos de prueba
 - Analizar, revisar y evaluar toda la documentación para las pruebas.
 - Diseñar y aplicar distintas pruebas para garantizar la estabilidad e integridad del software.
 - Evaluar el producto para garantizar la conformidad y aceptabilidad del producto por parte del usuario.
 - Identificar, documentar y dar seguimiento a los errores y riesgos encontrados.
 - Automatizar casos de pruebas según sea necesario.
 - Preparar y obtener datos de prueba.
 - Revisar pruebas desarrolladas por otros.
 - Crear y aplicar el cronograma de actividades relacionadas con la calidad de software.
 - Comunicar los resultados de las pruebas a las partes interesadas.
 - Identificar lecciones aprendidas que podrían mejorar los procesos.
 - Usar herramientas apropiadas para facilitar el proceso de prueba.

Es importante mencionar que los ingenieros de prueba pueden ser principiantes, intermedios y avanzados, esto dependerá de varios factores como, años de experiencia, estudios, dificultad de las funciones y responsabilidades del rol. Las funciones y responsabilidades no están limitadas solamente a las antes mencionadas.

4.3. Documentación

En este apartado se identifica y describe la documentación mínima que debe regir en el departamento SQA. La documentación del departamento de SQA es una colección de documentos creados y mantenidos por el equipo de SQA, con el fin de registrar todos los aspectos relevantes, estandarizar el proceso, aclarar terminologías y mantener informados a todos los miembros del equipo de SQA. Según la norma ISO 29119-3 (2013), la documentación del departamento de SQA se divide en proceso de prueba, gestión de pruebas y aplicación de pruebas y sus subdivisiones. En la Tabla 1, se muestran cada uno de los documentos.

La documentación debe ser revisada y actualizada periódicamente para asegurarse de que contiene información precisa, relevante y actualizada. Por lo que se recomienda utilizar un estricto control de versiones. La responsabilidad de la documentación es de todos los miembros del equipo.

Tabla 1. Descripción de la documentación. Elaborado a partir de la norma ISO 29119-3 (2013).

Tipo	Documento	Descripción
Proceso de prueba	Política de pruebas	Describe el propósito, los objetivos y el alcance general de las pruebas dentro de la organización.
	Estrategia de pruebas	Describe las pautas sobre cómo realizar las pruebas y lograr los objetivos establecidos a través de estas.
	Métricas de pruebas	Describe las diferentes métricas implementadas y utilizadas en el departamento de SQA para evaluar la calidad y eficacia en los procesos, productos y actividades.
Gestión de pruebas	Plan de pruebas	Describe las decisiones tomadas durante la planificación inicial, este documento incluye: descripción del proyecto, alcance, objetivos, cronograma, tareas, entregables, requerimientos, estrategia de pruebas, riesgos, entornos de pruebas y actividades, entre otros.
	Reporte de finalización de pruebas	Proporciona información sobre el estado de las pruebas que se realizan en un período específico.
Aplicación de pruebas	Casos de prueba	Describen paso a paso cómo realizar las pruebas, al comparar los resultados actuales con los resultados descritos, de esta manera se determina si todo funciona y satisface los estándares y requisitos. Se debe seguir un formato que contenga: título, objetivo, interfaz, precondiciones, ambiente de prueba, datos de entrada, pasos y resultados esperados (ISO 29119-5, 2016).
	Reporte de incidentes	Contiene toda la información sobre los defectos encontrados en el sistema. Reichert (2020), sugiere cada defecto reportado debe seguir un formato que contenga: título, descripción, pasos para reproducirlo, resultado actual, resultado esperado, prioridad, severidad y evidencia.

4.4. Estándares, normas, guías y comités

Este apartado detalla los principales estándares, normas, guías y comités que se aplican para garantizar un departamento de SQA.

Los estándares, normas, guías y comités para el aseguramiento de la calidad del software ayudan a las organizaciones a establecer pautas y criterios de éxito. Estos permiten garantizar que, sin importar el tipo de organización, esta cumpla con las especificaciones y los requerimientos de software solicitados (Carty, 2022).

En la Figura 2, se muestran algunos de los principales y más reconocidos estándares, normas, guías y comités que pueden ser utilizados como guía por la organización para establecer sus procesos de aseguramiento y control de la calidad del software.

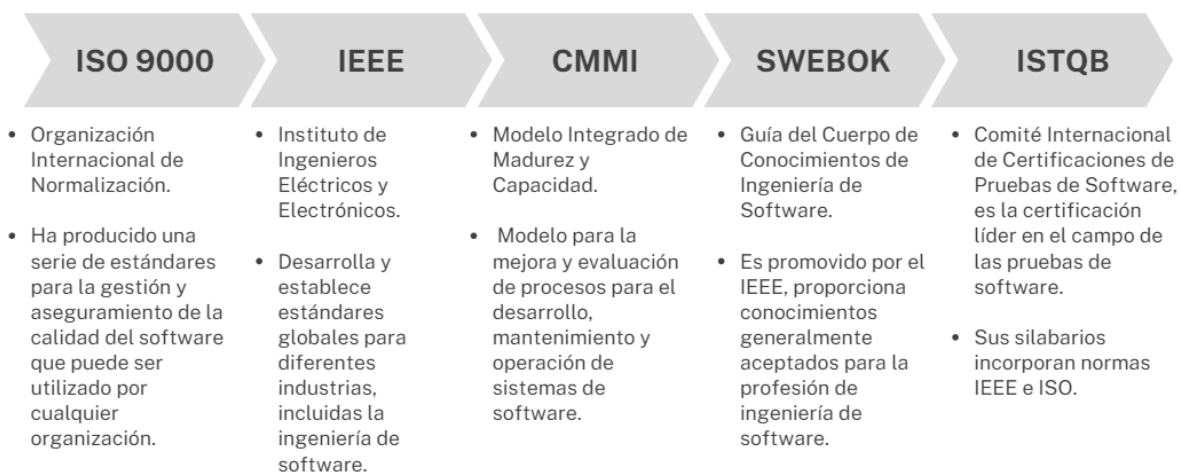


Figura 2. Estándares, normas, comités y guías. Elaborado a partir de Acosta et al. (2017).

Una de las tareas del departamento de SQA es asegurar que los estándares y normas que se hayan adoptado se sigan y que los productos del trabajo se apeguen a estos (Pressman, 2010).

4.5. Métricas

En este apartado se aborda la importancia que tienen las métricas para estimar el progreso, la calidad y la productividad, tanto del proceso como del producto y el proyecto. Ahmad (2022), comenta que las métricas en el departamento de SQA son un indicador que proporciona información valiosa y cuantitativa acerca de la eficacia de las actividades, del aseguramiento y control de calidad, de la eficiencia del equipo de trabajo, del resultado de las pruebas y del producto. Al realizar un seguimiento de estas métricas, las organizaciones pueden identificar áreas de mejora y tomar decisiones basadas en datos para garantizar que el departamento de SQA esté cumpliendo con su función y a la vez garantizar que el software cumple con los estándares de calidad deseados. Las métricas de prueba de software se dividen en tres grupos:

- Métricas de proceso: Evalúan la calidad y el desempeño del proceso de prueba. Este tipo de métrica ayuda a garantizar que el proceso de prueba esté alineado con los estándares, normas, objetivos, requerimientos y necesidades del cliente. También permite determinar el alcance, la cobertura, la integridad y la precisión de las pruebas. Algunas de las métricas de proceso más comunes son:
 - Cantidad de casos de prueba creados.
 - Cantidad de casos de prueba ejecutados.
 - Cantidad de casos de prueba fallidos.
 - Casos de prueba automatizados.
 - Cobertura de las pruebas.

- Métricas del producto: Walters (2022), menciona que este tipo de métricas miden la eficacia del producto al realizar un seguimiento y análisis sobre la cantidad, tipo, severidad, prioridad y estado de los defectos encontrados. Este tipo de métrica ayuda a identificar las causas fundamentales de los defectos. Algunas de las métricas de producto más comunes son:
 - Cantidad de defectos.
 - Gravedad y prioridad de los defectos.
 - Tiempo de resolución de defectos.
 - Miembro del equipo que ha identificado con éxito más defectos.

- Métricas del proyecto: Das (2023), indica que estas evalúan la calidad genérica del proyecto como tal. Este tipo de métrica ayuda a planificar, ejecutar y optimizar las actividades, recursos y métodos de prueba de una mejor manera. También permite determinar el costo, el tiempo y esfuerzo involucrado. Algunas de las métricas de proyecto más comunes son:
 - Progreso del proyecto.
 - Efectividad de las pruebas.
 - El esfuerzo de las pruebas.
 - Métricas de costos.

