**UADY**

**Facultad de Matemáticas**

**Licenciatura en Ingeniería de Software**

**Aseguramiento de la calidad**

**Documento de Aseguramiento de la Calidad**

**Profesor:**

M.I.T Edwin Jesús León Bojórquez

**Equipo 4**

**Integrantes:**

* Gómez Benítez Jonathan Gregorio
* Grimaldo Patiño Ricardo Alejandro
* Meza Magaña Joshua Immanuel
* Urtecho Quintal Rodrigo

**Fecha:**

09/05/2022­­­



Control de Documentación

Control de Configuración

|  |  |
| --- | --- |
| Título: | Documento de aseguramiento de la calidad |
| Referencia: | <https://github.com/RicGrimaldo/Control-de-versiones-del-documento-de-SQA> |
| Autor: | Joshua Immanuel Meza Magaña |
| Fecha: | 07 de mayo del 2022 |

Histórico de versiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versión** | **Fecha** | **Estado** | **Responsable** | **Nombre de archivo** |
| 1.2.0 | 16/05/2022 | A | Joshua Immanuel Meza Magaña | DocumentoDeSQA\_E4.docx |
| 1.1.0 | 09/05/2022 | A | Joshua Immanuel Meza Magaña | DocumentoDeSQA\_E4.docx |
| 1.0.0 | 07/05/2022 | A | Joshua Immanuel Meza Magaña | DocumentoDeSQA\_E4.pdf |

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

Histórico de cambios

|  |  |  |
| --- | --- | --- |
| **Versión** | **Fecha** | **Cambios** |
| 1.2.0 | 16/05/2022 | Se hacen los apartados de reporte de problemas y acción correctiva, herramientas, técnicas y metodologías, control de código, control de medios, control de registro, mantenimiento y retención, y entrenamiento. |
| 1.1.0 | 09/05/2022 | Se hacen los apartados de propósito, documentos referenciados, administración, documentación, estándares, prácticas, convenciones y métricas, revisiones y auditorías, y pruebas. |
| 1.0.0 | 07/05/2022 | Se genera por primera vez todo el documento de aseguramiento de la calidad. |

Índice

[Propósito 6](#_Toc103635257)

[Documentos referenciados 7](#_Toc103635258)

[Administración 9](#_Toc103635259)

[Documentación 9](#_Toc103635260)

[Propósito 9](#_Toc103635261)

[Mínimos requerimientos de documentación 9](#_Toc103635262)

[Especificaciones de los requerimientos de software 9](#_Toc103635263)

[Descripción del diseño de software 10](#_Toc103635264)

[Plan de verificación y validación de software 10](#_Toc103635265)

[Reporte de verificación y validación de software 10](#_Toc103635266)

[Documentación de usuario 10](#_Toc103635267)

[Plan de la administración de la configuración del software 11](#_Toc103635268)

[Otra documentación 11](#_Toc103635269)

[Plan de desarrollo de software 11](#_Toc103635270)

[Manual de estándares y procedimientos 11](#_Toc103635271)

[Plan de administración del proyecto de Software 11](#_Toc103635272)

[Manual de mantenimiento del software 11](#_Toc103635273)

[Documentación adicional sugerida 12](#_Toc103635274)

[Declaración de los requisitos de software 12](#_Toc103635275)

[Especificación de las interfaces externas 12](#_Toc103635276)

[Especificación de las interfaces internas 12](#_Toc103635277)

[Manual de operaciones 12](#_Toc103635278)

[Manual de instalación 12](#_Toc103635279)

[Manual de entrenamiento 12](#_Toc103635280)

[Plan de entrenamiento 12](#_Toc103635281)

[Plan de métricas de software 13](#_Toc103635282)

[Plan de seguridad del software 13](#_Toc103635283)

[Estándares, prácticas, convenciones y métricas 13](#_Toc103635284)

[Propósito 13](#_Toc103635285)

[Contenido 13](#_Toc103635286)

[Fase de requerimientos 13](#_Toc103635287)

[Fase de diseño 14](#_Toc103635288)

[Fase de implementación 14](#_Toc103635289)

[Fase de pruebas 14](#_Toc103635290)

[Fase de mantenimiento 14](#_Toc103635291)

[Documentación 14](#_Toc103635292)

[Métricas 15](#_Toc103635293)

[Revisiones y auditorías 15](#_Toc103635294)

[Propósito 15](#_Toc103635295)

[Requerimientos mínimos 15](#_Toc103635296)

[Revisión de los requerimientos de software 15](#_Toc103635297)

[Revisión del diseño preliminar 15](#_Toc103635298)

[Revisión de diseño crítico 16](#_Toc103635299)

[Revisión del plan de verificación y validación de software 16](#_Toc103635300)

[Auditoría funcional 16](#_Toc103635301)

[Auditoría física 16](#_Toc103635302)

[Auditorías en proceso 16](#_Toc103635303)

[Revisiones gerenciales 16](#_Toc103635304)

[Revisión del plan de administración de la configuración del software 16](#_Toc103635305)

[Revisión post mortem 16](#_Toc103635306)

[Otro 17](#_Toc103635307)

[Revisión de la documentación del usuario 17](#_Toc103635308)

[Auditoría del aseguramiento de la calidad 17](#_Toc103635309)

[Pruebas 17](#_Toc103635310)

[Reporte de problemas y acción correctiva 17](#_Toc103635311)

[Herramientas, técnicas y metodologías 18](#_Toc103635312)

[Herramientas 18](#_Toc103635313)

[Técnicas 18](#_Toc103635314)

[Metodologías 19](#_Toc103635315)

[Control de código 19](#_Toc103635316)

[Control de medios 20](#_Toc103635317)

[Acceso no autorizado 20](#_Toc103635318)

[Daño o degradación inadvertida 21](#_Toc103635319)

