Universidad Autónoma de Yucatán

Facultad de Matemáticas

Licenciatura en Ingeniería de Software

Aseguramiento de la calidad

Modelo de Procesos

Maestro: Edwin Jesús León Bojórquez

Universidad Autónoma de Yucatán

Alumnos:

* Andrade Ac Edwin Alonso
* Castrejón Cervantes Rodrigo Alejandro
* Cetina Escalante Fernan Enrique
* Gómez Gómez Horacio Jahir
* Pliego Vega Daniel

Fecha de entrega:

7 de marzo de 2023

**Contenido**

[Desarrollo y Mantenimiento de Software 2](#_Toc129102681)

[Gestión de la Configuración 2](#_Toc129102682)

[Estructura del documento: 2](#_Toc129102683)

[Gestión de la Configuración de Software: 2](#_Toc129102684)

[Definición general del proceso 5](#_Toc129102685)

[Propósito: 5](#_Toc129102686)

[Descripción: 5](#_Toc129102687)

[Objetivos: 6](#_Toc129102688)

[Indicadores: 6](#_Toc129102689)

[Metas cuantitativas: 6](#_Toc129102690)

[Responsabilidad y autoridad: 6](#_Toc129102691)

[Entradas: 7](#_Toc129102692)

[Salidas: 7](#_Toc129102693)

[Documentos de control – Fase de pruebas: 9](#_Toc129102694)

[Roles: 10](#_Toc129102695)

[Verificaciones y validaciones: 13](#_Toc129102696)

[Incorporación a la Base de Conocimiento: 15](#_Toc129102697)

[Guías de ajuste 16](#_Toc129102698)

[Bibliografía 18](#_Toc129102699)

[Índice de Estándares 20](#_Toc129102700)

# **Desarrollo y Mantenimiento de Software**

## **Gestión de la Configuración**

### Estructura del documento:

La estructura del documento siguiente está basada en la estructura básica de apartados obligatorios como opcionales propuestos por el estándar **IEEE 1063-2001[1]** adaptado para el uso del modelo de procesos basado en MoProSoft, siendo que el alcance del estándar se encuentre por fuera de la gestión de configuración de la documentación.

### Gestión de la Configuración de Software:

Para la gestión de la configuración de software, sea esto las acciones de administración de cambios, documentación adecuada de los cambios, registro y almacenamiento de las versiones del SCI aprobados al igual que suministrar las copias de versiones registradas a través del ciclo de vida del este elemento de la configuración de software, en este caso el Modelo de Procesos, se hace uso de la guía **ISO 9003[2**], el cual es una aplicación de las directrices planteadas dentro del estándar **ISO 9001[3],** estándar enfocado en precisar los requisitos que debería tener un sistema un correcto sistema de la gestión de la calidad dentro de su sistema productivo.

Asimismo, con base al estándar **IEEE 828-1983[4],** se hace uso de los requerimientos esenciales mínimos para la identificación de los elementos de productos de software, en este caso, este documento del Modelo de Procesos, así como también el manejo de la implementación de cambios, su registro, y reporte del estado de este, siendo su aplicación directa a lo largo de todo el ciclo de vida del software; para la aplicación de este se hace uso de la guía propuesta por el estándar **IEEE 1042-1987[5]** para la implementación del estándar anterior para la planeación de la gestión de actividades durante el desarrollo y mantenimiento en el ciclo de vida del software.

De esta forma, con los estándares anteriores y considerando la información mínima necesaria para una identificación correcta y efectiva dentro de la gestión de la configuración de software, se ha realizado las siguientes tablas de identificación de versiones de cambios para el escrito actual, donde se incluyen los puntos de identificación del documento, el historial de versiones junto con el estado de las versiones y un historial de cambios realizados.

|  |  |
| --- | --- |
| **Título del producto de software** | Modelo de Procesos |
| **Localización de referencia** | <https://github.com/Daniel-Pliego/SQA> |
| **Autor** | Daniel Pliego Vega |
| **Fecha de creación** | 05 de marzo de 2023 |

#### Tabla de Identificación del Elemento de la Configuración de Software

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versión del SCI** | **Fecha de elaboración** | **Estado de integración del cambio** | **Responsable del cambio** | **Identificador de la versión** |
| 2.0 | 07/03/2023 | EN REVISIÓN | Equipo | SQA\_Modelo\_Procesos\_V2.0 |
| 1.2 | 06/03/2023 | ACEPTADO | Equipo | SQA\_Modelo\_Procesos\_V1.2 |
| 1.1 | 06/03/2023 | ACEPTADO | Equipo | SQA\_Modelo\_Procesos\_V1.1 |
| 1.0 | 05/02/2023 | ACEPTADO | Equipo | SQA\_Modelo\_Procesos\_V1.0 |

#### Tabla de Historial de Versiones

#### Tabla de Historial de Cambios

|  |  |  |
| --- | --- | --- |
| **Versión del SCI** | **Fecha de la Solicitud de Cambios de Software** | **Descripción de cambios** |
| 2.0 | 07/03/2023 | Agregación de los documentos de control y el estándar para la realización de la fase de prueba; elaboración de versión base 2.0 del SCI actual. |
| 1.2 | 06/03/2023 | Especificación de estándares y plantillas dentro de la tabla de fases y actividades; Cambio de objetivos e indicadores. |
| 1.1 | 06/03/2023 | Secciones de Gestión de la configuración, Contenido e Índice de Estándares añadidos. |
| 1.0 | 07/02/2023 | Versión inicial de Modelo de procesos; estructuración base con respecto a MoProSoft |

## **Definición general del proceso**

### Propósito:

La organización se compromete a seguir el proceso de Desarrollo y Mantenimiento de Software descrito en este documento para garantizar la calidad de los productos desarrollados. Para ello, se realizarán las siguientes actividades de manera sistemática en cada proyecto: análisis, diseño, construcción, integración y pruebas del software cumpliendo con los requerimientos especificados.