Algunas de las recomendaciones de Walters, (2022) para elegir las métricas de control de calidad para el departamento de SQA son:

- Las métricas deben de alinearse con la estrategia de pruebas del departamento de SQA.
- Las métricas deben ser relevantes para los objetivos del departamento de SQA y la organización.
- Las métricas deben ser medibles y cuantificables.

Los criterios de selección para los estándares, las normas y las métricas pueden variar según el tipo de negocio e industria en que la organización se desenvuelva. El departamento de SQA, puede guiarse por un único estándar, por más de un estándar o la combinación de varios de estos.

4.6. Pruebas

En este apartado se aborda el proceso de probar el producto de software. Las pruebas de software son un conjunto de actividades que permiten descubrir defectos, evaluar la calidad y funcionalidad del software. Estas pueden ser dinámicas o estáticas. Las pruebas dinámicas implican la ejecución del software, mientras que las pruebas estáticas no (ISTQB, 2023).

Parte fundamental del trabajo del departamento de SQA es garantizar que las pruebas se planeen y realicen de forma apropiada, de modo que pueda llevar a cabo su objetivo principal.

- Proceso de prueba

En ISTQB (2018), se menciona que no existe un proceso de prueba de software universal, pero existe un conjunto de actividades en común que permiten que las pruebas alcancen los objetivos establecidos. Este conjunto de actividades de prueba son un proceso de prueba. En la Figura 3, se describen las principales actividades involucradas en un proceso de prueba.

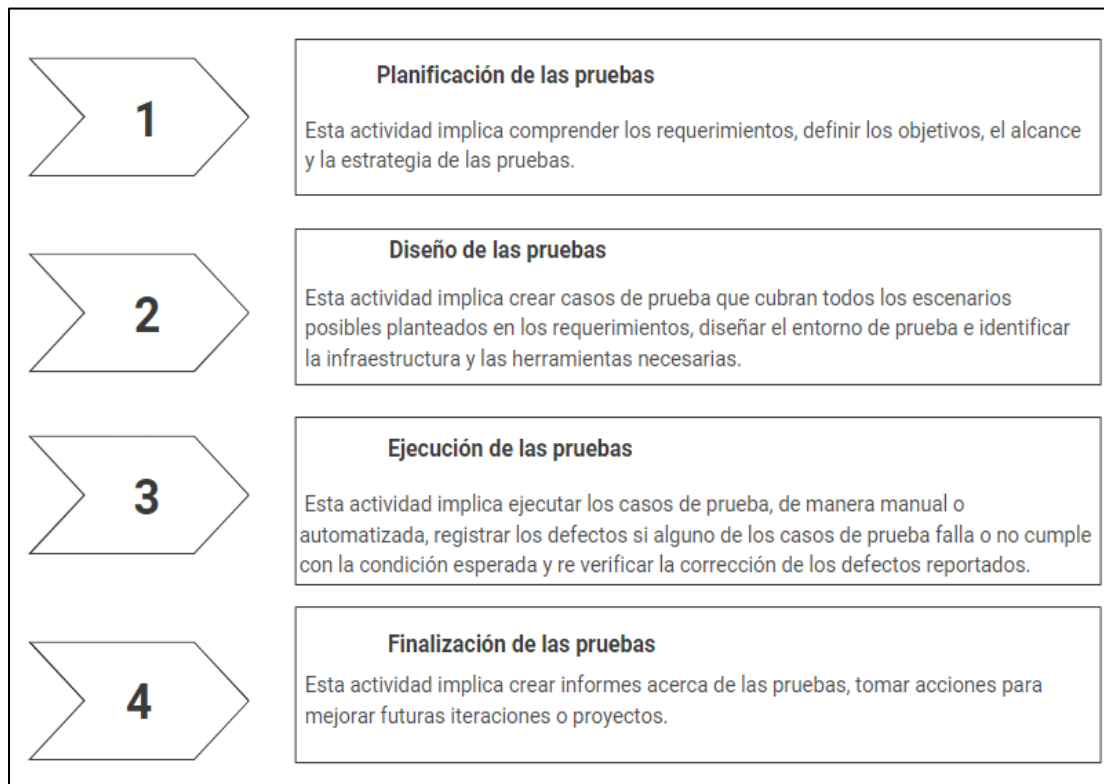


Figura 3. Proceso de prueba. Elaborado a partir de ISTQB (2018).

Anteriormente, se tenía la idea de que el proceso de las pruebas de software se debía realizar en la última etapa para consolidar la calidad de este; sin embargo, esta tendencia ha sido dejada de lado, se ha comprado a través de los años que entre más temprano inicien las pruebas mayores beneficios se obtienen, ya que estas se alinean al proceso de desarrollo (Paz, 2016).

- Tipos de prueba

Un tipo de prueba es un grupo de actividades destinadas a probar características específicas del producto de software (ISTQB, 2018). Existen varios tipos de pruebas de software, las pruebas automatizadas y las pruebas manuales, estas últimas se dividen en pruebas de caja blanca, pruebas de caja negra y pruebas de caja gris. Durante el proceso de prueba, los ingenieros de pruebas pueden utilizar varios tipos de técnicas pruebas, como pruebas funcionales, pruebas de rendimiento, pruebas de seguridad y pruebas de usabilidad (Doshi, 2023). En la Figura 4, se muestra la clasificación de los diferentes tipos de pruebas.

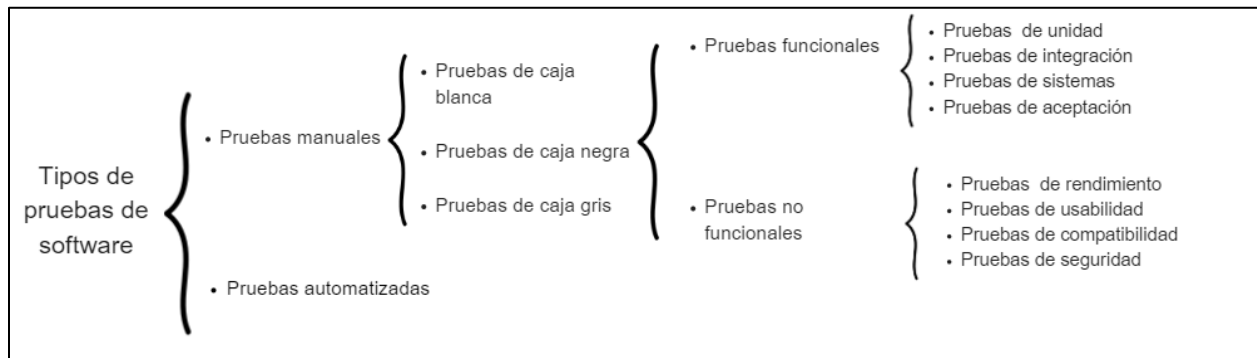


Figura 4. Tipos de pruebas. Elaborado a partir de Doshi, 2023.

Los tipos de pruebas utilizadas varían según el objetivo principal, el tipo de organización y las regulaciones específicas de la industria. En la Tabla 2, se muestra la descripción de cada una de las principales pruebas.

Tabla 2. Descripción de las pruebas. Elaborado a partir de Doshi, 2023 & ISTQB, 2023.

Tipo de Prueba	Descripción
Manuales	Implica que un evaluador humano realice el proceso de prueba manualmente, ejecutando los casos de prueba sin la ayuda de ninguna herramienta de automatización.
Automatizadas	Implican el uso de scripts y herramientas de automatización para ejecutar casos de prueba. Son especialmente efectivas cuando se tratan de pruebas repetitivas.
Caja blanca	Prueban el funcionamiento interno del sistema desde la perspectiva del desarrollador, centrándose en probar la lógica y la eficiencia del sistema.
Caja negra	Se centra en probar la funcionalidad de un sistema de software sin ningún conocimiento de su funcionamiento interno.
Caja gris	Combina los enfoques de las pruebas de caja blanca y negra.
Funcionales	Permiten garantizar que el sistema funcione según lo previsto y cumpla con los requisitos funcionales especificados por las partes interesadas.
No funcionales	Evalúan las características de los sistemas. Este tipo de prueba responde a la pregunta ¿qué tan bien se comporta el Sistema?
Unidad	Las unidades/componentes individuales se prueban de forma aislada del resto del sistema para garantizar que funcionen según lo previsto.
Integración	Se centra en probar las interacciones entre componentes o módulos individuales
Sistema	Prueba todo el sistema en su conjunto. Normalmente este tipo de prueba se realiza después de las pruebas de integración.
Aceptación	Verifican si el sistema cumple con los criterios de aceptación especificados. Por lo general, lo realizan los usuarios finales o las partes interesadas para garantizar que el software cumpla con sus requisitos y es adecuado.
Rendimiento	Evalúan cómo se desempeña un sistema bajo una carga de trabajo particular.
Usabilidad	Permiten comprobar si el sistema es fácil, adecuado y conveniente de usar.
Compatibilidad	Evalúan la compatibilidad del sistema con diferentes hardware, software, sistemas operativos, navegadores y otros dispositivos o componentes.
Seguridad	Permiten garantizar que la aplicación no tenga vulnerabilidades que puedan conducir a la pérdida de datos o amenazas.