[Control de proveedores 21](#_Toc103635320)

[Colección de registros, mantenimiento y retención 21](#_Toc103635321)

[Colección de registros 21](#_Toc103635322)

[Mantenimiento de registros 22](#_Toc103635323)

[Retención de registros 22](#_Toc103635324)

[Entrenamiento 22](#_Toc103635325)

# Propósito

El objetivo del presente plan es establecer y proveer las actividades, contenido y preparación necesaria con el fin de garantizar la calidad de la Aplicación Web de lector de RSS feeds personalizable, y para ello, por cada actividad se describirá la metodología a realizar para garantizar la calidad de cada de los siguientes productos cubiertos por este plan de aseguramiento de la calidad:

* Documentación.
  + Especificaciones de los requerimientos de software.
  + Descripción del diseño de software.
  + Plan de verificación y validación de software.
  + Reporte de verificación y validación de software.
  + Documentación de usuario.
  + Plan de la administración de la configuración del software.
* Otra documentación.
  + Plan de desarrollo de software.
  + Manual de estándares y procedimientos .
  + Plan de administración del proyecto de Software.
  + Manual de mantenimiento del software.
* Documentación adicional sugerida.
  + Declaración de los requisitos de software.
  + Especificación de las interfaces externas .
  + Especificación de las interfaces internas.
  + Manual de operaciones.
  + Manual de instalación.
  + Manual de entrenamiento.
  + Plan de entrenamiento.
  + Plan de métricas de software.
  + Plan de seguridad del software.
* Estándares, prácticas, convenciones y métricas.
  + Plan de Pruebas del Sistema.
  + Documento de Análisis y Diseño.
  + Plan de Pruebas de Integración.
  + Componentes del sistema.
  + Pruebas.
  + Métricas utilizadas.

Independientemente del modelo de ciclo de vida seleccionado para el proyecto, las fases en las que se enfocará el plan de aseguramiento de la calidad serán las definidas en el Documento de Procesos, es decir:

* Fase de Inicio.
* Fase de Requerimientos.
* Fase de Análisis y Diseño.
* Fase de Codificación.
* Fase de Integración y Pruebas.
* Fase de Cierre.

A la vez, se asegurará la calidad de cada uno de los productos resultantes de dichas fases.

De esta forma y con este plan se espera que en el proyecto de la Aplicación Web de lector de RSS feeds personalizable se minimice el costo de garantizar la calidad de este último previniendo las causas de los errores y problemas para detectarlos y corregirlos de forma temprana, así como también revisar y auditar los productos y actividades de software para verificar que se cumplen los procedimientos y los estándares establecidos para el proyecto.

# Documentos referenciados

GeeksforGeeks. (2019). *Coding Standards and Guidelines*. <https://www.geeksforgeeks.org/coding-standards-and-guidelines/>

Git. (s.f.) *Documentation*. <https://git-scm.com/doc>

GitHub Docs. (2022). *Need help?* GitHub. <https://docs.github.com/en>

Gómez, J., Grimaldo, R., Meza, J. & Urtecho, R. (2022). *Documento De SQA de Procesos\_E4*. (v. 1.3.3).

Institute of Electrical and Electronics Engineers, International Electrotechnical Commission, & International Organization for Standardization. (2021). *ISO/IEC/IEEE International Standard - Software and systems engineering - Software testing -- Part 2: Test processes* (IEEE/ISO/IEC Standard No. 29119-2-2021).

Institute of Electrical and Electronics Engineers, International Electrotechnical Commission, & International Organization for Standardization. (2021). *IEEE/ISO/IEC International Standard - Software and systems engineering--Software testing--Part 4: Test techniques* (IEEE/ISO/IEC Standard No. 29119-4-2021).

Institute of Electrical and Electronics Engineers, International Electrotechnical Commission, & International Organization for Standardization. (2006). *Software Engineering — Software Life Cycle Processes — Maintenance* (IEEE/ISO/IEC Standard No. 14764-2006).

Institute of Electrical and Electronics Engineers. (1987). *IEEE Standard for Software Unit Testing* (IEEE Standard No. 1008-1987).

Institute of Electrical and Electronics Engineers. (1995). *IEEE Guide for Software Quality Assurance Planning* (IEEE Standard No. 730.1-1995).

Institute of Electrical and Electronics Engineers. (1998). *IEEE Recommended Practice for Software Requirements Specifications* (IEEE Standard No. 830-1998).

Institute of Electrical and Electronics Engineers. (1998). *IEEE Standard for a Software Quality Metrics Methodology* (IEEE Standard No. 1061-1998).

Institute of Electrical and Electronics Engineers. (2002). *IEEE Standard for Software Quality Assurance Plans*. (IEEE Std 730-2002)

Institute of Electrical and Electronics Engineers. (2008). *IEEE Standard for Software and System Test Documentation* (IEEE Standard No. 829-2008).

Institute of Electrical and Electronics Engineers. (2012). *IEEE Standard for Configuration Management in Systems and Software Engineering* (IEEE Standard No. 828-2012).

Institute of Electrical and Electronics Engineers. (2016). *IEEE Standard for System, Software, and Hardware Verification and Validation* (IEEE Standard No. 1012-2016).

International Electrotechnical Commission, & International Organization for Standardization. (2007). *Systems and software engineering — Measurement process* (ISO/IEC Standard No. 15939).

International Electrotechnical Commission, & International Organization for Standardization. (2019). *Software and systems engineering — Tools and methods for product line architecture design* (ISO/IEC Standard No. 26552:2019).

International Electrotechnical Commission, & International Organization for Standardization. (2017). *Systems and software engineering — Life cycle management — Part 5: Software development planning* (ISO/IEC Standard No. TS 24748-5:2017).