### Descripción:

El proceso de Desarrollo y Mantenimiento de Software que la organización llevará a cabo se compone de al menos un ciclo de desarrollo, donde cada ciclo consta de diversas fases, las cuales se describirán a continuación:

* **Inicio**: Revisión y análisis detallado del Plan de Desarrollo para lograr el entendimiento común del proyecto, obteniendo compromiso y responsabilidad de los involucrados dentro del desarrollo de este.
* **Requerimientos**: Realización del conjunto de actividades cuya intención se es dirigida a la mejora del proceso de obtención del Documento de Especificación de Requisitos de Software de una forma pronta y de calidad, permitiendo un entendimiento efectivo entre el cliente y el proyecto.
* **Análisis y Diseño**: Conjunto de actividades dirigidas al análisis profundo de los requerimientos obtenidos durante la fase de Requerimientos especificados para la producción de una descripción detallada de la estructura de componentes de software, elemento necesario para la fase de construcción; obtención de la documentación del Análisis y Diseño, elementos que podrían ser utilizados en un posterior análisis, y el Plan de Pruebas del Sistema.
* **Construcción:** Conjunto de actividades de producción de componentes de software que se encuentren conformes a lo estipulado durante la etapa de Análisis y Diseño; realización de las pruebas unitarias, obteniendo los componentes de software comprobados.
* **Integración y Pruebas:** Actividades de integración y pruebas realizados a los componentes de software basados en los documentos de Planes de Pruebas de Integración y de Sistema, obteniendo software que satisfaga los requerimientos especificados en el Documento de Especificación de Requisitos.
* **Cierre:** Integración final de la Configuración de Software generado durante las fases previas, obteniendo la entrega parcial o total del producto de software; generación del Reporte de Mediciones y Sugerencias de Mejora

De manera adicional, dentro de cada una de las fases anteriormente descritas, se irán realizando las siguientes actividades con el fin de llevar un óptimo proceso de trabajo:

* Distribución de roles, responsabilidades y tareas específicas para cada miembro del equipo.
* Producción, verificación supervisada y validación de todos los productos, así como las correcciones pertinentes y adecuadas en caso de ser necesarias.
* Generación de Bitácora de Avance y Reporte de Actividades.

### Objetivos:

O1. Lograr los Objetivos del proyecto en tiempo y costo mediante la coordinación y el manejo de los recursos de este.

O2. Mantener informado al Cliente mediante la realización de reuniones de avance del proyecto.

O3. Atender las solicitudes de cambio del cliente mediante la recepción y análisis de estas.

### Indicadores:

* (O1) El Plan del Proyecto y el Plan de Desarrollo contemplan a los Objetivos establecidos en la Descripción del Proyecto y a las Metas Cuantitativas para el Proyecto.
* (O1) Las actividades del proyecto se realizan conforme a lo establecido en el Plan del Proyecto y en el Plan de Desarrollo.
* (O1) El tiempo y costo real están acordes con lo estimado.
* (O2) Las reuniones de avance del proyecto se realizan conforme a lo acordado con el Cliente.
* (O3) El mecanismo de recepción y análisis se aplica a todas Solicitudes de Cambios.

### Metas cuantitativas:

* El 80% de las actividades son terminadas antes o el mismo día que marca su periodo de finalización.
* El 100% de los artefactos generados son verificados y validados.
* El 75% de las reuniones que son agendas, son concretadas.

### Responsabilidad y autoridad:

* Se asignará un responsable de desarrollo y mantenimiento de software.
* Se asignará un responsable de Pruebas y aseguramiento de la calidad.
* Se asignará una autoridad para el cumplimiento de las tareas en el tiempo asignado y que se cumplan los requisitos solicitados.

### Entradas:

*Plan de Desarrollo*: Será necesario contar con un documento para la gestión de proyectos de software para describir el producto a desarrollar, los entregables esperados, el proceso de desarrollo y mantenimiento de software, la formación del equipo de trabajo y el calendario de actividades y fechas de entrega. El plan es fundamental para garantizar una planificación eficiente, una ejecución adecuada y una entrega de calidad del producto.

### Salidas:

*Especificación de Requerimientos*: Documento que incluirá la descripción de los requerimientos, que contará con el mismo formato según el mencionado por Karl E. Wiegers (1999); a grandes rasgos se mantiene un formato de introducción, descripción general, funcionalidades del sistema, requerimientos no funcionales y apéndices.

*Análisis y Diseño*: Documento que describe la estructura de los componentes de software con una descripción textual y gráfica. La estructura incluye la parte arquitectónica, que contiene la estructura interna del sistema y la identificación de los componentes y relaciones de interacción entre ellos; por otro lado, la parte detallada proporciona información detallada para la construcción y prueba en el ambiente de programación.

*Componente*: Hace referencia al conjunto de unidades de código relacionadas.

*Software*: Sistema de software con estructura compuesta por subsistemas y componentes, diseñado para cumplir con las necesidades del cliente o usuario.

*Configuración de Software*: Conjunto de productos de software en los que se incluye

* *Especificación de Requerimientos*
* *Análisis y Diseño*
* *Software*
* *Registro de Rastreo*
* *Plan de Pruebas de Sistema*
* *Reporte de Pruebas de Sistema*
* *Plan de Pruebas de Integración*
* *Reporte de Pruebas de Integración*
* *Manual de Usuario*
* *Manual de Operación*
* *Manual de Mantenimiento*

*Manual de Usuario*: Documento que describe el cómo hacer uso del sistema de software, basado en la interfaz de usuario; este documento deberá de estar redactado en términos fáciles de comprender para los usuarios.