4.7. Herramientas

Este apartado brinda las consideraciones sobre las diferentes herramientas de prueba. Según ISTQB (2023), las herramientas de prueba apoyan y facilitan muchas actividades de prueba. Por tanto, la herramienta de prueba adecuada aumenta la eficacia y eficiencia del proceso de prueba. Las herramientas de prueba se clasifican de la siguiente manera:

- Herramientas de gestión: Aumentan la eficiencia del proceso de prueba al facilitar la gestión de los requisitos, pruebas, defectos y demás datos importantes para el proyecto.
- Herramientas de diseño de pruebas: Facilitan la creación de casos de prueba y datos de prueba.
- Herramientas de ejecución y cobertura de pruebas: Permiten la ejecución automatizada de pruebas y la medición de la cobertura.
- Herramientas de prueba no funcionales: Permiten al ingeniero de pruebas realizar pruebas no funcionales que son difíciles de realizar manualmente.

Según Raulamo-Jurvanen et al. (2019), en la práctica de la ingeniería de software, evaluar y seleccionar las herramientas de prueba de software que mejor se adapten al departamento es una tarea importante y desafiante. ISTQB (2018), menciona que las herramientas de prueba son una inversión a largo plazo, que probablemente sean aplicables a muchos proyectos. Por ende, es importante tener presente las siguientes recomendaciones al momento de seleccionar una herramienta de pruebas de software:

- Conocer la madurez de la organización.
- Identificar y conocer los requisitos que se espera sean proporcionados por la herramienta.
- Evaluar y analizar la herramienta (costo/beneficio).
- Evaluar al proveedor, ya que se requiere que proporcione aspectos relacionados con el soporte técnico y la capacitación a los usuarios.
- Facilidad de uso, se debe considerar la curva de aprendizaje para garantizar que los usuarios puedan aprender a usar la herramienta rápidamente.
- Considerar diferentes modelos de licencia.

La elección de la herramienta de pruebas dependerá de las necesidades específicas de la organización y del proyecto que se esté llevando a cabo.

5. Costos y beneficios

Este apartado muestra el costo de las pruebas y cómo medirlo. Las pruebas de software toman un porcentaje del cronograma de desarrollo y a la vez estas pruebas son un porcentaje del costo del proyecto. Por lo tanto, el costo dependerá de la tarifa por hora y del número de ingenieros de pruebas involucrados en el proyecto (Shah, 2023). Los salarios de los ingenieros de pruebas pueden variar según factores como la ubicación geográfica, la experiencia, la especialidad y la industria. En la Tabla 3, se detallan los costos por hora para estos profesionales en Costa Rica.

Tabla 3. Costos por hora. Elaborado a partir de ERI Economic Research Institute, 2023.

Rol	Salario por hora
Ingeniero de pruebas	\$25
Líder de QA	\$30
Gerente de QA	\$34

Se debe tomar en cuenta que el personal necesario para probar un software depende de muchos factores como el tipo de pruebas, la complejidad, el tamaño y tipo de software, el plazo de entrega del mismo. Por ejemplo, si el software se desarrolla desde cero, es posible que necesite un ingeniero de pruebas por cada 5 o 6 desarrolladores, esto partiendo del principio de que los desarrolladores realicen pruebas unitarias y de integración. Por otro lado, si se está personalizando un software existente, el número de ingenieros de pruebas puede variar según la complejidad de este. Generalmente, el equipo ideal para probar un software pequeño es entre 2 y 5 ingenieros de prueba y entre 5 y 10 para un software de mediana escala (Shah, 2023).

Además, del costo del personal también se debe contemplar el costo de las herramientas de prueba de software. En la Tabla 4, se muestran algunos de los precios de estas herramientas.

Tabla 4. Costos de herramientas. Elaborado a partir de los sitios web de Function, Practicatest y de Click Up.

Herramienta	Precio (por usuario)	Categoría
Jira	\$8.17	Gestión
Test Rail	\$34	
Test Collab	\$29	
Soap UI	\$895	Ejecución
Test Complete	\$1626	
BrowserStack	\$30	Funcionales
dotCover	\$469	Cobertura

Estas son solo algunas de las muchas herramientas de pago disponibles para pruebas de software en el mercado. También existen las herramientas de código abierto las cuales los costos dependen de la implementación de cada empresa.

Por otro lado, Warcholinski (2023), menciona que el proceso de aseguramiento y control de calidad del software es realmente necesario en el desarrollo de software. Una buena gestión del departamento de SQA permite establecer objetivos claros y realistas, prevenir y detectar defectos en el software. Sin embargo, la calidad se obtiene a un precio. Ese precio es el costo de los esfuerzos de calidad, tiempo, recursos, equipo y herramientas que se necesitan para llevar a cabo este proceso. Ruiz & Ramos (2001), explican que el término Costo de Calidad o “Cost of Quality” (CoQ) en inglés, es una técnica útil que permite comprender las compensaciones económicas involucradas en la entrega de software de buena calidad. CoQ se basa en medir dos tipos principales de costos, como se muestra en la Figura 5.

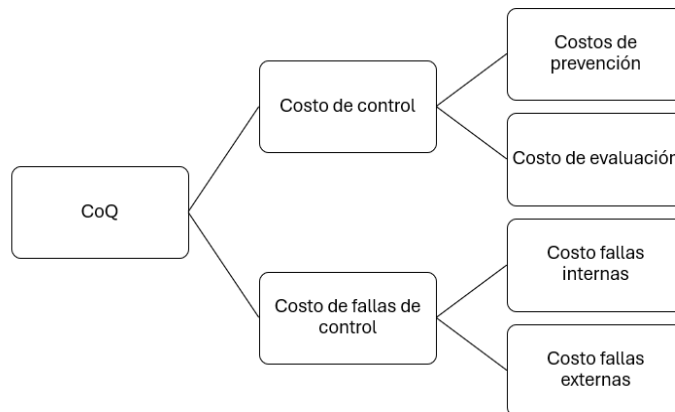


Figura 5. Tipos de costos asociados con la calidad de software. Adaptado de Warcholinski, 2023

En la Tabla 5, se muestra la definición de los tipos de costos asociados con la calidad de software.

Tabla 5. Definición de los tipos de costos asociados con la calidad de software. Elaborado a partir de Warcholinski, 2023.

Tipos de Costos	Descripción	Incluye
Prevención	Son todos los costos derivados del esfuerzo para asegurar la calidad del software y prevenir defectos en todas las fases del desarrollo de software.	Aseguramiento de la calidad (planeación de la calidad, mejora y definición de procesos, políticas y estándares), análisis de requerimientos, administración del proyecto (planeación, capacitación, diseño y métricas), entre otros.
Evaluación	Son todos aquellos costos derivados de probar el software para evitar que los defectos lleguen a producción.	Diseño de los casos de prueba, ejecución de diferentes tipos pruebas como las pruebas de seguridad, pruebas de rendimiento, etc y demás actividades asociadas.
Fallas internas	Son los costos del esfuerzo para detectar y corregir problemas o defectos antes de que el usuario final los detecte.	Esto incluye el retrabajo (los costos de desarrollo requerido para corregir los defectos y el costo de volver a probar).
Fallas externas	Son los costos de los problemas o defectos detectados por el usuario después de que el software ha sido entregado.	Este tipo de costos puede suponer un gasto importante para la organización, ya que incluye el daño producido al usuario, la pérdida de confianza, la mala reputación a la organización y las posibles sanciones.