International Electrotechnical Commission, & International Organization for Standardization. (2016). *Systems and software engineering — Life cycle management — Part 6: System integration engineering* (ISO/IEC Standard No. TS 24748-6:2016).

International Electrotechnical Commission, & International Organization for Standardization. (2009). *Systems and software engineering — Life cycle processes — Project Management* (ISO/IEC Standard No. TS 16326:2009).

International Organization for Standardization, International Electrotechnical Commission & Institute of Electrical and Electronics Engineers. (2019). *ISO/IEC/IEEE International Standard - Systems and software engineering - Life cycle processes - Project management.* (IEEE/ISO/IEC 16326-2019).

International Organization for Standardization, International Electrotechnical Commission & Institute of Electrical and Electronics Engineers. (2018). *ISO/IEC/IEEE International Standard - Systems and software engineering - Requirements for acquirers and suppliers of information for users.* (IEEE/ISO/IEC 26512-2017).

International Organization for Standardization. (2015). *Quality management systems — Requirements* (ISO Standard No. 9001:2015).

Kummer, D. (2017) *git-flow cheatsheet*. <https://danielkummer.github.io/git-flow-cheatsheet/>

Martin, R. (2009). *Clean Code A Handbook of Agile Software Craftsmanship*. Prentice Hall.

Mojica, C. (2001). *ControlDeLaConfiguracion* (v. 1.0.0).

Mojica, C. (2001). *Documento de Especificación de Requerimientos de Software* (v. 0.1).

National Institute of Standards and Technology. (2006). *Guide for Developing Security Plans for Federal Information Systems* (NIST Standard No. SP800-18).

Software & Systems Engineering Standards Committee of the Institute of Electrical and Electronics Engineers Computer Society. (2009). *IEEE Standard for Information Technology—Systems Design— Software Design Descriptions*. (IEEE Standard 1016-2009).

Software & Systems Engineering Standards Committee of the Institute of Electrical and Electronics Engineers Computer Society. (2008). *IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits)*. (IEEE Std 1028-2008).

# Administración

La administración describe cada elemento importante de la **organización** que influye en la calidad del Software, lista todas las **actividades** cubiertas por el plan de SQA, identifica las **responsabilidades** específicas de la organización para cada una de las tareas y provee los **costos y recursos** estimados que serán necesarios para el aseguramiento de la calidad junto a las tareas del control de la calidad.

El formato y contenido necesarios están descritos en el estándar IEEE/ISO/IEC 16326-2019 (ISO/IEC/IEEE International Standard - Systems and software engineering - Life cycle processes - Project management), así como las tareas a llevar a cabo por parte del plan de SQA están descritas desde el apartado 3.4 hasta 3.15 del estándar IEEE Std 730-2002 (IEEE Standard for Software Quality Assurance Plans).

# Documentación

## Propósito

El presente apartado tiene como propósito identificar la documentación, sea impresa o electrónica, que rige el desarrollo, verificación y validación, y mantenimiento del software; aunado a ello, la lista de los documentos que se revisarán o auditarán para verificar su adecuación. Por cada uno de los documentos listados, se identificará las revisiones o auditorías a llevar a cabo junto a los criterios mediante los cuales se confirmará la adecuación, con referencia a la sección de revisiones y auditorías del presente plan de SQA.

En el caso que no haya una verificación o validación independiente, entonces se deberán identificar los procedimientos de aseguramiento de la calidad que se utilizarán en el proyecto.

## Mínimos requerimientos de documentación

### Especificaciones de los requerimientos de software

Las especificaciones de los requerimientos de software (ERS) son la obtención y definición de aquellas características que debe poseer el producto final de software para poder cubrir la o las necesidades de los interesados y resolver la o las problemáticas existentes, y para ello se deberán estipular todas las regulaciones impuestas por las leyes o el cliente sobre el proyecto, conforme al estándar IEEE-STD-830-1998 (IEEE Recommended Practice for Software Requirements Specifications).

Asimismo, el documento de especificación de requerimientos se encuentra anexado como “Documento de Especificación de Requerimientos de Software”.

### Descripción del diseño de software

De acuerdo con la IEEE, la descripción del diseño de software es la representación de un sistema de software que es usado como medio de comunicación de la información del diseño de software, es decir, debe representar la estructura adecuada del software de tal forma que satisfaga los requerimientos establecidos en el ERS.

Su contenido incluye la descripción de cada uno de los componentes y subcomponentes del diseño de software, interfaces internas y base de datos incluidas.

El estándar IEEE 1016-2009 (IEEE Standard for Information Technology--Systems Design--Software Design Descriptions) describe los diseños de software y establece el contenido de la información y la organización de una descripción de diseño de software.

### Plan de verificación y validación de software

Este plan describe el plan general para la verificación y validación del software y lo que podría ser producido y revisado de manera incremental. Asimismo, se describen todas las tareas, métodos y criterios necesarios para realizar las verificaciones y validaciones adecuadamente.

Debido a que el plan de verificación y validación de software especifica los requisitos mínimos para la documentación de pruebas, es necesario consultar el estándar IEEE 829-2008 (IEEE Standard for Software and System Test Documentation).

Por otro lado, con el fin de asegurar que, durante el desarrollo de cada producto, se hayan seguido y aplicado correctamente los estándares establecidos y aprobar las versiones finales de estos, se aplicará el estándar IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits) durante cada proceso de verificación y validación de cada uno de los artefactos con el fin de revisar dichos productos.

Además, cada verificación y validación debe estar estructurada y llevada a cabo conforme a lo establecido en el estándar IEEE 1012-2016 (IEEE Standard for System, Software, and Hardware Verification and Validation).

### Reporte de verificación y validación de software

Su propósito es describir los resultados obtenidos de la ejecución del plan de verificación y validación de software y resumiendo el estado de este último, cuyo contenido está descrito en el estándar IEEE 1012-2016 (IEEE Standard for System, Software, and Hardware Verification and Validation).