*Manual de Operación*: Documento que explica cómo instalar y administrar el software y su entorno de funcionamiento, incluyendo el ambiente de operación; destinado a personal encargado de la operación y redactado de manera comprensible.

*Manual de Mantenimiento*: Documento de texto que explica la Configuración de Software, así como el ambiente usado para el desarrollo y pruebas; deberá incluir descripciones de los compiladores, herramientas de análisis y diseño, construcción y pruebas. El documento está destinado al personal de mantenimiento y será redactado respectivamente.

*Reporte de Actividades*: Calendario para el registro periódico de actividades, en donde se incluya las fechas de inicio y fin, responsables y mediciones adecuadas.

*Lecciones Aprendidas*: Registro de las mejores prácticas utilizadas, así como las experiencias exitosas dentro de la solución de problemas y los problemas recurrentes que fueron identificados. El documento deberá ser redactado con el objetivo de mejorar y agilizar los ciclos de desarrollo subsecuentes.

*Reporte de Mediciones y Sugerencias de Mejora*: Registro que incluya las mediciones de los indicadores anteriormente mencionados del proceso de Desarrollo y Mantenimiento del Software. También deberá incluir sugerencias de mejora del proceso, tales como métodos, herramientas, formatos, etc.

*Registro de Rastreo*: Relación entre los requerimientos, elementos de análisis y diseño, componentes y planes de pruebas.

*Plan de Pruebas de Sistem*a: Documento con las pruebas que fueron identificadas por el equipo para el cumplimiento de los requerimientos detallados establecidos en la fase de Requerimientos.

*Reporte de Pruebas de Sistema*: Documento con el registro de las pruebas realizadas en el sistema; deberá de registrar los defectos identificados, los miembros del equipo que participaron en su aplicación, fechas, lugar y duración.

*Plan de pruebas de Integración:* Se especifican las pruebas que se ejecutaran para verificar que los diferentes componentes resultantes de la fase de análisis y diseño se integran de manera adecuada

*Reporte de Pruebas de Integración: S*e registra los fallos encontrados junto con una descripción con los siguientes datos: lugar del fallo, condiciones para su replicación y los participantes del equipo que introdujeron el fallo.

*Reportes de Verificación:* Se registran los requerimientos trabajados y su correspondencia con funcionalidades visibles del producto o en su defecto, pruebas que avalen el cumplimiento.

*Reportes de Validación:* Se registran las observaciones de cliente respecto a las funcionalidades analizadas durante la reunión, así como el requerimiento con el que dicha funcionalidad tiene correspondencia.

### Documentos de control – Fase de pruebas:

De acuerdo al **ISO/IEC/IEEE 29119[6]** tenemos los siguientes documentos de control para el proceso de pruebas:

* *Plan de pruebas:* este documento describe la estrategia general de pruebas, incluyendo los objetivos de las pruebas, el enfoque de las pruebas, el alcance de las pruebas y los recursos necesarios para llevar a cabo las pruebas.
* *Diseño de casos de prueba:* este documento describe los casos de prueba individuales que se utilizarán para verificar que el software cumple con los requisitos especificados. Cada caso de prueba debe incluir información sobre el propósito de la prueba, los datos de entrada, los pasos de la prueba, los resultados esperados y los criterios de aceptación.
* *Especificación de requisitos de pruebas:* este documento describe los requisitos de pruebas específicos que deben ser cumplidos por el software. Esto incluye los requisitos funcionales, los requisitos no funcionales y cualquier otro requisito relevante para las pruebas.
* *Informe de pruebas:* este documento proporciona un resumen de los resultados de las pruebas. Incluye información sobre los casos de prueba ejecutados, los resultados obtenidos, los errores encontrados y las recomendaciones para futuras pruebas.
* *Registro de incidentes:* este documento se utiliza para registrar todos los errores encontrados durante las pruebas. Cada registro debe incluir información sobre el incidente, incluyendo la descripción del problema, la gravedad del problema y cualquier información adicional relevante.

### Roles:

|  |  |  |
| --- | --- | --- |
| **Rol** | **Abreviatura** | **Capacitación** |
| Responsable de Administración del Proyecto Específico | RAPE | Capacidad de liderazgo con experiencia en la toma de decisiones, planificación estratégica, manejo de personal y desarrollo de software. |
| Responsable de Desarrollo y Mantenimiento de Software | RDM | Conocimiento y experiencia en el desarrollo y mantenimiento de software. |
| Analista | AN | Conocimiento y experiencia en la obtención, especificación y análisis de los requerimientos. |
| Diseñador de Interfaz de Usuario | DU | Conocimiento en diseño de interfaces de usuario y criterios ergonómicos. |
| Diseñador | DI | Conocimiento y experiencia en el diseño de la estructura de los componentes de software. |
| Programador | PR | Conocimiento y/o experiencia en la programación, integración y pruebas unitarias. |
| Responsable de Pruebas | RPU | Conocimiento y experiencia en la planificación y realización de pruebas de integración y de sistema. |
| Revisor | RE | Conocimiento en las técnicas de revisión y experiencia en el desarrollo y mantenimiento de software. |
| Responsable de Manuales | RM | Conocimiento en las técnicas de redacción y experiencia en el desarrollo y mantenimiento de software. |
| Equipo de Trabajo | ET | Conocimiento y experiencia de acuerdo con su rol. |
| Cliente | CL | Interpretación del estándar de la especificación de requerimientos. |
| Usuario | US | Ninguna |