Se puede medir el costo de las pruebas de software considerando los cuatro tipos de costos mostrados en la Tabla 5, utilizando la siguiente fórmula:

$$CoQ = \text{costo de control} + \text{costo de fallas de control}.$$

6. Conclusión

El proceso de implementación de un departamento de Aseguramiento de la Calidad del Software, conocido en inglés como Software Quality Assurance (SQA), desde cero será similar, independientemente del tamaño, la experiencia y el tipo de negocio o industria de la organización. Como se mostró previamente lo primero es establecer los objetivos que el departamento de pruebas ayudará a alcanzar, seguidamente se debe determinar la cantidad de personal necesario, seleccionarlo y contratarlo para iniciar la gestión en sí del departamento. Esta gestión debe abordar aspectos relacionados con la documentación, los estándares, normas y métricas que aplican para garantizar el éxito de un departamento de SQA, sin dejar de lado temas como las pruebas, el proceso de prueba, los tipos de pruebas y las herramientas. Implementar un departamento de SQA puede resultar un desafío, especialmente para aquellas organizaciones que no cuentan con una guía de implantación. Por ende, esta guía tiene como fin proporcionar el marco necesario para garantizar un enfoque coherente al momento de establecer un departamento de SQA dentro de la organización.

7. Bibliografía

Acosta, N. J., Espinel, L. A., & García, J. L. (2017). Estándares para la calidad de software. TIA Tecnología, investigación y academia, 5(1), 75-92.

Ahmad, N. (2022). QA Metrics Tutorial: A Comprehensive Guide With Examples And Best Practices. Lambdatest. Recuperado 28 /09/ 23, de <https://www.lambdatest.com/learning-hub/qa-metrics>

Babaian, A. (2019). The Role of a QA Manager in Scrum. Software Quality Professional, 21(4), 49-51.

Burnstein, I. (2006). Practical software testing: a process-oriented approach. Springer Science & Business Media.

Das, A. (2023). Metrics for Testing and Quality assurance: A detailed guide. Testsigma Blog. Recuperado 28 de septiembre de 2023, de <https://testsigma.com/blog/metrics-for-testing/>

Doshi, K. (2023). Different types of testing in software | BrowserStack. <https://www.browserstack.com/guide/types-of-testing>

ERI Economic Research Institute. (2023). Salary Expert - QA Engineer Salary Costa Rica. Salary Expert. Recuperado 7 de octubre de 2023, de <https://www.salaryexpert.com/salary/job/qa-engineer/costa-rica>

Carty, D. (2022). Software quality assurance (SQA) explained. Applause. <https://www.applause.com/blog/software-quality-assurance-sqa-explained>

Chu, S. (2023). How measuring cost of quality prevents software development from failing - ICTS Custom Software — ICTS Custom software. Recuperado 6 de octubre de 2023, de <https://icts.io/2022/11/24/how-measuring-cost-of-quality-prevents-software-development-from-failing/>

Hayes, A. (2023). Quality control: what it is, how it works, and QC careers. Investopedia. <https://www.investopedia.com/terms/q/quality-control.asp>

IEEE Std 730 (2014): Software Quality Assurance, The IEEE Computer Society, IEEE, Nueva York.

ISO 29119-3 (2013): Software and systems engineering- Software testing - Part 3: Test Documentation.

ISO 29119-5 (2016): Software and systems engineering- Software testing - Part 5: Keyword-Driven Testing.

ISTQB (2023). Foundation level syllabus version 4.0 (CTFL). International Software Testing Qualifications Board.

ISTQB, Expert Level. (2011). Test Management (Managing Testing, Testers, and Test Stakeholders) version 2011. International Software Testing Qualifications Board.

Lewis, W. E. (2017). Software testing and continuous quality improvement. CRC press.

Leloudas, P. (2023). Test Execution. In: Introduction to Software Testing. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-9514-4_6

Owens, D. M., & Khazanchi, D. (2009). Software quality assurance. In Handbook of research on technology project management, planning, and operations (pp. 242-260). IGI Global.

Paz, J. A. M. (2016). Análisis del proceso de pruebas de calidad de software. Ingeniería solidaria, 12(20), 163-176.

Pressman, R. S (7th Ed.) (2010). Ingeniería del Software. “Un Enfoque Práctico”, México: McGraw-Hill.

Project Management Institute. (2017). A guide to the Project Management Body of Knowledge (PMBOK guide) (6th ed.). Project Management Institute.

Quiroa, M. (2022). Aseguramiento de la calidad. Economipedia. <https://economipedia.com/definiciones/aseguramiento-de-la-calidad.html>

Raulamo-Jurvanen, P., Hosio, S., & Mäntylä, M. V. (2019, April). Practitioner evaluations on software testing tools. In Proceedings of the 23rd International Conference on Evaluation and Assessment in Software Engineering (pp. 57-66).

Reichert, A. (2020). How to write effective software defect reports. TechBeacon. <https://techbeacon.com/app-dev-testing/how-write-effective-software-defect-reports>

Ruiz Carreira, M., & Ramos Román, I. (2001). Estimación del Coste de la Calidad del Software a través de la Simulación del Proceso de Desarrollo. *Revista Colombiana de Computación-RCC*, 2 (1), 75-87.

Shah, H. (2023). How much does software testing cost? 9 Proven ways to optimize it. Simform - Product Engineering Company. <https://www.simform.com/blog/software-testing-cost/>

Walters, A. (2022). QA metrics to improve your software testing - TestRigor. testRigor AI-Based Automated Testing Tool. <https://testrigor.com/blog/essential-qa-metrics-to-improve-your-software-testing/>

Warcholinski, M. (2023). How to estimate the cost of quality in software development in 2023. Recuperado 5 de octubre de 2023, de <https://brainhub.eu/library/cost-of-quality-in-software-development>

Whittaker, J. A. (2000). What is software testing? And why is it so hard?. IEEE software, 17(1), 70-79.

Apéndice 4: Revisión de José Chacón

Instrucciones para completar el instrumento de revisión par:

1. Leer el documento titulado Propuesta de implementación VF.
2. Completar la información solicitada en el cuadro de dictamen general.
3. Completar el cuadro enunciados a considerar y clasifíquelos utilizando la escala de Likert donde:
 1. Muy en desacuerdo.
 2. En desacuerdo.
 3. Ni de acuerdo ni en desacuerdo.
 4. De acuerdo.
 5. Muy de acuerdo.
4. Si así lo desea puede hacer comentarios dentro del documento, pero esto no es requisito.

Instrumento de revisión par

Este instrumento está dirigido a los evaluadores de la propuesta creada para implementar un departamento de calidad en una empresa de software. Dicha propuesta fue enviada el mismo día que este instrumento a los evaluadores.

Se debe evidenciar si existen debilidades a subsanar con el fin de que se mejore la calidad de la propuesta presentada y dictaminar bajo el criterio de experto si dicha propuesta es útil, viable y realista.

DICTAMEN GENERAL LUEGO DE LA REVISIÓN (MARQUE UNA X):

1. **Se recomienda aprobar la propuesta sin correcciones.**
2. **Se recomienda aprobar la propuesta con correcciones.**
3. **Se recomienda no aprobar la propuesta.**

Califique según corresponda, los diferentes atributos de la propuesta revisada. Indique con un número entre 1 y 5, siendo 1 la nota más baja y 5 la más alta de la escala.

Enunciados a considerar	Valor (de 1 a 5)
La propuesta realmente establece un plan para crear un departamento estándar de SQA.	5
La propuesta puede usarse para controlar y asegurar el proceso de la gestión de calidad.	5
La propuesta establece los procesos de SQA.	5
La propuesta define los roles, responsabilidades y las prácticas de las funciones del departamento de SQA.	2
La propuesta permite medir la efectividad de los procesos y sugerir mejoras.	5
La propuesta permite garantizar la conformidad y aceptabilidad del producto por parte del usuario final.	5
Usted implementaría dicha propuesta en su lugar de trabajo.	5

Si lo desea escriba aquí las observaciones o comentarios que tenga al respecto de la propuesta.

- En la sección “4.2. Gestión”: ¿Qué pasa en empresas que no tienen definido un departamento de QA ni lo van a tener? Yo trabajo en una empresa donde no hay QA manager ni QA lead. El jefe del tester es el mismo jefe de desarrollo. En metodologías ágiles los QA son parte del equipo de desarrollo y no tienen una estructura de managers o líderes de QA
- En “herramientas”, sería bueno mencionar herramientas para automatizar.

Nombre completo del revisor: José Miguel Chacón Araya

Apéndice 5: Revisión de Mario Cordero

Instrucciones para completar el instrumento de revisión par:

1. Leer el documento titulado Propuesta de implementación_VF.
2. Completar la información solicitada en el cuadro de dictamen general.
3. Completar el cuadro enunciados a considerar y clasifíquelos utilizando la escala de Likert donde:
 1. Muy en desacuerdo.
 2. En desacuerdo.
 3. Ni de acuerdo ni en desacuerdo.
 4. De acuerdo.
 5. Muy de acuerdo.
4. Si así lo desea puede hacer comentarios dentro del documento, pero esto no es requisito.