### Documentación de usuario

La documentación de usuario guía a los usuarios en la instalación, operación, administración y mantenimiento (no se aplica cuando se modifica el código fuente del software) de los productos de software.

Su contenido debe especificar y describir los datos requeridos y las entradas de control, las secuencias de entrada, las opciones, las limitaciones del programa y otras actividades o elementos necesarios para la ejecución exitosa del software. Asimismo, se identificarán todos los mensajes de error y se describirán las acciones correctivas.

Esta documentación es aplicable a cualquier parte del software integrado con el que el usuario interactúe directamente. Para saber la estructura y contenido más detallado del mismo, se puede consultar el estándar IEEE/ISO/IEC 26512-2017 (ISO/IEC/IEEE International Standard - Systems and software engineering - Requirements for acquirers and suppliers of information for users).

### Plan de la administración de la configuración del software

Este plan debe documentar los métodos que se usarán para identificar a los ítems de software, el control e implementación de cambios, así como registrar e informar acerca del estado de la implementación de cambios.

Adicionalmente, describe las tareas, la metodología y las herramientas necesarias para garantizar que los procedimientos y controles adecuados de la Gestión de la configuración del software estén documentados y se implementen correctamente.

El contenido y estructura necesaria para realizar dicho plan está especificado en el estándar IEEE 828-2012 (IEEE Standard for Configuration Management in Systems and Software Engineering).

## Otra documentación

### Plan de desarrollo de software

El plan de desarrollo de software es, según el estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning), el documento de planificación de más alto nivel dentro del desarrollo de un producto de software que gobierna la ejecución de un proyecto. El contenido necesario para asegurar un buen plan de desarrollo se especifica dentro del estándar que se mencionó anteriormente y su estructura e implementación para el cumplimiento de las tareas relacionadas al desarrollo de software de un proyecto se describen en el estándar ISO/IEC/IEEE 24748-5-2017 (International Standard - Systems and Software Engineering--Life Cycle Management--Part 5: Software Development Planning).

### Manual de estándares y procedimientos

Este manual debe proveer información a detalle de cada uno de los estándares y procedimientos empleados en el desarrollo del proyecto, dicha información describe el uso de los estándares, así como, de los procedimientos, es decir, dónde y cómo son utilizados en el proyecto. El mínimo de estándares requeridos que deben considerarse en el manual se describe en el punto 3.5.2 del estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning).

### Plan de administración del proyecto de Software

Este documento puede usarse en lugar del Plan de desarrollo de software dado que posee mayor alcance y propósito. El Plan de administración del proyecto de Software debe identificar todas las actividades técnicas y administrativas que estén asociadas al desarrollo de software; el formato y contenidos necesarios son descritos en el estándar ISO/IEC/IEEE 16326-2009 (International Standard - Systems and Software Engineering--Life Cycle Processes--Project Management).

### Manual de mantenimiento del software

El manual describe los procedimientos a realizar para el mantenimiento y soporte del producto de software a desarrollar en el proyecto siguiendo las actividades descritas en el estándar ISO/IEC/IEEE 14764-2006 (International Standard for Software Engineering - Software Life Cycle Processes – Maintenance).

## Documentación adicional sugerida

### Declaración de los requisitos de software

La declaración de los requisitos de software sirve para dar seguimiento a la solicitud de requisitos realizados por usuarios a un sistema que ya es operacional, requisitos solicitados en un sistema que no posee un documento de Especificación de Requisitos de Software o requisitos que se le serán entregados a un subcontratista. Los elementos necesarios para la descripción de estas declaraciones se encuentran descritos en el punto 3.4.4.1 del estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning).

### Especificación de las interfaces externas

Las interfaces externas deben ser especificadas en el documento de Especificación de Requisitos de Software o en el Documento de Control de las Interfaces cuya estructura se basará del formato implementado en el documento “ERGO\_D1\_3\_InterfaceControlDocument\_V2.2.pdf” y tendrá como contenidos a considerar los descritos en el punto 3.4.4.2 del estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning).

### Especificación de las interfaces internas

Las interfaces internas deben ser especificadas en el documento de Especificación de Requisitos de Software o en el Documento de Control de las Interfaces cuya estructura se basarán del formato implementado en el documento “ERGO\_D1\_3\_InterfaceControlDocument\_V2.2.pdf” y tendrá como contenidos a considerar los descritos en el punto 3.4.4.3 del estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning).

### Manual de operaciones

Los contenidos del manual de operaciones se describen en el punto 3.4.4.4 “Operations Manual” del estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning).

### Manual de instalación

El manual de instalación debe contener las instrucciones correspondientes para la instalación del producto de software en el equipo computacional para el cual fue diseñado. Los contenidos detallados necesarios para el manual están descritos en el punto 3.4.4.5 “Installation Manual” del estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning).

### Manual de entrenamiento

El manual de entrenamiento contiene información necesaria para el entrenamiento de los usuarios y operadores en el uso del sistema. Los contenidos mínimos esperados en este manual se encuentran enlistados en el punto 3.4.4.6 “Training Manual” del estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning).

### Plan de entrenamiento

El plan de entramiento es necesario cuando el producto de software desarrollado posee características de interacción complejas o desconocidas para los usuarios. Los contenidos necesarios para este documento se describen en el punto 3.4.4.7 “Training Plan” del estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning).

### Plan de métricas de software

El plan de métricas, especificado como el Plan de Pruebas del Sistema en el Documento de Procesos, el cual está a su vez regulado por los estándares ISO/IEC 15939 (Systems and software engineering — Measurement process) y el IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning), los cuales especifican procesos para definir las métricas correctas para la información que se necesita y que información debería contener el documento, respectivamente.