**Actividades:**

|  |  |
| --- | --- |
| **Rol** | **Descripción** |
| **A1. Fase de Inicio** | |
| ET | A1.1. Revisar con los miembros del equipo de trabajo el Plan de Desarrollo actual para lograr un entendimiento común y obtener su compromiso con el proyecto. |
| RDM | A1.2. Elaborar el Reporte de Actividades registrando las actividades realizadas, fechas de inicio y fin, responsable por actividad y mediciones requeridas |
| **A2. Fase de Requerimientos** | |
| RDM AN | A2.1. Distribuir tareas a los miembros del equipo de trabajo según su rol, de acuerdo con el Plan de Desarrollo actual. |
| AN  CL  US  DU | A2.2. Documentar o modificar la Especificación de Requerimientos.   * Identificar y consultar fuentes de información (clientes, usuarios, sistemas previos, documentos, etc.) para obtener nuevos requerimientos. * Analizar los requerimientos identificados para delimitar el alcance y su factibilidad, considerando las restricciones del ambiente del negocio del cliente o del proyecto. * Elaborar o modificar el prototipo de la interfaz con el usuario. * Generar o actualizar la Especificación de Requerimientos. |
| RE | A2.3. Verificar la Especificación de Requerimientos. |
| AN  DU | A2.4. Corregir los defectos encontrados en la Especificación de Requerimientos con base en el Reporte de Verificación y obtener la aprobación de las correcciones. |
| CL  US  RPU | A2.5. Validar la Especificación de Requerimientos. |
| AN  DU | A2.6. Corregir los defectos encontrados en la Especificación de Requerimientos con base en el Reporte de Validación y obtener la aprobación de las correcciones. |
| RPU  AN | A2.7. Elaborar o modificar Plan de Pruebas de Sistema con base a lo descrito en la plantilla **DI-IPSC-81438A[7.a]** del estándar **MIL-STD-498[7]**. |
| RE | A2.8. Verificar el Plan de Pruebas de Sistema con base en el estándar IEEE 1012. |
| RPU | A2.9. Corregir los defectos encontrados en el Plan de Pruebas de Sistema con base en el Reporte de Verificación y obtener la aprobación de las correcciones. |
| RM | A2.10. Documentar la versión preliminar del Manual de Usuario o modificar el manual existente. |
| RE | A2.11. Verificar el Manual de Usuario. |
| RM | A2.12. Corregir los defectos encontrados en el Manual de Usuario con base en el Reporte de Verificación y obtener la aprobación de las correcciones. |
| RDM | A2.13. Incorporar Especificación de Requerimientos, Plan de Pruebas de Sistema y Manual de Usuario como líneas base a la Configuración de Software. |
| RDM | A2.14. Elaborar el Reporte de Actividades registrando las actividades realizadas, fechas de inicio y fin, responsable por actividad y mediciones requeridas. |
| **A3. Fase de Análisis y Diseño** | |
| RDM  AN  DI | A3.1. Distribuir tareas a los miembros del equipo de trabajo según su rol, de acuerdo con el Plan de Desarrollo actual. |
| AN  DI  DU | A3.2. Documentar o modificar el Análisis y Diseño:   * Analizar la Especificación de Requerimientos para generar la descripción de la estructura interna del sistema y su descomposición en subsistemas, y éstos a su vez en componentes, definiendo las interfaces entre ellos. * Describir el detalle de la apariencia y el comportamiento de la interfaz con base en la Especificación de Requerimientos de forma que se puedan prever los recursos para su implementación. * Describir el detalle de los componentes que permita su construcción de manera evidente. * Generar o actualizar el Análisis y Diseño. * Generar o modificar el Registro de Rastreo. |
| RE | A3.3. Verificar el Análisis y Diseño y el Registro de Rastreo. |
| AN  DI  DU | A3.4. Corregir los defectos encontrados en el Análisis y Diseño y en el Registro de Rastreo con base en el Reporte de Verificación y obtener la aprobación de las correcciones. |
| CL  RPU | A3.5. Validar el Análisis y Diseño. |
| AN  DI  DU | A3.6. Corregir los defectos encontrados en el Análisis y Diseño con base en el Reporte de Validación y obtener la aprobación de las correcciones. |
| RPU | A3.7. Elaborar o modificar Plan de Pruebas de Integración con base a lo descrito en la plantilla **DI-IPSC-81438A[7.a]**. |
| RE | A3.8. Verificar el Plan de Pruebas de Integración. |
| RPU | A3.9. Corregir los defectos encontrados en el Plan de Pruebas de Integración con base en el Reporte de Verificación y obtener la aprobación de las correcciones |
| RDM | A3.10. Incorporar Análisis y Diseño, Registro de Rastreo y Plan de Pruebas de Integración como líneas base a la Configuración de Software. |
| RDM | A3.11. Elaborar el Reporte de Actividades registrando las actividades realizadas, fechas de inicio y fin, responsable por actividad y mediciones requeridas. |
| **A4. Fase de Construcción** | |
| RDM | A4.1. Distribuir tareas a los miembros del equipo de trabajo según su rol, de acuerdo con el Plan de Desarrollo actual. |
| PR | A4.2. Construir o modificar el(los) Componente(s) de software:   * Implementar o modificar Componente(s) con base a la parte detallada del Análisis y Diseño. * Definir y aplicar pruebas unitarias para verificar que el funcionamiento de cada componente esté acorde con la parte detallada del Análisis y Diseño. * Corregir los defectos encontrados hasta lograr pruebas unitarias exitosas (sin defectos). * Actualizar el Registro de Rastreo, incorporando los componentes construidos o modificados. |
| RE | A4.3. Verificar el Registro de Rastreo. |
| PR | A4.4. Corregir los defectos encontrados en el Registro de Rastreo con base en el Reporte de Verificación y obtener la aprobación de las correcciones. |
| RDM | A4.5. Incorporar Componentes y Registro de Rastreo como líneas base a la Configuración de Software. |
| RDM | A4.6. Elaborar el Reporte de Actividades, registrando las actividades realizadas, fechas de inicio y fin, responsable por actividad y mediciones requeridas. |
| **A5. Fase de Integración y Pruebas** | |
| RDM | A5.1. Distribuir las tareas entre el equipo de trabajo de acuerdo con el rol y al plan de desarrollo actual. |
| PR, RPU | A5.2. Inicio del proceso de integración y pruebas. El proceso de integración de software consiste en combinar los componentes individuales en sistemas o en el sistema completo del software. Se deben realizar pruebas siguiendo el plan previamente establecido, registrando los resultados en un informe de pruebas de integración. En caso de encontrar errores o defectos, es necesario corregirlos hasta obtener una prueba de integración exitosa sin fallas. Finalmente, se debe actualizar el registro de seguimiento para tener un control adecuado del progreso. |
| RM | A5.3. Crear o modificar el Manual de Operación con base a lo establecido en la plantilla **DI-IPSC-81444A[7.b]**. |
| RE | A5.4 Verificar el Manual de Operación. |
| RM | A5.5. Identificar y corregir los errores o incidencias presentes en el Manual de Operación. Una vez realizadas las correcciones, es necesario obtener la aprobación de estas. |
| RPU | A5.6. Realizar las pruebas de sistema siguiendo el Plan de Pruebas de Sistema, documentando los resultados en un Reporte de Pruebas de Sistema. Para la elaboración de este reporte, se utilizará la plantilla **DI-IPSC-81440A[7.c]**. |
| PR | A5.7 Corregir defectos encontrados en pruebas de sistema a partir del Reporte de Pruebas de Sistema, con base en el estándar **IEEE 1044-2009[8]** y obtener aprobación de las correcciones. |
| RM | A5.8. Crear o modificar el Manual de Usuario usando como base la plantilla **DI-IPSC-81443A[7.d]**. |
| RE | A5.9. Verificar el Manual de Usuario, con base en el estándar IEEE 1012. |
| RM | A5.10. Identificar y corregir los errores o incidencias presentes en el Manual de Usuario, con base en el estándar IEEE 1044. Una vez realizadas las correcciones, es necesario obtener la aprobación de estas. |
| RDM | A5.11. Incorporar el Software, Reporte de Pruebas de Integración, Registro de Rastreo, Manual de Operación y Manual de Usuario para ser utilizado como referencia para algún cambio en la Configuración de Software. |
| RDM | A5.12. Registrar las actividades efectuadas, las fechas en las que comenzaron y finalizaron, la persona responsable de cada tarea y las mediciones necesarias para evaluar el progreso y el éxito de las actividades. El registro de actividades será con base en el estándar **ANSI/IEEE 1058-1998 [9].** |
| **A6. Fase de Cierre** | |
| RM | A6.1. Crear o modificar el Manual de Mantenimiento. |
| RE | A6.2. Verificar Manual de Mantenimiento. |
| RM | A6.3. Identificar y corregir los defectos encontrados en el Manual de Mantenimiento. Una vez realizadas las correcciones, es necesario obtener la aprobación de estas. |
| RDM | A6.4. Integrar el Manual de Mantenimiento en la Configuración de Software. |
| RDM, ET | A6.5. El proceso de identificación Lecciones Aprendidas implica revisar los resultados y experiencias de un proyecto para identificar los aspectos que pueden mejorarse o que han sido exitosos. Estas lecciones incluyen mejores prácticas, éxitos en la gestión de riesgos, problemas recurrentes, entre otros aspectos relevantes. |
| RDM, ET | A6.6. Crear el Reporte de Mediciones y Sugerencias de Mejora. |
| RDM | A6.7. Realizar el Reporte de Actividades, registrar las actividades que se han llevado a cabo, incluyendo las fechas de inicio y finalización, la persona responsable de cada tarea y las mediciones requeridas para evaluar el progreso y el éxito de las actividades. |