Instrumento de revisión par

Este instrumento está dirigido a los evaluadores de la propuesta creada para implementar un departamento de calidad en una empresa de software. Dicha propuesta fue enviada el mismo día que este instrumento a los evaluadores.

Se debe evidenciar si existen debilidades a subsanar con el fin de que se mejore la calidad de la propuesta presentada y dictaminar bajo el criterio de experto si dicha propuesta es útil, viable y realista.

DICTAMEN GENERAL LUEGO DE LA REVISIÓN (MARQUE UNA X):

1. Se recomienda aprobar la propuesta sin correcciones.
2. Se recomienda aprobar la propuesta con correcciones.
3. Se recomienda no aprobar la propuesta.

Califique según corresponda, los diferentes atributos de la propuesta revisada. Indique con un número entre 1 y 5, siendo 1 la nota más baja y 5 la más alta de la escala.

Enunciados a considerar	Valor (de 1 a 5)
La propuesta realmente establece un plan para crear un departamento estándar de SQA.	5
La propuesta puede usarse para controlar y asegurar el proceso de la gestión de calidad.	5
La propuesta establece los procesos de SQA.	5
La propuesta define los roles, responsabilidades y las prácticas de las funciones del departamento de SQA.	5
La propuesta permite medir la efectividad de los procesos y sugerir mejoras.	5
La propuesta permite garantizar la conformidad y aceptabilidad del producto por parte del usuario final.	5
Usted implementaría dicha propuesta en su lugar de trabajo.	5

Si lo desea escriba aquí las observaciones o comentarios que tenga al respecto de la propuesta:

- Considero que se debe contemplar términos como establecer e implementar. Sentar las bases es establecer, pero la implementación ya requiere evidencia de la aplicación.
- La parte de gestión está bastante completa, valide si fuese bueno agregar más niveles relacionados con el rol de ingeniero de pruebas según las diferentes necesidades.
- Respecto a la documentación, se debe validar los tipos de documentación. Proceso + Gestión + Aplicación.
- Revisar si se puede asociar los entregables a cada una de las siguientes etapas:
 - Test Planning and Control.
 - Test Analysis and Design.
 - Test Implementation and Execution.

- Evaluating Exit Criteria and Reporting.
- Test Closure.

- Es importante agregar una etapa de cierre de pruebas, donde se finiquiten resultados finales. Un reporte de finalización de pruebas debe ir justamente en una etapa de cierre.

- En relación con las normas es importante especificar la norma ISO 9001:2015 y también ISO/IEC 25010.

- El resto de los elementos los veo bastante bien como base para lo que quieres definir.

Nombre completo del revisor: Mario Cordero

Anexos

Anexo 1: Respuestas de las entrevistas

José Miguel Chacón Araya	
Respuesta 1	Me gusta que los QA sean parte del equipo de desarrollo, así no se ven como un departamento aparte y fluye más la comunicación. Esto se utiliza mucho en el desarrollo de metodologías ágiles.
Respuesta 2	Estudios en informática afines a sistemas y mínimo la certificación ISTQB.
Respuesta 3	Dejaría un QA lead para armonizar y ordenar los QA en sus respectivos equipos de trabajo. Pero un departamento aparte no me gustaría solo si es software de desarrollo en cascada si considero que debe tener un departamento aparte con QA Manager, QA leads y testers.
Respuesta 4	Se debe evaluar con métricas de cuantos bugs se encuentran por ejemplo y evaluar por trabajos exitosos liberados. Es importante también tener una métrica de incidentes encontrados en producción.
Respuesta 5	Tendría opción de que los nuevos que entran puedan ir subiendo de puesto a (Mid, Senior y posteriormente Lead). Incentivaría el estudio de las certificaciones.
Respuesta 6	Hay que plantear si todas las empresas ocupan un departamento de QA o si los QA como parte del equipo de desarrollo es suficiente.

Julio Fallas Díaz

Respuesta 1	<p>El equipo de calidad debe de cubrir desde el planeamiento del proyecto hasta la entrega del mismo. Si el equipo es grande se podría delegar funciones a diferentes personas. Las funciones serían:</p> <ul style="list-style-type: none">● Definición de casos de prueba● Revisión de cada integración nueva de código● Ejecución de casos de prueba.● Si el equipo es pequeño entonces el planeamiento debe incluir riesgos con base a tiempo de pruebas
Respuesta 2	<p>Un QA de software debe de saber programar y conocer el código que se usa para el desarrollo del software y tener habilidades que le permitan poder usar y comprender el software. Con estos dos perfiles el tester puede hacer pruebas de caja negra, blanca y gris. De esta manera el nivel de calidad aumenta considerablemente</p>
Respuesta 3	<p>El departamento de SQA debe implementarse de la mano con los desarrolladores para garantizar la calidad incluso del código que se integra a la solución, siendo el SQA un revisor de integraciones de código.</p>
Respuesta 4	<p>El departamento de evalúa con base a la estabilidad de software, un error o pulga debe ser identificado por el departamento mucho antes de integrarse al código, si la diferencia de pulgas encontradas en producción contra la cantidad de pulgas encontradas en desarrollo muestra que en producción hay más pulgas identificadas entonces quiere decir que el proceso de calidad no es el adecuado.</p>
Respuesta 5	<p>Una mejora sería en capacitaciones para el equipo para mantener siempre el departamento actualizado con las tecnologías más recientes.</p>
Respuesta 6	<p>Entrenar al departamento de SQA para que apunte más aún perfil de un SDET (software development engineer in testing) donde una persona hace desarrollo y también implementación de pruebas automatizadas para el código que está modificado.</p>

Alejandro Navarro	
Respuesta 1	Estructura, líder y equipo sólido. Los cuales lleven a cabo pruebas, configuración de ambiente, reporte de problemas.
Respuesta 2	Responsables, proactivos, mucha comunicación oral y escrita. Que de buena retroalimentación y esté pendiente del Proyecto.
Respuesta 3	Este debe ser implementado, respaldado y aprobado por la organización. Si no se cuenta con este apoyo muy probablemente este departamento no sea tomado en cuenta realmente.
Respuesta 4	Cantidad de regresiones encontradas y pulgas reportadas. Si se encuentran pocas pulgas puede que algo no se esté haciendo bien, no existe el software perfecto.
Respuesta 5	Un equipo, cada uno involucrado en el proyecto, adicional que el mismo equipo se encargue de tareas de automatización para mantener la información del flujo y detectar regresiones.
Respuesta 6	N/A

Melvin Álvarez	
Respuesta 1	La estructura debería de ser, un departamento bien gestionado (gente altamente capacitada, capaz de dar entrenamientos, actualizado y que posea habilidades blandas). Que cuente con: Líderes de área o Test mánager (capaces de liderar, entrenar, cada uno especializado en su departamento manual y automatización.). Además, con áreas de calidad Manual, seguridad, automatización WEN ,APIS (estas llevadas a cabo por parte de los colaboradores).
Respuesta 2	Yo considero que el perfil debería de ser: <ul style="list-style-type: none"> ● Ingeniero en sistemas de información o carrera a fin ● Habilidades blandas y de comunicación ● Cursos de ISTQB ● Manejo de APIS con Postman ● SQL ● Programación básica ● Luego las especificaciones para la automatización y seguridad
Respuesta 3	Esto dependerá de cada empresa y de las necesidades que ésta requiera. Pero si es importante la definición de los procesos los cuales va con lo mencionado anteriormente, utilizar herramientas adecuadas, mantener al personal motivado y entrenado, mejora continua y muy importante QA involucrado desde las primeras etapas.
Respuesta 4	Mejora continua, velar que se estén actualizando. Definir estándares o métricas que se deben cumplir para cada rol.
Respuesta 5	Hay que: <ul style="list-style-type: none"> ● Identificar a los colaboradores “curiosos y motivados” darles el ambiente y el tiempo adecuados para sus propuestas. ● Escuchar propuestas para capacitar a los colaboradores. ● Rotación de equipos, proyectos cada 6 meses. (depende de la criticidad del proyecto).
Respuesta 6	Impulsar la creatividad de los colaboradores, reconocerla, darle el ambiente y confianza de expresar además de implementarlas.