### Plan de seguridad del software

El plan de seguridad del software consiste en, según el estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning), un documento que define cómo deberá asegurarse que el software y la información serán protegidos de accesos no autorizados o daños. Dicho documento está regulado por definición y contenidos por el estándar anteriormente mencionado, y en estructura e implementación por el estándar NIST SP800-18 (Guide for Developing Security Plans for Federal Information Systems).

# Estándares, prácticas, convenciones y métricas

## Propósito

Esta sección del Plan de Aseguramiento de la Calidad del Software es la encargada de definir todos aquellos estándares (requerimientos obligatorios), prácticas (acercamiento recomendado), convenciones (guías aceptadas) y métricas (sistema de medida) que serán empleados en todo aquello que involucre el proyecto, incluida la administración y los proveedores. De igual forma, deberá especificar todas las fases del ciclo de vida en las que aplica, así como de qué forma se va a asegurar y monitorear la conformidad.

## Contenido

### Fase de requerimientos

Etapa descrita en el Documento de Procesos como parte de una de las actividades del proceso de Desarrollo y Mantenimiento de Software, perteneciente a la categoría de Operación, y denominada como actividad de Requerimientos. Como tal, sus dos productos principales son los documentos de Especificación de Requerimientos y el Plan de Pruebas del Sistema.

Tal y como el Documento de Procesos menciona, el documento de Especificación de Requerimientos está regulado por el estándar IEEE 830-1998 (IEEE Recommended Practice for Software Requirements Specifications), además de contar con la documento anexada “Documento de Especificación de Requerimientos de Software.docx” para facilitar su desarrollo, mientras que el Plan de Pruebas del Sistema está regulado por los estándares ISO/IEC/IEEE 29119-2:2021 (Software and systems engineering — Software testing — Part 2: Test processes) y el ISO/IEC/IEEE 29119-4:2021 (Software and systems engineering — Software testing — Part 4: Test techniques).

### Fase de diseño

Etapa descrita en el Documento de Procesos como parte de una de las actividades del proceso de Desarrollo y Mantenimiento de Software, perteneciente a la categoría de Operación, y denominada como actividad de Análisis y Diseño. Como tal, sus dos productos principales son los documentos de Análisis y Diseño y el Plan de Pruebas de Integración.

En cuanto a los productos, el documento de Análisis y Diseño está regulado por el estándar ISO/IEC 26552:2019 (Software and systems engineering — Tools and methods for product line architecture design), mientras que el de Plan de Pruebas de Integración está regulado por el ISO/IEC TS 24748-6:2016 (Systems and software engineering — Life cycle management — Part 6: System integration engineering).

### Fase de implementación

Etapa descrita en el Documento de Procesos como parte de una de las actividades del proceso de Desarrollo y Mantenimiento de Software, perteneciente a la categoría de Operación, y denominada como actividad de Codificación. Como tal, los productos que resultan de esta fase son los componentes del sistema que fueron definidos en la actividad de Análisis y Diseño.

La implementación deberá seguir un conjunto de reglas de codificación bien definidas y ampliamente utilizadas, los cuales tienen el propósito de estandarizar la forma en que se codifica y de mejorar la calidad del código. A continuación, se listan las principales fuentes a tomar en cuenta:

* Libro “Clean Code A Handbook of Agile Software Craftsmanship” de Martin R.
* Convenciones establecidas del lenguaje de programación seleccionado para el proyecto.
* Página web “Coding Standards and Guidelines” de GeekforGeeks.

### Fase de pruebas

Etapa descrita en el Documento de Procesos como parte de una de las actividades del proceso de Desarrollo y Mantenimiento de Software, perteneciente a la categoría de Operación, y denominada como actividad de Integración y Pruebas. Como tal, sus productos principales son los documentos de Manual del Usuario, Manual de Operación y el Manual de Mantenimiento, los cuales ya han sido definidos a detalle en secciones anteriores.

Tal y como está descrito en el Documento de Procesos, las pruebas están reguladas por el estándar IEEE 1012-2016 (IEEE Standard for System, Software, and Hardware Verification and Validation).

### Fase de mantenimiento

Esta etapa está descrita y regulada por el estándar Std 14764-2006 (Software Engineering — Software Life Cycle Processes — Maintenance).

### Documentación

Todos los documentos deben cumplir con la obligatoriedad de aplicar los principios de la Gestión de la Configuración y del Control de la Documentación, tal y como está establecido en los estándares IEEE 828-2012 (IEEE Standard for Configuration Management in Systems and Software Engineering) e ISO 9001:2015 (ISO Standard For a Quality Management System), respectivamente. Como parte del cumplimiento de esta norma, se tiene el anexo ControlDeLaConfiguracion, el cual debe colocarse en la hoja siguiente a la portada.

### Métricas

Esta etapa está descrita y regulada por el estándar IEEE 1061-1998 (IEEE Standard for a Software Quality Metrics Methodology).

# Revisiones y auditorías

## Propósito

Ya que los productos de software producidos durante el proceso del ciclo de vida del software deben ser evaluados y auditados para determinar la extensión del progreso, evaluar la adecuación técnica del trabajo y su conformidad con los requerimientos de software y estándares, esta sección tiene la finalidad de identificar las auditorias y revisiones técnicas específicas a ser realizadas con respecto a los planes de desarrollo de software, los cronogramas y el ambiente, así como definir el proceso de implementación, los participantes y sus responsabilidades, quién generará los reportes de resultados, qué formato deben tener, quién lo debe recibir, qué actividades de seguimiento deben ser aplicadas, entre otros detalles.

