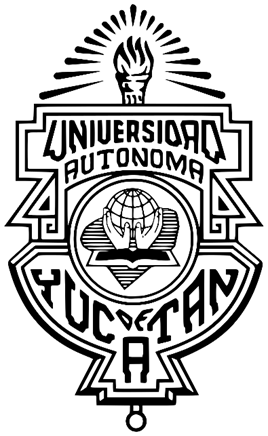
CALIFICACIÓN FINAL: 44

1. Propósito 3/7 pts.
2. Documentos referenciados 2/3 pts.
3. Gestión 5/15 pts.
4. Documentación 5/15 pts.
5. Estándares, prácticas, convenciones y métricas 3/10 pts.
6. Revisiones y auditorías 5/15 pts.
7. Pruebas 3/5pts.
8. Reporte de pruebas y acciones correctivas 4/5 pts.
9. Herramientas, técnicas y metodologías. 5/5 pts.
10. Control de código 0/5 pts.
11. Control de medios 2/5 pts.
12. Recolección de registros, mantenimiento y resguardo 2/5 pts.
13. Entrenamiento 5/5 pts.

Plan de SQA



Alumnos

Álvarez Vázquez Jesús Miguel

Avila Pacheco David de Jesús

Gonzalez Herrera Miguel Humberto

Echeverria Leon Eduardo Leonel

Chan Zurita Mario Jesús

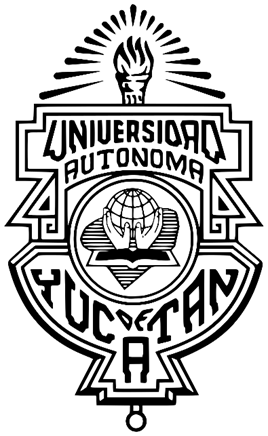
Edwin de Jesús León Bojórquez

Universidad Autónoma de Yucatán

Facultad de Matemáticas

Licenciatura en Ingeniería de Software

Aseguramiento de la Calidad de Software



CopyPaste

Plan de aseguramiento de la calidad

Aseguramiento de la calidad

Documento del Plan de SQA

Versión 1.2

**Autores**

Álvarez Vázquez Jesús Miguel

Avila Pacheco David de Jesús

Gonzalez Herrera Miguel Humberto

Echeverria Leon Eduardo Leonel

Chan Zurita Mario Jesús

**Fecha de finalización**

15 de mayo de 2022

**Persona(s) que aprobaron el documento**

Comité de revisión de diseño formal (FDR):

Álvarez Vázquez Jesús Miguel

Avila Pacheco David de Jesús

Gonzalez Herrera Miguel Humberto

Echeverria Leon Eduardo Leonel

Chan Zurita Mario Jesús

**Fecha de aprobación**

15 de mayo de 2022

**Firma(s) del auto(es) y persona(s) que lo aprobaron**

Álvarez Vázquez Jesús Miguel

Avila Pacheco David de Jesús

Gonzalez Herrera Miguel Humberto

Echeverria Leon Eduardo Leonel

Chan Zurita Mario Jesús

**Lista de versiones y revisiones anteriores**

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Versión** | **Autor** |
| 05/05/2022 | 1.0 | Equipo de desarrollo |
| 12/05/2022 | 1.1 | Equipo de desarrollo |
| 15/05/2022 | 1.2 | Equipo de desarrollo |

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** |
| 05/05/2022 | 1.0 | Comité de revisión de diseño formal (FDR) |
| 12/05/2022 | 1.1 | Comité de revisión de diseño formal (FDR) |
| 15/05/2022 | 1.2 | Comité de revisión de diseño formal (FDR) |

**Lista de circulación**

El documento será subido a un repositorio, en el cual solo los miembros de este podrán obtenerlo y visualizarlo (requiriendo del correo electrónico de cada miembro para dicho procedimiento).

**Restricciones de confidencialidad**

No se puede tomar fotografías, capturas y realizar copias del documento.

Solamente el personal autorizado puede realizar cambios en el documento y con la condición de contar con la autorización aprobada para hacerlo.