Eduardo Peña

Respuesta 1

La estructura debe ser lo más holística posible y cubrir áreas tales como:

- Planeación de Pruebas de calidad: Se encarga de planificar, diseñar y ejecutar pruebas en el software para identificar defectos y asegurar su correcto funcionamiento. Esto debería incluir pruebas unitarias, pruebas de integración, pruebas de sistema, pruebas de rendimiento, pruebas de seguridad, entre otras.
- Gestión de Control de calidad: Esta área se ocupa de establecer y mantener los estándares de calidad para el desarrollo de software. Esto implica definir procesos, metodologías y prácticas de desarrollo de calidad, así como asegurarse de que se sigan correctamente.
- Aseguramiento de la calidad: Esta área se encarga de monitorear y evaluar los procesos de desarrollo de software para identificar áreas de mejora. También realizar auditorías y revisiones de calidad.
- Gestión de incidencias: Se dedica a gestionar el ciclo de vida de los defectos identificados durante las pruebas y el uso del software. Incluye la recolección de informes de defectos, la asignación y seguimiento de tareas de corrección, y la verificación de la resolución de los defectos.

Respuesta 2

Debe de tener:

- Conocimiento técnico: Conocimiento técnico en desarrollo de software, lenguajes de programación, arquitectura de software, bases de datos, frameworks y herramientas de prueba. Esto les permite comprender el funcionamiento del software y detectar posibles problemas.
- Habilidades de prueba: Deben tener experiencia y conocimiento en diferentes tipos de pruebas, como pruebas de unidad, pruebas de integración, pruebas de sistema, pruebas de rendimiento y pruebas de seguridad. También deben estar familiarizados con herramientas de automatización.
- Orientado a la calidad: Debe tener la capacidad de analizar detalles minuciosamente y tener una actitud proactiva para garantizar que el software cumpla con los estándares de calidad.
- Habilidades de comunicación: Comunicación efectiva para colaborar con los miembros del equipo de desarrollo y aquellos involucrados. Esto

	<p>implica la capacidad de expresar los problemas de calidad, proponer soluciones y brindar retroalimentación constructiva.</p> <ul style="list-style-type: none"> ● Mentalidad analítica: Deben tener habilidades analíticas para analizar datos y métricas de calidad, identificar tendencias y patrones, y tomar decisiones basadas en evidencia. ● Actitud de aprendizaje continuo: Es importante que los profesionales del área tengan una actitud de aprendizaje continuo. Deben estar dispuestos a mantenerse actualizados sobre las nuevas tendencias, tecnologías y prácticas relacionadas con la calidad de software.
Respuesta 3	<p>Considero que se debe implementar en base a varios supuestos partiendo de:</p> <ul style="list-style-type: none"> ● Definir los objetivos y alcance: Establecer los objetivos y alcance del departamento en relación con la organización y el desarrollo de software, determinar qué tipo de actividades de aseguramiento de calidad se desea llevar a cabo y cómo se alineará con los procesos existentes. ● Establecer la estructura organizativa: Define la estructura del departamento de QA, incluyendo roles y responsabilidades. Determinar si se necesitará un líder del departamento de QA, el equipo de calidad requerido o incluso si los miembros del equipo de desarrollo también desempeñarán funciones de calidad. ● Desarrollar políticas y estándares de calidad: Crea políticas y estándares de calidad. Se debe incluir la definición de procesos de desarrollo, criterios de aceptación, metodologías y mejores prácticas específicas que se seguirán en todo el ciclo de vida del software. ● Proporcionar capacitación y formación: Dar capacitación y formación adecuada a los miembros del departamento y a otros equipos involucrados. Esto les permitirá adquirir las habilidades y conocimientos necesarios para realizar las tareas de calidad de manera efectiva. ● Implementar herramientas y tecnologías: Identificar las herramientas y tecnologías necesarias para respaldar las actividades de aseguramiento de calidad. ● Fomentar la colaboración y comunicación: Promueve la colaboración y una comunicación abierta entre el departamento y otros equipos.
Respuesta 4	A través de:

	<ul style="list-style-type: none"> ● Cumplimiento de estándares y procesos: Evalúa el grado en que el departamento de SQA está siguiendo los estándares y procesos establecidos. Verifica si se están aplicando las políticas de calidad, si se están ejecutando las pruebas de acuerdo con los procedimientos establecidos y si se están cumpliendo los plazos y entregables. ● Eficiencia de las pruebas: Evalúa la eficiencia de las pruebas realizadas por el departamento de SQA. Analiza el tiempo y los recursos dedicados a las pruebas, la cobertura alcanzada, la detección temprana de defectos y la capacidad para identificar problemas críticos antes del lanzamiento. ● Retroalimentación de otros departamentos: Solicita retroalimentación de otros departamentos, como el equipo de desarrollo, los usuarios y los clientes. Pregunta sobre su satisfacción con la calidad del software, la colaboración con el departamento de QA y si sus expectativas se han cumplido. ● Evaluación del desempeño del equipo: Evalúa el desempeño individual y colectivo de los miembros del departamento de SQA. Considera aspectos como la productividad, la competencia técnica, la capacidad de resolución de problemas, la colaboración y la contribución al logro de los objetivos de calidad. ● Mejora continua: Fomentar la cultura de mejora continua dentro del departamento. Animar a los miembros del equipo a proponer ideas y soluciones para mejorar los procesos, la eficacia de las pruebas y la calidad del software. Realiza seguimiento de las acciones de mejora implementadas y evalúa su impacto en la calidad general.
Respuesta 5	<ul style="list-style-type: none"> ● Definir una visión clara: definir una visión clara y comunicándola a todos los miembros del equipo. Esto ayudará a alinear a todos hacia metas comunes y brindará un sentido de propósito. ● Desarrollo de un plan de carrera: Crea un plan de carrera para los profesionales del departamento de SQA, con oportunidades de crecimiento y desarrollo bien definidas. Esto puede incluir niveles de experiencia, roles especializados y programas de capacitación y certificación. ● Establecer programas de capacitación: Proporciona oportunidades regulares de capacitación y desarrollo profesional para los miembros del departamento de SQA. Esto puede incluir cursos internos, asistencia a

	<p>conferencias, talleres, seminarios web y programas de mentoría.</p> <ul style="list-style-type: none"> ● Fomentar la participación en comunidades profesionales del área: Anima a los profesionales del departamento de SQA a participar en comunidades profesionales relacionadas con la calidad de software. Esto puede ser a través de grupos de usuarios, conferencias, foros en línea y redes sociales. La interacción con otros profesionales del campo fomentará el intercambio de conocimientos y experiencias. ● Establecer programas de reconocimiento: Reconoce y valora el trabajo y los logros de los profesionales del departamento de SQA. Puedes implementar programas de reconocimiento, premios o menciones especiales para aquellos que han realizado contribuciones significativas a la calidad del software. ● Fomentar la colaboración y el aprendizaje en equipo: Crea un entorno de trabajo colaborativo en el que los profesionales del departamento de SQA puedan compartir conocimientos, experiencias y mejores prácticas entre ellos. ● Apoyar la participación en conferencias y eventos: Brinda la oportunidad y el apoyo para que los profesionales del departamento de SQA asistan a conferencias, eventos y seminarios relevantes. Estas experiencias les permitirán mantenerse actualizados con las últimas tendencias y establecer contactos en la industria.
Respuesta 6	<ul style="list-style-type: none"> ● Colaboración desde el inicio: Involucra al departamento de SQA desde las etapas iniciales del ciclo de vida del desarrollo de software. Esto permitirá la identificación temprana de problemas de calidad, la definición de requisitos y estándares de calidad, y la planificación adecuada de las actividades de aseguramiento de calidad. ● Comunicación efectiva: Establece canales de comunicación claros y efectivos entre el departamento de SQA, el equipo de desarrollo, los stakeholders y otros departamentos relevantes. ● Automatización de pruebas: Automatizar las pruebas tanto como sea posible esto ayuda a mejorar la eficiencia, la consistencia y la cobertura de las pruebas, permite una detección más rápida de problemas y reduciendo el tiempo y los recursos necesarios para las pruebas manuales. ● Integración continua: Implementa prácticas de integración continua para

	<p>asegurar que las nuevas funcionalidades y cambios en el software sean probados de manera regular e integrados de manera exitosa.</p> <ul style="list-style-type: none">● Mejora continua: Establece un proceso de mejora continua para el departamento de SQA.● Adaptabilidad y flexibilidad: Mantenerse abierto a adoptar nuevas metodologías, herramientas y prácticas que puedan mejorar la calidad del software y la eficiencia del departamento de SQA.● Cultura de calidad: Promueve una cultura de calidad en toda la organización. Fomenta la responsabilidad compartida por la calidad del software y la importancia de la prevención de defectos en lugar de la detección tardía.
--	--