## Requerimientos mínimos

### Revisión de los requerimientos de software

Consiste en la evaluación de la Especificación de los Requerimientos de Software (ERS) para asegurar la adecuación, la viabilidad tecnológica y la completitud de los requerimientos. Los atributos para evaluar en el ERS están definidos en el estándar IEEE Std 830-1998 (IEEE Recommended Practice for Software Requirements Specifications), el cual dice que un ERS debe ser correcto, inequívoco, completo, consistente, debe delinear lo importante, comprobable, modificable e identificable.

Los resultados de la revisión deben documentarse en un informe RRS (Revisión de los requerimientos del software) que identifique todas las deficiencias descritas en la revisión, así como proporcionar un plan y un cronograma para la acción correctiva. Además, deberá especificar la decisión sobre si debe o no proceder con la corrección, basado en estimaciones de costos, estudios de factibilidad y evaluaciones de riesgo del proyecto.

### Revisión del diseño preliminar

Para asegurar la calidad en la construcción de los diferentes ítems a revisar durante el diseño preliminar, se seguirá el estándar IEEE Std 830-1998 (IEEE Recommended Practice for Software Requirements Specifications) para su verificación, mismo que contiene una serie de guías para alcanzar los diferentes criterios de calidad como pueden ser la consistencia y la compatibilidad entre los requerimientos y la relación del sistema con otras interfaces.

Los resultados deben documentarse en un informe RDP (Revisión de Diseño Preliminar) que identifique todas las deficiencias descubiertas durante la revisión y un plan y calendario para la acción correctiva. Dicho informe estará realizado con base al anexo DESIGN REVIEW CHECKLIST, que contiene los elementos a revisar apegados al estándar mencionado con anterioridad.

### Revisión de diseño crítico

Una vez aplicadas las acciones correctivas descritas durante la revisión preliminar, nuevamente se generará un reporte siguiendo las especificaciones y consideraciones mencionadas con anterioridad, esta vez bajo el nombre RDC (Revisión de Diseño Crítico) siguiendo el mismo anexo DESIGN REVIEW CHECKLIST en dicho informe se debe identificar todas las deficiencias descubiertas durante la revisión y un plan y cronograma para las acciones correctivas.

### Revisión del plan de verificación y validación de software

Para realizar las verificaciones y validaciones del software se seguirán las normas y lineamientos establecidos en el estándar IEEE Std 1012-2016 (IEEE Standard for System, Software, and Hardware Verification and Validation), mismo que contiene una serie de formatos y consideraciones para tener en cuenta, los cuales serán aplicados para asegurar la calidad.

### Auditoría funcional

Para realizar la auditoría funcional, misma que abarca documentos como la especificación de requisitos, se seguirá el estándar IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits) que contiene una serie de consideraciones y normas para llevar a cabo la auditoría de los documentos funcionales que se esperan abarcar en el plan de SQA.

### Auditoría física

El estándar mencionado para la auditoría funcional es igualmente aplicable a los productos de la auditoría física, por lo que se hará uso de este para llevar a cabo esta actividad.

### Auditorías en proceso

El resultado de todas las auditorías en proceso son medidas de qué tan bien está funcionando el proceso de desarrollo. Deben documentarse en informes de auditoría continuos que identifiquen todas las discrepancias encontradas y los planes y calendarios de resolución. Para ello, se hará uso del documento IT AUDIT CHECKLIST, mismo que se encuentra como anexo y puede ser ajustado en caso de hacer falta algún parámetro.

### Revisiones gerenciales

Periódicamente se realizan revisiones gerenciales para evaluar la ejecución de todas las acciones y los elementos identificados en el SQAP. Estas revisiones serán realizadas por un elemento organizacional independiente de la unidad que se está revisando, o por un tercero calificado. Para ello, nuevamente se hará uso de los lineamientos establecidos en el estándar IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits), mismo que es aplicable para este proceso.

### Revisión del plan de administración de la configuración del software

La revisión se lleva a cabo para garantizar que los procedimientos de gestión de la configuración del software descritos sean adecuados y proporcionen el control necesario sobre la documentación y el código. Para ello se hará uso del estándar IEEE STD 828-2012 (IEEE Standard for Configuration Management in Systems and Software Engineering) para la revisión, por lo que se espera que cumpla con los lineamientos que se establecen con base a las guías establecidas.

### Revisión post mortem

Esta revisión se lleva a cabo al final del proyecto para evaluar las actividades de desarrollo implementadas en ese proyecto y proporcionar recomendaciones para la acción apropiada. Por lo que para la llevar a cabo dichas recomendaciones se hará uso del documento POSTMORTEM, mismo que se encuentra como anexo.

## Otro

### Revisión de la documentación del usuario

La revisión se lleva a cabo para determinar la capacidad técnica del enfoque y el diseño de la documentación tal como se describe en las versiones preliminares de la documentación del usuario (copia electrónica o copia impresa). Entonces, haciendo se llevará acabo haciendo uso del estándar IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits), mismo que es aplicable a este producto.

### Auditoría del aseguramiento de la calidad

Haciendo nuevamente empleo del estándar mencionado en la revisión anterior, se aplicarán los lineamientos necesarios para realizar la auditoría del aseguramiento de la calidad, ya que en este se hace referencia a que el mismo estándar usado para la implementación de este plan IEEE STD 730-1995 (IEEE Guide for Software Quality Assurance Planning), es candidato para auditar o revisar con estas normas.

# Pruebas

Para la definición de los diferentes tipos de pruebas y su implementación se seguirá el estándar IEEE Std 1008-1987 (IEEE Standard for Software Unit Testing), mismo que describe cómo determinar las pruebas que se harán, el diseño, planificación y ejecución de estas. Además, estas pruebas serán documentadas con base al estándar IEEE 829-2008 (IEEE Standard for Software and System Test Documentation). Por último, la correcta construcción del plan de pruebas y su documentación será verificada y validada con base al estándar IEEE Std 1012-2016 (IEEE Standard for System, Software, and Hardware Verification and Validation).