**Índice**

[**Propósito**](#_xwh7r8izddsr) **6**

[**Documentos referenciados**](#_cb7edk79vmnn) **6**

[**Gestión**](#_vdi9dmgcvbtg) **6**

[Organización](#_fvuatgg00tuq) 6

[Actividades](#_1fuzuy2lkrlj) 7

[Responsabilidades](#_42whbdprjkcc) 10

[**Documentación**](#_4j8qouhmtfk9) **11**

[Propósito](#_w0rvwab4ohp) 11

[Documentación mínima requerida](#_ii71q2cqluj0) 11

[**Estándares, prácticas, convenciones y métricas.**](#_62sdxvca063q) **11**

[Estándares](#_wr63rpqy7nxc) 11

[Métricas](#_33cgrunoogdg) 12

[**Revisiones y auditorías**](#_kw44lu4jbkys) **12**

[Objetivo](#_57nsfjz03dxh) 12

[Requerimientos mínimos](#_sanjqsq6eg9u) 12

[Revisión de los Requisitos de Software (RRS):](#_afhz4vcwifyk) 12

[Revisión del Diseño Preliminar (PDR)](#_2efkgz538ldm) 13

[Revisión del Diseño Crítico (CDR)](#_io70dmvqo08o) 13

[Revisión Técnica Formal (RTF)](#_48mgshs8zap8) 13

[Revisiones a realizar](#_svhdufbprtpe) 14

[**Pruebas**](#_kc6kkd2zct0r) **16**

[**Informe de problemas y acción correctiva**](#_6xl11lgmvafl) **16**

[Reporte del proceso de auditorías](#_z1l0b4hes1nd) 16

[Presentación del reporte del proceso auditoría](#_yj5o38nmswxs) 16

[Procedimiento de escalamiento para la Resolución de no ocurrencia en el Proceso de Reporte de Auditoría](#_jc0suat1jfe5) 17

[**Técnicas, herramientas y metodologías**](#_yke8qrxefchl) **17**

[Herramientas](#_3c97r26cq14w) 17

[Técnicas](#_sfcdnjxiseaq) 18

[Metodologías](#_hnxubq7kke7u) 18

[**Control del código**](#_n61qf2wt320y) **18**

[**Control de medios**](#_6jsn31ksocnr) **20**

[Medio de almacenamiento](#_73m6u1jz30me) 20

[Proceso de copias de seguridad](#_8e4m8lni7uyd) 20

[Puntos de control](#_ividuwlglo1i) 20

[**Recopilación, mantenimiento y retención de registros**](#_zdnrqi1fhhsk) **20**

[**Entrenamiento**](#_1dmpn5dwh5o8) **21**

[**Anexos**](#_mk6x27q9shsh) **23**

# 

# Propósito

El objetivo de este plan es establecer las actividades que deben desarrollarse para garantizar la calidad del producto. Se indicará para cada actividad los atributos de calidad relevantes, los métodos de evaluación y los responsables.

Garantizando el cumplimiento de los estándares y de las metodologías, se brindarán los elementos de apoyo a la gestión del proyecto para realizar verificaciones sobre la adecuación al proceso y así detectar desvíos que puedan resultar en acciones correctivas en etapas tempranas.

El alcance del plan abarca las fases del ciclo de vida relacionadas con: elaboración, construcción, evaluación y despliegue. No estando contempladas, por salirse del alcance del proyecto, las etapas relacionadas con el mantenimiento del producto.

# Documentos referenciados

[B1] IEEE Std 7301-1995, IEEE Guide for Software Quality Assurance Planning.

[B2] IEEE Std 730-2002, IEEE Software Quality Assurance Plans.

[B3] IEEE Std 730-2014, IEEE Software Quality Assurance Processes.