### Verificaciones y validaciones:

|  |  |  |  |
| --- | --- | --- | --- |
| **Actividad** | **Producto** | **Rol** | **Descripción** |
| A2.3 | Especificación de Requerimientos | RE | Verificar la Especificación de Requerimientos, asegurarse de que estén claramente escritos y que se correspondan con la Descripción del Producto y con los estándares de documentación requeridos. Revisar y garantizar que sean completos y no tengan ambigüedades o contradicciones. Si se encuentran defectos, se deben documentar en un Reporte de Verificación. |
| A2.5 | Especificación de Requerimientos | Cl, US, RPU | Realizar una prueba de usabilidad de la interfaz del usuario para evaluar su eficacia, eficiencia y cumplimiento con la Especificación de Requerimientos. Si se encuentran defectos en la validación, se deben documentar en un Reporte de Validación. |
| A2.8 | Plan de Pruebas de Sistema | RE | Verificar el Plan de Pruebas con la Especificación de Requerimientos y con los estándares de documentación requeridos en el Proceso Específico. Si se encuentran desviaciones o defectos, se deben documentar en un Reporte de Verificación. |
| A2.11 | Manual de Usuario | RE | Verificar que el Manual de Usuario esté alineado con la Especificación de Requerimientos y cumpla con los estándares de documentación requeridos en el Proceso Específico. Si se encuentran desviaciones o defectos, se deben registrar en un Reporte de Verificación. |
| A3.3 | Análisis y Diseño, Registro de Rastreo | RE | Verificar la documentación del Análisis y Diseño en cuanto a claridad, viabilidad, consistencia con la Especificación de Requerimientos y cumplimiento con el estándar de documentación requerido. Revisar que el Registro de Rastreo esté actualizado y tenga las correspondencias adecuadas entre requerimientos y elementos de Análisis y Diseño. Documentar cualquier defecto en un Reporte de Verificación |
| A3.5 | Análisis y Diseño | CL, RPU | Validar que los requisitos del cliente estén correctamente interpretados y representados en el diseño del software. Si se encuentran desviaciones o defectos en el Análisis y Diseño, documentarlos en un Reporte de Validación. |
| A3.8 | Plan de Pruebas de Integración | RE | Verificar que el Plan de Pruebas de Integración esté en línea con la arquitectura del software y cumpla con los estándares de calidad requeridos. Si se encuentran desviaciones o defectos, documentarlos en un Reporte de Verificación. |
| A4.3 | Registro de Rastreo | RE | Verificar la integridad y coherencia del Registro de Rastreo, asegurándose de que estén establecidas las correspondencias correctas entre los componentes y los elementos del Análisis y Diseño. Los defectos identificados se deben registrar en un Reporte de Verificación. |
| A5.4 | Manual de Operación | RE | Realizar una revisión de la consistencia del Manual de Operación con el software y asegurarse de cumplir con el estándar de documentación requerido en el Proceso Específico. Cualquier defecto encontrado se registra en un Reporte de Verificación. |
| A5.9 | Manual de Usuario | RE | Evaluar la coherencia del Manual de Usuario con el sistema de Software y con los estándares de documentación requeridos en el Proceso Específico. Documentar cualquier defecto encontrado en un Reporte de Verificación. |
| A6.2 | Manual de Mantenimiento | RE | Verificar que el Manual de Mantenimiento está sincronizado con la Configuración de Software y cumple con los estándares de documentación requeridos en el Proceso Específico. Los defectos identificados deben ser registrados en un informe de verificación. |