# Reporte de problemas y acción correctiva

Cuando se identifique algún problema durante cualquier fase del desarrollo, es importante abordarlo y de ser posible eliminarlo, para ello el sistema de notificación de problemas y acciones correctivas será implementado como parte de la gestión de configuración de software y documentado en el mismo plan. Además, estos reportes serán llevados a cabo una vez finalizado cada producto o etapa, exceptuando momentos críticos donde algún problema evite el correcto desarrollo de lo que se esté haciendo.

Para generar un reporte de problemas se seguirá el documento Corrective Action Report, mismo que especifica la información necesaria para generar el reporte y las acciones correctivas presentando una serie de lineamientos y procesos, los encargados, la evidencia, dónde se encontró el problema y las acciones que llevar a cabo cabe destacar que este documento está hecho con base al ISO 9001:2015 (Sistemas de Gestión de la Calidad). Se puede encontrar como anexo.

# Herramientas, técnicas y metodologías

## Herramientas

Dentro de las herramientas a utilizar para mejorar el proceso de aseguramiento de la calidad del software se encuentran las siguientes:

* GitHub: Herramienta que permitirá el trabajo colaborativo en la nube del equipo de desarrollo, del personal encargado en la gestión de la documentación resultante de las diversas fases del ciclo de vida de un proyecto de software y del administrador del proyecto, al permitir la visualización de cambios, adiciones y eliminaciones dentro de los contenidos de los archivos y documentos, así como, la capacidad de identificar al responsable de dichos cambios.
* Git: Herramienta que permite un control de versiones de forma local de los archivos y documentos relacionados con el producto de software, tales como el código fuente, documentos de control de versiones, documentos de ERS, entre otros productos que componen al software y su configuración.
* Visual Studio Code: Editor de código ligero con gran cantidad de extensiones que facilitan el desarrollo de código de software en diferentes lenguajes de programación, así como marcos de trabajo y otras herramientas como Git.

## Técnicas

Como parte del aseguramiento de la calidad del software se implementarán las siguientes técnicas:

* Auditorías: Implica la inspección de los productos de trabajo y su información relacionada para determinar si el conjunto de procesos estándares se ha seguido. Se realizará conforme al estándar IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits).
* Revisiones: Reunión de inspección del producto de software con participación de los interesados internos y externos para obtener sus comentarios, así como su aprobación. Se realizará conforme al estándar IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits).
* Inspección de código: Revisión formal que incluye pruebas estáticas para la búsqueda de bugs y disminución de defectos en etapas posteriores. Se realiza por personal entrenado con un nivel igual o superior en experiencia que el autor del código. La inspección se realizará conforme a lo recomendado en el estándar IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits). Y las pruebas conforme a lo especificado en el estándar IEEE 829-2008 (IEEE Standard for Software and System Test Documentation).
* Inspección de diseño: Inspección formal que verifica que se haya hecho un buen trabajo en un listado de áreas del diseño de la estructura del software. Dicha inspección se realizará conforme al estándar IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits).
* Pruebas funcionales: Técnica que verifica lo que hace el sistema sin considerar cómo lo hace. Se enfoca principalmente en la puesta en prueba de las funcionalidades y características del sistema. Esta prueba de caja negra se realizará conforme al estándar IEEE 829-2008 (IEEE Standard for Software and System Test Documentation).
* Estandarización: Actividad que disminuye la ambigüedad, así como las suposiciones en los productos de trabajo del software, para así, mejorar en el aseguramiento de la calidad.
* Pruebas de estrés: Prueba en la que se lleva al límite al sistema con una sobrecarga de trabajo para determinar sus capacidades. Los resultados se presentarán conforme al estándar IEEE 829-2008 (IEEE Standard for Software and System Test Documentation).

## Metodologías

La metodología a utilizar se basará en el uso de GitHub para un control de versiones minucioso aplicando diferentes técnicas de las antes mencionadas en cada versión de la configuración del software con el objetivo de mantener los productos dentro de la conformidad de los planes y contratos.

Las técnicas a implementar en la finalización de cada versión son:

* Auditorías de los documentos y productos que recibieron cambios significativos en relación con la versión anterior de la configuración del software.
* Revisiones de los documentos para confirmación por parte de los interesados acerca de los cambios realizados y para obtener comentarios sobre propuestas de mejora. De igual forma se revisará que los mecanismos de control de versiones del equipo de desarrollo se hayan respetado para la entrega de la nueva versión.
* Pruebas funcionales para aseguramiento de las funcionalidades entregadas en la nueva versión de la configuración del software.
* Inspección de diseño de las nuevas funcionalidades entregadas para determinación de cambios necesarios que provean mayor eficiencia en dicha funcionalidad.

Se consideraron los elementos necesarios en la metodología mencionados en el punto 3.9.3 del estándar IEEE Std 730.1-1995 (IEEE Guide for Software Quality Assurance Planning).