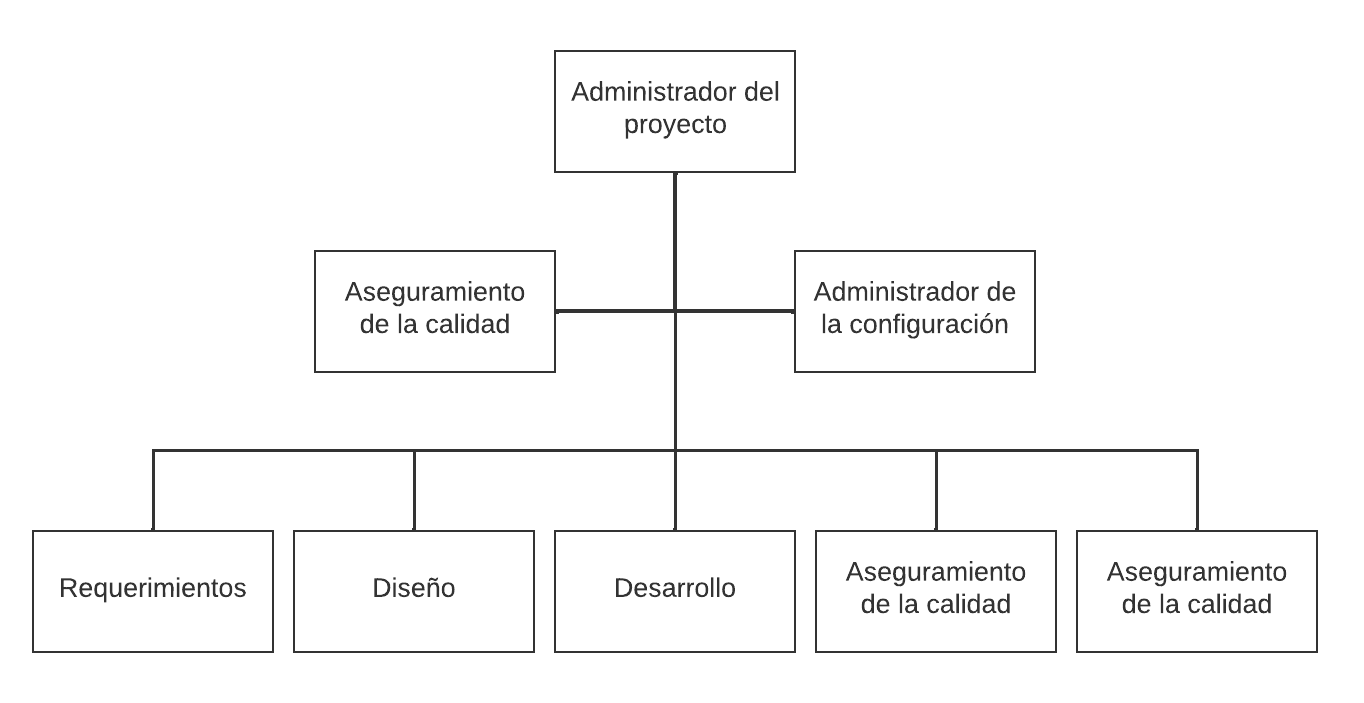
# Gestión

La gestión del proyecto está a cargo del administrador del proyecto, sin embargo será monitoreada tanto por este, como por el responsable de SQA. Se intenta controlar que las actividades se ajusten al plan propuesto y minimizar posibles desviaciones

## Organización

La línea directa de trabajo seguida se basa en las 5 fases del desarrollo de software:

* Análisis.
* Diseño.
* Implementación.
* Verificación y pruebas.
* Mantenimiento.



La estructura del equipo de trabajo será de la siguiente manera:

|  |  |
| --- | --- |
| Rol | Responsable |
| Responsable de SQA - Responsable de Verificación. | Jesús Álvarez |
| Asistente de SQA - Asistente de Verificación. | David Ávila |
| Diseñador de interfaz - Analista. | Mario Chan |
| Administrador del proyecto - Analista | Eduardo Echeverría |
| Especialista Técnico - Implementador | Miguel Gonzalez |