### Incorporación a la Base de Conocimiento:

Una base de conocimientos es una herramienta que almacena información relevante sobre un tópico específico. Se compone de datos extraídos de los siguientes productos:

* *Especificación de los Requerimientos*
* *Análisis y Diseño*
* *Componentes*
* *Software*
* *Plan de Pruebas de Sistema*
* *Reporte de Pruebas de Sistema*
* *Plan de Pruebas de Integración*
* *Reporte de Pruebas de Integración*
* *Manual de Usuario*
* *Manual de Operación*
* *Manual de Mantenimiento*
* *Registro de Rastreo*
* *Reportes de Actividades*
* *Lecciones Aprendidas*
* *Reportes de Verificación*
* *Reportes de Validación*

**Recursos de Infraestructura:**

|  |  |
| --- | --- |
| **Actividad(es)** | **Recurso** |
| A1, A2, A3, A4, A5, A6 | Herramienta para documentación. |
| A2 | Herramientas para la Especificación de Requerimientos. |
| A3 | Herramientas para el Análisis y Diseño. |
| A4 | Herramientas para la construcción. |
| A4, A5 | Herramientas para la realización de pruebas. |

**Mediciones:**

Al finalizar cada fase del proceso de desarrollo, se evalúa el progreso a través de un informe de rendimiento de los indicadores relevantes en relación con las metas establecidas. Para lograr una evaluación precisa, se recomiendan las siguientes medidas:

1. **M1**: Revisión de los Reportes de Verificación y Validación, así como la documentación generada con la herramienta utilizada para la especificación de requerimientos, análisis y diseño, construcción, y pruebas. Esto permite confirmar que se han realizado correctamente estas actividades y que se han corregido cualquier defecto identificado.
2. **M2:** Verificación de la Configuración de Software, asegurándose de que los productos utilizados durante el ciclo actual son los mismos que se identificaron previamente.
3. **M3:** Comparación entre el Plan de Desarrollo actual y el Reporte de Actividades correspondiente para evaluar el desempeño contra lo planificado. Esto permite conocer cualquier desviación y tomar medidas para corregirlas.

**Capacitación:**

El RDM debe asegurarse de proporcionar los recursos necesarios para que los colaboradores involucrados en el proceso de Desarrollo y Mantenimiento de Software participen en las actividades relacionadas con el programa actual de Capacitación de la Base de Conocimientos. Esto incluye facilitar el acceso a los materiales de formación, así como asegurarse de que los participantes tengan el tiempo necesario para asistir a las sesiones de entrenamiento. Al fomentar la participación en la capacitación, se mejorará la calidad del software y se aumentará la eficiencia del equipo de desarrollo.

**Situaciones excepcionales:**

El RDM es responsable de garantizar que el proceso de Desarrollo y Mantenimiento de Software se desarrolle sin interrupciones. Para ello, debe estar al tanto de cualquier situación que pueda impedir el cumplimiento de las actividades asignadas a los miembros del equipo.

Es importante que los roles involucrados en este proceso informen de manera oportuna al RDM sobre cualquier obstáculo o imposibilidad para llevar a cabo sus tareas. De esta manera, el RDM podrá trabajar en la resolución de estas situaciones y garantizar la continuidad del proceso.

Si el RDM no es capaz de solucionar un problema o si éste excede su ámbito de competencia, será necesario que se lo informe al RAPE para que se encargue de dar una solución adecuada.

**Lecciones aprendidas:**

Los miembros involucrados en el Proceso de Desarrollo y Mantenimiento de software deberán de consultar la Base de Conocimientos antes de comenzar sus tareas asignadas. Esto permitirá que se aproveche la experiencia previa de la organización. Al revisar las Lecciones Aprendidas, los miembros del equipo podrán tomar medidas preventivas para evitar problemas similares en el futuro.

### Guías de ajuste

Requerimientos: *Especificación de Requerimientos:* Dentro de la Especificación de Requerimientos se pueden incluir interfaces de usuario o prototipos sencillos que no tengan una funcionalidad mientras permitan un entendimiento más claro para la documentación de los requerimientos.