Oscar Miranda

Respuesta 1	<p>Las áreas que debe cubrir un departamento de calidad son relativas, esto va muy de la mano del tipo de empresa, pero a grandes rasgos deberían de ser un área que se encargue de establecer el departamento, guiarlo, desarrollar los estándares y encargarse de ejecutarlos. Considero que es muy importante que el departamento de SQA esté presente desde el inicio del proyecto.</p>
Respuesta 2	<ul style="list-style-type: none">● Debe ser una persona que sea detallista.● Que sepa crear un defecto y un plan de pruebas.● Que tenga buenas habilidades blandas y de comunicación, ya que tendrá que decirles a los desarrolladores que hay bugs y esto hay que saberlo decir para que no se creen roces.● Sin duda debe tener una carrera a fin con la ingeniería de software.● Debe ser una persona que sea bastante técnica pero que también pueda tomar el papel de usuario básico cuando sea necesario.● Debe ser una persona que le guste interactuar y rebuscar bugs.● Que le guste estar en constante aprendizaje, ya que se requiere que se certifiquen.● Que pueda trabajar independientemente y bajo presión.● Que sepa priorizar y resolver problemas
Respuesta 3	<p>El departamento de SQA debe implementarse en todas las organizaciones de software sin duda, este debe ser respaldado por toda la organización y desde mi experiencia, lo mejor es que sea un departamento aparte del del equipo de desarrollo.</p> <p>Se debe crear una cultura dentro de la organización de que no existe una rivalidad entre el departamento de SQA y el de desarrollo, más bien todo lo contrario que ambos están para apoyarse.</p> <p>El departamento no puede crearse de la nada, este debe estar basado en estándares y fundamentos válidos para poder llevarlo a cabo.</p> <p>La unidad de SQA, debería estar conformada por un gerente de SQA y un staff</p>

	de ingenieros de calidad. Sin embargo, el número exacto de miembros está fuertemente ligado por el tamaño y las necesidades de la organización, así como también por los recursos disponibles
Respuesta 4	Las evaluaciones pueden ser de diversas, en este momento se me ocurre cantidad y prioridad de bugs reportados además de solicitar retroalimentación de las personas que trabajan con este departamento.
Respuesta 5	<p>Buscaría que este departamento esté alineado a los objetivos de calidad y entrega de la organización.</p> <p>Sería importante implementar un sistema de evaluación que refleje puntos de mejora.</p> <p>Constante capacitación por parte de los integrantes del equipo.</p>
Respuesta 6	Ninguna.

Luis Solano

Respuesta 1	QA Manager, área de control de calidad, aseguramiento de calidad y automatización de pruebas.
Respuesta 2	Ingenieros en sistemas, que tengan al menos 2 años de experiencia en programación, generación y documentación de casos de requerimientos, casos de prueba, ambientes de pruebas, automatización, etc.
Respuesta 3	El departamento de SQA, debería tener un QA Manager, QA Leads, QA analyst y QA Automation Engineer
Respuesta 4	Se debería de evaluar de acuerdo con sus asignaciones en proyectos, casos de pruebas creados, casos de pruebas validadas, pulgas creadas/resueltas/devueltas/cerradas, y casos de pruebas automatizados.
Respuesta 5	Deberían tomar cursos de actualización de calidad y diseño de pruebas, certificación de pruebas (ISTQB), herramientas de automatización de pruebas, recolección de requerimientos, etc.
Respuesta 6	N/A

Alexa Ramírez	
Respuesta 1	<ul style="list-style-type: none"> ● Gestión de calidad: Establecer políticas, procedimientos y supervisar su implementación y cumplimiento. ● Control de calidad: Responsable de efectuar las pruebas para asegurar que el producto cumple con los estándares de calidad establecidos. ● Aseguramiento de claridad: Planes de control, gestión de riesgos, auditorías. ● Mejora continua: identificar oportunidades de mejora.
Respuesta 2	<ul style="list-style-type: none"> ● Conocimiento técnico. ● Orientación a la calidad. ● Experiencia o conocimiento en pruebas de software. ● Atención al detalle. ● Habilidades analíticas y de resolución de problemas. ● Habilidades de trabajo en equipo.
Respuesta 3	<ul style="list-style-type: none"> ● Definir los objetivos y alcance. ● Definir estructura organizativa (roles y responsabilidades). ● Establecer procesos y estándares de calidad. ● Definir métricas de calidad y seguimiento. ● Implementación de herramientas. ● Capacitación y desarrollo continuo.
Respuesta 4	<ul style="list-style-type: none"> ● Métricas de calidad de software (estabilidad del software, satisfacción del cliente). ● Métricas de rendimiento. ● Retroalimentación. ● Análisis de mejora continua. ● Evaluación de procesos con auditorías.
Respuesta 5	<ul style="list-style-type: none"> ● Definir una visión y estrategia clara y alineada con los objetivos de la organización. ● Promover la cultura de la calidad. ● Proporcionar oportunidades de crecimiento. ● Establecer un ambiente de aprendizaje y desarrollo profesional. ● Reconocimiento y recompensa.

Respuesta 6	N/A
-------------	-----

Fabiana Salas	
Respuesta 1	La estructura va a depender mucho del tipo de empresa y producto que ofrecen al mercado. Pero las áreas que deberían cubrirse es el control de calidad (verificando que los productos cumplan con los estándares definidos), aseguramiento de calidad (Planeamiento de actividades que permitan asegurar que se va a cumplir todos los requisitos de calidad) y mejora continua (apoyar para poder identificar oportunidades de mejora tanto del equipo como de otras áreas).
Respuesta 2	Considero que deberían poseer un perfil técnico, ya que a la hora de trabajar con software es importante tener conocimiento en esta área con el fin de poder realizar un testeo más exhaustivo al producto. Inclusive para poder realizar pruebas de caja blanca y poder ver que el software sea desarrollado de acuerdo con los estándares de la empresa. Además, debe contar con habilidades de comunicación tanto escritas como verbales.
Respuesta 3	Primero se debe realizar un análisis del proceso o procesos a los cuales se les desea realizar un seguimiento de QA esto con el fin de establecer objetivos claros. Cuando esto se defina, se deberían especificar las responsabilidades/roles del departamento de SQA durante cada una de las etapas del proceso. Por último, se debe desarrollar los estándares y procesos de QA y capacitar al personal tanto del departamento como con los demás departamentos de la empresa esto con el objetivo que se tenga claro cuál es el rol e importancia del departamento y así poder colaborar y comunicarse de manera efectiva.
Respuesta 4	Se deben establecer métricas de desempeño para poder evaluar objetivamente al departamento. Evaluar la cobertura de las pruebas, la colaboración con otros equipos, la eficiencia y productividad del equipo. Sin embargo, considero que no se debería evaluar por ejemplo cuantas pulgas el tester reporto del producto, sino más bien cuantas pulgas validas reportó el

	cliente esto para medir la calidad de los entregables de software.
Respuesta 5	Como menciono en el punto 2.3 sería muy importante generar una colaboración y comunicación efectiva con el resto de los departamentos esto con el fin de darle visibilidad y además poder tener una mejora continua a través de experiencia y feedback. Además, sería importante que el jefe del departamento tenga una matriz de skills donde pueda saber los fuertes y debilidades del equipo, y así pueda impulsar a sus trabajadores a que se ayuden mutuamente a crecer, y así también pueda ver que áreas necesitan más capacitación y buscar ayuda en la gerencia para que sus trabajadores tengan acceso a certificaciones.
Respuesta 6	N/A

María Fernanda Rodríguez	
Respuesta 1	Un departamento de calidad debe contar con personal que sepa analizar y transcribir las necesidades de los clientes en productos de software, diseñar, implementar, ejecutar pruebas y dar seguimiento de las mismas tanto manuales como automáticas.
Respuesta 2	Una persona que trabaje en el área de calidad debe ser curiosa, analítica, con atención a los detalles, buena comunicación, capacidad de escuchar y ser objetivo, capacidad de monitoreo.
Respuesta 3	El departamento de calidad debe ser creado independientemente de los departamentos de desarrollo, esto para poder tener una imagen más global de los procesos y ser más objetivos independientemente de los proyectos.
Respuesta 4	Mediante la satisfacción de los clientes sobre la calidad de los productos entregados si los mismos cumplen con sus expectativas y necesidades.
Respuesta 5	El departamento debe trabajar en cada uno de los proyectos de modo que se pueda mostrar y medir su valor y beneficios. En el mercado siempre existen nuevas técnicas, herramientas y métodos por lo que esto permite que el profesional crezca tanto personal como profesionalmente.
Respuesta 6	N/A