# Control de código

Cualquier código desarrollado por la organización debe de seguir los principios de la Gestión de la Configuración, tal y como está estipulado en el estándar ISO/IEC/IEEE 90003:2018 (Software engineering — Guidelines for the application of ISO 9001:2015 to computer software). Dicho esto, se emplearán herramientas como el software Git, la biblioteca git-flow y el repositorio GitHub, los cuales sirven para manejar versiones de código, aplicar principios de desarrollo organizado y guardar archivos (y su historial de versiones) en la nube. El repositorio de cada producto de software de los proyectos específicos deberá ser privado y estar creado con el usuario de la empresa, y para garantizar el acceso del líder de proyecto y el equipo de desarrollo, obtendrán los privilegios de contribuidor durante el tiempo que trabajen con dicho software. Cada que una versión estable sea finalizada y haya sido verificada por el equipo de pruebas y aprobada por el cliente, el líder de proyecto deberá generar una nueva liberación, la cual puede ser de utilidad para rastrear las diversas versiones estables del producto o para poder entregar copias a los clientes sin a la necesidad de que tengan acceso al repositorio. Entre algunas de las reglas que serán necesarias de aplicar en el repositorio se tiene que la rama principal no pude ser modificada directamente, que no se pueden mandar los cambios a la rama principal de forma directa (se tiene que generar una revisión de cambio que debe ser aceptado por el líder del proyecto específico) y que las solicitudes de cambio no pueden ser aceptadas por el mismo usuario que las generó. Finalmente, cada cambio realizado en el código afectará a la documentación de la rastreabilidad de los requisitos de software.

# Control de medios

En esta sección se establecen los métodos y facilidades a usar para identificar los medios para cada producto de computadora y la documentación requerida para almacenar los medios, incluyendo el proceso de copia y recuperación, así como también el programa físico de la media de computadora de un acceso no autorizado y daño inadvertido o degradado durante todas las fases del ciclo de vida del software.

Se deberá garantizar que:

* La recuperación y almacenamiento del software están asegurados.
* El software es accesible únicamente por los que necesitan el acceso.
* Se proporciona tanto almacenamiento como recuperación fuera del sitio para software crítico, así como copias del código de referencia.
* El contorno está controlado de tal forma que los medios físicos en los que el software está almacenado no se degraden.
* Una descripción sea proporcionada de cómo se asegura el cumplimiento de los anteriores puntos.

Todas las actividades y procesos con respecto a los puntos anteriores están relacionados con el *Plan de la administración de la configuración del software*, cuyo formato y contenido estás especificado en su respectiva sección.

Por otra parte, todos los documentos entregables del aseguramiento de la calidad se documentarán en alguna de las siguientes aplicaciones de software de la compañía de Microsoft:

* Word.
* Excel.
* Power Point.

Para todos los documentos entregables se asegurará tener una copia de seguridad digital, a menos que se especifique lo contrario.

## Acceso no autorizado

Para el acceso no autorizado, las actividades y métodos con respecto a la seguridad del sistema, es necesaria la creación del *Plan de Seguridad del Software*, contenido y actividades especificadas en su respectiva sección.

A través de dicho plan, se identificarán las actividades necesarias para llevar a cabo la protección y seguridad del programa, y a su vez, se definen las responsabilidades y comportamiento esperado de cada uno de los que tengan acceso al programa.

## Daño o degradación inadvertida

Con el fin de minimizar el daño o degradación inadvertida se deberá proveer técnicas adecuadas de la gestión de la configuración, así como lugares de almacenamiento seguros debido a factores externos (tales como bóvedas a prueba de incendios, revisión periódica para garantizar el uso de prácticas ambientales, tener en un ambiente adecuado a las computadoras y elementos de hardware, entre otros).

Todas las actividades de SQA realizadas para verificar la implementación adecuada de los procedimientos que buscan minimizar el daño o degradación de los medios es necesario documentarlas en base a lo establecido en el *Plan de Seguridad del Software.*

# Control de proveedores

La empresa realizará una auditoría del Plan de Aseguramiento de la Calidad del Software, así como de los Planes de Medición del Software para determinar si se adecuan o asemejan con los propios. Dicha auditoría se realizará siguiendo lo recomendado en el estándar IEEE Std 1028-2008 (IEEE Standard for Software Reviews and Audits).

En caso de que el proveedor no posea un Plan de Aseguramiento de la Calidad del Software que se adecue o asemeje al nuestro, podrá evaluar los productos desarrollados que se desean utilizar conforme al presente Plan o deberá realizar la búsqueda de otro proveedor que cubra los requisitos de Planificación, conforme a los establecido en el estándar IEEE Std 730-2002 (IEEE Standard for Software Quality Assurance Plans).

# Colección de registros, mantenimiento y retención

Las siguientes personas tendrán acceso a la colección de documentación asociada al proyecto:

* Responsable de Administración del Proyecto Específico
* Cliente (Únicamente a los que el contrato especifique)
* Administrador de la Configuración del Software
* Administrador de Control de Documentos

## Colección de registros

Los documentos cuyas versiones finales estarán en la colección son:

* Especificación de Requisitos de Software
* Plan de Aseguramiento de la Calidad
* Configuración de Software
* Plan de Pruebas de Sistema
* Reportes
* Manuales

## Mantenimiento de registros

Los documentos se almacenarán de forma digital en un repositorio privado en Github, cuyo acceso está limitado a las personas anteriormente mencionadas.

## Retención de registros

A pesar de no existir un tiempo de vida definido en el repositorio, la colección de registros estará disponible al menos durante el tiempo de vida del sistema o bien hasta que la administración decida destruirlos.

# Entrenamiento

Primero, se debe evaluar la necesidad de implantar el entrenamiento o la actualización al personal encargado de realizar las actividades definidas en este plan de Aseguramiento de la Calidad, identificando la actividad, las habilidades requeridas para dicha tarea, y las habilidades del personal designado para realizar la tarea; todo esto con el fin de identificar de manera rápida las necesidades de la formación de cada actividad e individuo en caso de ser necesario, sea entrenamiento o actualización del personal.

Una vez identificada la necesidad de implementar el entrenamiento (o actualización) al personal de SQA, es importante considerar las herramientas, técnicas y metodologías para las capacitaciones necesarias para la capacitación, como es el caso del plan de entrenamiento (más información en su respectiva sección del presente plan). En este caso, ya sea que programas existentes de entrenamiento sean adaptados, o bien, desarrollar nuevos programas, siendo en ambos casos con el fin de satisfacer las necesidades del entrenamiento.