Requerimientos: *Manual de Usuario:* Dentro de la fase de Requerimientos se puede omitir la elaboración y verificación de un Manual de Usuario, siendo que este producto no sea primordial para esta fase; esta actividad deberá realizarse y completarse cuando mucho antes de terminar la fase de verificación, esto es, antes de terminar con la fase de integración y pruebas.

Requerimientos: *Plan de Pruebas del Sistema:* En caso de que sea aceptado y acordado con el cliente, el Plan de Pruebas del Sistema puede ser validado con el cliente.

Análisis y Diseño*: Análisis y Diseño:* En caso de que sea aceptado y acordado con el cliente, se puede hacer la omisión de la validación del Análisis y Diseño; esta actividad deberá realizarse y completarse cuando mucho antes de terminar la fase de verificación, esto es, antes de terminar con la fase de integración y pruebas.

Construcción*: Revisión entre compañeros del código:* Con el fin de obtener un producto con menos defectos de las fases anteriores, antes de realizar pruebas unitarias se puede hacer la inclusión de revisiones entre compañeros de con el objetivo de verificar el código de los componentes con respecto al *Análisis y Diseño;* dentro de algunos beneficios de esta actividad se encuentra la reducción del tiempo de corrección.

Construcción*: Pruebas unitarias:* Para el uso de pruebas unitarias se puede hacer uso del estándar IEEE Std 1008-1987 (R 1993) for Software Unit Testing para una definición más sistemática en la documentación.

Construcción*: Prototipo de interfaz:* La elaboración o modificación del prototipo de una interfaz, en caso de tenerla, para la realización de pruebas con un usuario es una actividad que puede ser añadida para la identificación de defectos críticos de uso.

*Reporte de Actividades:* Las mediciones requeridas en el Reporte de Actividades pueden ser modificadas de acuerdo con las necesidades de la organización o del proyecto.

## **Bibliografía**

IEEE Guide for Software Requirements Specifications. (1984). *IEEE Org*. <https://doi.org/10.1109/ieeestd.1984.119205>

Oktaba, H., & Vázquez, A. (2005). MoProSoft®. *Software Process Improvement for Small and Medium Enterprises*, 170-176. <https://doi.org/10.4018/978-1-59904-906-9.ch008>

IEEE. (1986). IEEE Standard for Software Unit Testing. *ANSI/IEEE Std 1008-1987*. <https://doi.org/10.1109/ieeestd.1986.81001>

"IEEE Standard for Software User Documentation," in IEEE Std 1063-2001, vol., no., pp.1-24, 20 Dec. 2001, doi: 10.1109/IEEESTD.2001.93368.

"IEEE Standard for Software Configuration Management Plans," in IEEE Std 828-1983, vol., no., pp.1-10, 24 June 1983, doi: 10.1109/IEEESTD.1983.7439689.

"IEEE Guide to Software Configuration Management," in ANSI/IEEE Std 1042-1987, vol., no., pp.1-93, 12 Sept. 1988, doi: 10.1109/IEEESTD.1988.94582.

Online Browsing Platform. (2015). *ISO 9001:2015(es) Sistemas de gestión de la calidad*. Recuperado 5 de marzo de 2023, de <https://www.iso.org/obp/ui/#iso:std:iso:9001:ed-5:v1:es>

Bamford, R., & Deibler, J. (1999). Configuration Management and ISO 9001. *Methods & Tools*, *7*(2), 2-8. <https://www.methodsandtools.com/mt/download.html>

"IEEE Standard Classification for Software Anomalies," in IEEE Std 1044-2009 (Revision of IEEE Std 1044-1993) , vol., no., pp.1-23, 7 Jan. 2010, doi: 10.1109/IEEESTD.2010.5399061.

"IEEE Standard for Software Project Management Plans," in IEEE Std 1058-1998 , vol., no., pp.1-28, 22 Dec. 1998, doi: 10.1109/IEEESTD.1998.88822.

"ISO/IEC/IEEE International Standard - Software and systems engineering --Software testing --Part 1:General concepts," in ISO/IEC/IEEE 29119-1:2022(E) , vol., no., pp.1-60, 27 Jan. 2022, doi: 10.1109/IEEESTD.2022.9698145.

"ISO/IEC/IEEE International Standard - Software and systems engineering - Software testing -- Part 2: Test processes," in ISO/IEC/IEEE 29119-2:2021(E) , vol., no., pp.1-64, 28 Oct. 2021, doi: 10.1109/IEEESTD.2021.9591508.

"IEEE/ISO/IEC International Standard for Software and systems engineering--Software testing--Part 3:Test documentation," in ISO/IEC/IEEE 29119-3:2021(E) , vol., no., pp.1-98, 28 Oct. 2021, doi: 10.1109/IEEESTD.2021.9591577.

"IEEE/ISO/IEC International Standard - Software and systems engineering--Software testing--Part 4: Test techniques," in ISO/IEC/IEEE 29119-4:2021(E) , vol., no., pp.1-148, 28 Oct. 2021, doi: 10.1109/IEEESTD.2021.9591574.

"ISO/IEC/IEEE International Standard - Software and systems engineering -- Software testing -- Part 5: Keyword-Driven Testing," in ISO/IEC/IEEE 29119-5 First edition 2016-11-15 , vol., no., pp.1-69, 15 Nov. 2016, doi: 10.1109/IEEESTD.2016.7750539.