Viviana Calvo	
Respuesta 1	<p>Las áreas que debe cubrir son:</p> <ul style="list-style-type: none"> ● QA manager (De ser posible): que sea el orquestador de los diferentes equipos de trabajo, sin estar necesariamente metido al detalle de lo operativo, pero sí en constante comunicación con los QA leads. Que colabore en procesos de control de calidad y mejora continua. ● QA leads: que estén al tanto de los detalles operativos y de proyectos que ayuden en el día a día a los QA Analyst proveen las guías respectivas para atención de tareas, resolución de problemas, etc. ● QA analyst: los ejecutores del día a día en el control de la calidad.
Respuesta 2	<p>Mi idea de perfil es:</p> <ul style="list-style-type: none"> ● Atención al detalle ● Muy buena comunicación ● Analíticos ● Buenos documentando ● Proactivos ● Buen trabajo bajo presión
Respuesta 3	<p>Al implementar un departamento de calidad se deben tomar en cuenta aspectos como:</p> <ul style="list-style-type: none"> ● Objetivos muy claros y alineados a los de la organización y del equipo. ● Definición de roles y responsabilidades. ● Claridad en los perfiles y sus opciones de desarrollo. ● Políticas adecuadas de medición de desempeño y planes de acción claros y documentados. ● Planes de comunicación claros. ● Canales de escalación definidos. ● Documentación apropiada. ● Definición de las herramientas a utilizar ● Documentación apropiada de procesos.
Respuesta 4	<p>Usando dashboard de desempeño.</p> <ul style="list-style-type: none"> ● Llevar estrictas revisiones del desempeño.

	<ul style="list-style-type: none"> ● Hacer exhaustivos análisis de hallazgos, causas y potenciales medidas correctivas. ● Dar seguimiento a las medidas correctivas. ● Teniendo retroalimentación del cliente. ● Revisiones 180 que incluya criterio de stakeholders internos.
Respuesta 5	<p>Definiéndole objetivos alineados tanto a los de la organización como los del equipo.</p> <ul style="list-style-type: none"> ● Proporcionando las herramientas necesarias que puedan limitar el desempeño. ● Procesos constantes de revisiones de rendimiento y planes de acción y/o reconocimiento con base en resultados.
Respuesta 6	N/A

Anexo 2: Anotaciones de las observaciones

<p>Fecha 06 de julio</p>	<p>Hora 4:00 pm</p>
<p>Observación</p> <p>Un QA lead para 6 QA's. No existe la figura de QA manager se maneja un departamento aparte del de desarrollo cada QA se asigna a un proyecto diferente, no trabajan todos los QA's en un solo proyecto.</p> <p>Uno de los QA's trabaja para 3 proyectos al mismo tiempo, comenta que tiene que crear los casos de prueba para un proyecto al cual a la semana le dedica 20 horas, a los otros 10 horas c/u por semana.</p> <p>El QA Lead, les recuerdo que deben realizar la certificación de ISTQB fundamentals a las 2 personas que faltan de certificarse y que después hay que presentarle al gerente que otras pruebas pueden realizar (esto con el fin de que la empresa pague las certificaciones)</p>	<p>Role</p>
<p>Comentarios adicionales empresa de migración de software.</p>	

Fecha 7 de julio	Hora 4:00 pm
Observación Hay un dashboard del proyecto en la herramienta que ellos usan (Jira), contiene un epic padre, que tiene historias y dentro de estas historias están las tareas de desarrollo y las de QA dentro. Dentro de las tareas de QA están los casos de prueba (test cases) El QA toma un caso de prueba para validarlo en el proyecto y lo termina, después de esto registra el tiempo que utilizó para ejecutarlo y lo sube a la tarea no al caso de prueba. * Metrics	Role
Comentarios adicionales Empresa J	

Fecha 10 de Julio	Hora 8:00 am
Observación un QA está instalando un servidor de prueba con la última versión disponible en producción del framework del software. Después de esto instala el nuevo software con el que van a trabajar de ahora en adelante QA es el encargado de tener los ambientes de prueba actualizados, los desarrolladores no les facilitan esto.	Role
Comentarios adicionales Empresa M.	

Fecha 12 de Julio	Hora 4:00 pm
Observación QA lead y QA senior se reúnen para tomar decisiones y planear la estrategia de testing para un nuevo proyecto hablan sobre los siguientes temas: <ul style="list-style-type: none"> - El tipo de proyecto que es./y el tamaño - Revisión de los requerimientos más complejos - los involucrados que van a trabajar este proyecto - El ambiente donde van a llevar el seguimiento de los casos de prueba y los defectos (el cliente quiere que sea en los ambientes de ellos y no en los de la empresa observada). - el alcance y el tipo de pruebas (manuales/automatizadas) - Métricas para evaluar y medir el éxito del proyecto - fechas de entrega. 	Role
Comentarios adicionales Empresa de migración	

Fecha	Hora
Viernes 14 de julio	3:00 pm
Observación	Role
<p>Entrevista de un QA. Ocupan contratar un QA mid.</p> <p>En la entrevista, están presentes: un desarrollador QA lead y un QA senior</p> <p>Se le pregunto al entrevistado acerca de:</p> <ul style="list-style-type: none"> - "veo que no ha terminado la U y ya ha pasado mucho tiempo, dice "ongoing": la va terminar?" - Tiene alguna certificación de calidad? - Le interesa sacar alguna? <ul style="list-style-type: none"> - Qué pasa si tiene q' entregar un trabajo y no tiene tiempo? - Qué pasa si no se lleva bien con un dev y tiene que trabajar con él? - Si hay que volver a la oficina que pasa? - Fuente acerca de algo q' no esta orgulloso y algo de lo que si? - Como se ve en 5 años? - Por qué quiere cambiar de trabajo? 	
Comentarios adicionales	
<p>sobre la experiencia solo preguntan como hacer una estrategia de pruebas, como crear cuales son los pasos p/ crear un caso de prueba y un bug. recuerda comenta q' no se enfocan tanto en esto ya que la industria en la que ellos se desarrollan es atipica (son juegos)</p>	
<p>Empresa J</p>	

Fecha Lunes 17 de julio	Hora 8:00 am
Observación <p>Puede observar la reunión semanal que hacen los QA's y el QA lead pl dar el status de sus proyectos. Esta duro 45 min.</p> <p>El QA lead les preguntaba acerca de:</p> <ul style="list-style-type: none"> • Cuantas pruebas faltan de realizar/corer • Cuantos erros hoy abierto / re abiertos • El nivel de gravedad de estos • El status del proyecto, en q' etopa esta este • Comentarios / observaciones adicionales. • en caso de que el proyecto tenga pruebas automatizadas como esta el status de estas refiriendose al CI/CD <p>☞ Dos proyectos estaban en rojo o incendiadas por lo que el jefe (QA lead) les pidio una llamada aparte pl revisar y buscar soluciones y no atrasar la llamada.</p> <p>Se revisaron 9 proyectos, 4 ya estaban finalizando.</p>	Role
Comentarios adicionales Empresa de migración de software	

Fecha 18 Julio	Hora 4:00pm
Observación <p>Hay un QA que tiene el rol de encargarse del Career path del departamento de SQA.</p> <p>Parece que este QA tiene bastante experiencia en el tipo de software que desarrollan en esta empresa. Él analiza las certificaciones y dice cuáles deberían de hacer los QA del equipo.</p> <p>Hoy están en clases, se están preparando para la certificación de mobile de ISTQB. Reciben clases una vez a la semana, en horario laboral.</p> <p>Al parecer la empresa les provee el profesor y si pasan el examen se los pagan, de lo contrario lo hace el empleado.</p> <p>* Mejora continua</p>	Role
Comentarios adicionales	
Empresa J	

Fecha 19 de Julio	Hora 4:00pm
Observación <p>La empresa envió un correo de nominación, este contiene un link que permite nominar a los compañeros q usted considera que lo hicieron bien este mes. No se permite que los empleados se auto nominen.</p> <p>El correo incluye los premios y la fecha de cierre de la nominación.</p>	Role
Comentarios adicionales Empresa J	

Fecha 21 de Julio	Hora 9:00 am
Observación <p>Uno de los QA ingresa a confluence (herramienta que sirve para documentar) y este esta documentando un nuevo proceso de QA. Este es acerca de la dinámica del "group testing"</p> <p>se lee que el "group testing" deben realizarlo al terminar el equipo de QA de revisar el software, se debe enviar un correo a todas los equipos involucrados que deben iniciar las pruebas con el objetivo de ve encontrar más bugs pl mejorarlo antes de salir a producción. (otros QA y desarrolladores de otros proyectos/equipos se involucran).</p>	Role
Comentarios adicionales	