## **Índice de Estándares**

1. **IEEE 1063-2001, Estándar para la Documentación de Software para Usuarios:** Estándar que define los requerimientos mínimos para la estructura, información contenida, y el formato de la documentación para usuarios, tanto para documentos digitales como impresos usados dentro del entorno de trabajo en donde se encuentren usuarios de un sistema que contenga software.
2. **ISO 9003, Guía para la Aplicación de ISO 9001 para el Desarrollo, Implementación y Mantenimiento de Software:** Guía para el control de calidad aplicado dentro de tiempo de vida de un sistema que haga uso de software, siendo aplicado dentro de todas las fases de la producción de software hablando desde los productos realizados de manera temprana durante el diseño y desarrollo y la proliferación de los productos de software como tal.
3. **ISO 9001, Estándar para el Sistema de gestión de la Calidad – Requisitos:** Estándar enfocado en establecer los requerimientos que deben ser cumplidos por una organización para un correcto sistema de la gestión de la calidad dentro de su sistema productivo; se describen las actividades mínimas que deben de estar presentes en una compañía u organización para satisfacer los requerimientos de un cliente en la producción consistente de productos.
4. **IEEE 828-1983, Estándar para los Planes de la Gestión de Configuración de Software:** Estándar que provee los requerimientos mínimos para la preparación y el contenido a utilizar dentro de los planes de la gestión de configuración de software, incluyendo métodos utilizados para la identificación de elementos producto de software, el control e implementación de cambios y el registro del estado de la implementación de cambios.
5. **IEEE 1042-1987, Guía para la Gestión de la Configuración de Software:** Guía que provee factores a considerar, sugerencias y ejemplos detallados, al igual que una interpretación de la manera en la que el estándar IEEE 828-1983 puede ser utilizado para la planeación de diferentes actividades de desarrollo y mantenimiento para un programa de computadora
6. **ISO/IEC/IEEE 29119, Estándares para Ingeniería de Software y Sistemas – Pruebas de Software:** Conjunto de estándares internacionales para la pruebas de software, siendo esta la que defina los procesos, documentación, técnicas y procesos para un modelo de aseguramiento que pueda ser utilizado dentro de cualquier ciclo de vida de desarrollo de un software; se encuentra conformado por cinco secciones:
   1. **ISO/IEC/IEEE 29119-1:2022, Parte 1: Conceptos Generales:** Especifica conceptos y definiciones clave para la proceso de pruebas
   2. **ISO/IEC/IEEE 29119-2:2021, Parte 2: Procesos de Pruebas:** Especifica los procesos de prueba que pueden utilizarse para gobernar, gestionar y aplicar pruebas de software para cualquier organización, proyecto o actividad de prueba. Incluye descripciones genéricas de procesos de prueba que definen los procesos de prueba de software. También se proporcionan diagramas informativos de apoyo que describen los procesos.
   3. **ISO/IEC/IEEE 29119-3:2021, Parte 3: Documentación de Pruebas**: Especifica plantillas de documentación de pruebas de software que pueden utilizarse en cualquier organización, proyecto o actividad de pruebas. Describe la documentación de prueba resultado de los procesos especificados en ISO/IEC/IEEE 29119-2.
   4. **ISO/IEC/IEEE 29119-4:2021, Parte 4: Técnicas de Pruebas:** Define las técnicas de diseño de pruebas que pueden utilizarse durante el proceso de diseño e implementación de pruebas que se define en ISO/IEC/IEEE 29119-2.
   5. **ISO/IEC/IEEE 29119-5:2016, Parte 5: Pruebas Basadas en Palabras Clave:** define una solución eficaz y consistente para el *Keyword-Driven Testing.*
7. **MIL-STD-498, Estándar militar de los Estados Unidos para el desarrollo de software y sistemas de información:** Proporciona un marco para el desarrollo de sistemas de información y software para el Departamento de Defensa de los Estados Unidos. Define los procesos y procedimientos necesarios para desarrollar, documentar y mantener sistemas de software.
   1. **DI-IPSC-81438A, Plan de Pruebas de Software (STP):** Describe el entorno de prueba que se utilizará para las pruebas, identifica las pruebas que se realizarán y proporciona calendarios para las actividades de prueba.
   2. **DI-IPSC-81444A, Manual del Operador de Software (SCOM):** El SCOM proporciona al personal de un centro informático o de otra instalación de software centralizada o en red información sobre cómo instalar y utilizar un sistema de software.
   3. **DI-IPSC-81440A, Reporte de Pruebas de Software (STR):** STR es un registro de las pruebas de cualificación realizadas en un elemento de configuración de software informático (CSCI), un sistema o subsistema de software u otro elemento relacionado con el software. El STR permite a la entidad adquirente evaluar las pruebas y sus resultados.
   4. **DI-IPSC-81443A, Manual de Usuario (SUM):** explica a un usuario práctico de software cómo instalar y utilizar un Elemento de Configuración de Software Informático (CSCI), un grupo de CSCI relacionados o un sistema o subsistema de software.
8. **IEEE 1044-2009, Estándar de Clasificación para Anomalías de Software:** Incluye orientaciones sobre la resolución de defectos y define 4 actividades, dentro de las cuales están:
   1. Notificación de defectos: Consiste en informar de los defectos al equipo o persona responsable. El informe debe incluir una descripción clara del defecto y los pasos para reproducirlo.
   2. Análisis del defecto: Consiste en analizar el defecto notificado para determinar su causa e impacto
   3. Corrección del defecto: Consiste en desarrollar una solución para el defecto y verificar que la solución ha resuelto el problema
   4. Volver a probar: Consiste en probar el producto de software para garantizar que la corrección del defecto no ha introducido nuevos problemas y que el producto funciona como se esperaba. También hace hincapié en la importancia de documentar el proceso de resolución de defectos y mantener registros de los defectos y sus correcciones.
9. **IEEE 1058-1998, Estándar para Planes de Gestión de Proyectos**: Este estándar actúa como una guía completa que proporciona orientación sobre la gestión de proyectos de software, incluyendo la planificación, el seguimiento y el control del proyecto. Además, proporciona directrices para la gestión de riesgos, la estimación de costos, la programación y la documentación del proyecto, por lo que es utilizado ampliamente en la industria para asegurar la calidad y el éxito de los proyectos de software.