*Tabla 1. Roles del equipo*

## Actividades

El Plan de SQA abarca las etapas de: Requerimientos y Análisis, Diseño, Implementación y Verificación. Para cada una de las etapas se revisará la calidad de cada uno de los siguientes productos (adicionalmente se incluyen las de Gestión), las cuales:

|  |  |
| --- | --- |
| **Etapa** | **Productos a revisar la calidad** |
| Requerimientos y Análisis | * Especificación de requerimientos. * Modelos de casos de uso. * Alcance del sistema. * Pautas para la interfaz del usuario. |
| Diseño | * Modelo de dominio. * Descripción de la arquitectura. |
| Implementación y verificación | * Informe de verificación unitaria. * Plan de verificación. * Plan de validación. * Plan de implementación |
| Gestión del Proyecto | * Plan del proyecto. * Plan de iteración. * Gestión de riesgos. |
| Gestión de la Configuración y Control de Cambios | * Plan de Configuración de SCM * Informe de la Línea Base del Proyecto |

*Tabla 2. Etapas y productos a revisar la calidad*

Las tareas a ser llevadas a cabo deberán reflejar las evaluaciones a realizar, los estándares a seguir, los productos a revisar, los procedimientos a seguir en la elaboración de los distintos productos y los procedimientos para informar de los defectos detectados a sus responsables y realizar el seguimiento de los mismos hasta su corrección.

Las actividades que se realizarán son:

• Revisar cada producto

• Revisar el ajuste al proceso

• Realizar Revisión Técnica Formal (RTF)

• Asegurar que las desviaciones son documentadas

Las actividades quedan acorde a la siguiente tabla:

|  |  |  |
| --- | --- | --- |
| **Actividad** | **Descripción** | **Entregable** |
| Revisar cada producto | En esta actividad se revisan los productos que se definieron como claves para verificar en el  Plan de calidad.  Se debe verificar que no queden correcciones sin resolver en los informes de revisión previos,  si se encuentra alguna no resuelta, debe ser incluida en la siguiente revisión. Se revisan los  productos contra los estándares, utilizando la checklist definida para el producto.  Se debe identificar, documentar y seguir la pista a las desviaciones encontradas y verificar  que se hayan realizado las correcciones. | Informe de revisión de SQA |
| Revisar el ajuste al proceso | En esta actividad se revisan los productos que se definieron como claves para verificar el cumplimiento de las actividades definidas en el proceso. Con el fin de asegurar la calidad en el producto final del desarrollo, se deben llevar a cabo revisiones sobre los productos durante todo el ciclo de vida del software. Se debe recoger la información necesaria de cada producto, buscando hacia atrás los productos previos que deberían haberse generado, para poder establecer los criterios de revisión y evaluar si el producto cumple con las especificaciones. Esta información se obtiene de los siguientes documentos: Plan del Proyecto, Plan de iteración, Plan de Verificación. Antes de comenzar, se debe verificar en los informes de revisión previos que todas las desviaciones fueron corregidas, si no es así, las faltantes se incluyen para ser evaluadas. | Informe de revisión de SQA correspondiente a la evaluación de ajuste al Proceso |
| Realizar Revisión Técnica Formal (RTF) | El objetivo de la RTF es descubrir errores en la función, la lógica ó la implementación de cualquier producto del software, verificar que satisface sus especificaciones, que se ajusta a 5 los estándares establecidos, señalando las posibles desviaciones detectadas. Es un proceso de revisión riguroso, su objetivo es llegar a detectar lo antes posible, los posibles defectos o desviaciones en los productos que se van generando a lo largo del desarrollo. Por esta característica se adopta esta práctica para productos que son de especial importancia. En la reunión participan el responsable de SQA e integrantes del equipo de desarrollo. Se debe convocar a la reunión formalmente a los involucrados, informar del material que ellos deben preparar por adelantado, llevar una lista de preguntas y dudas que surgen del estudio del producto a ser revisado. La duración de la reunión no debe ser mayor a dos horas. | Informe de RTF |
| Asegurar que las desviaciones son documentadas | Las desviaciones encontradas en las actividades y en los productos deben ser documentadas y ser manejadas de acuerdo a un procedimiento establecido. Se debe chequear que los responsables de cada plan los modifiquen cada vez que sea necesario, basados en las desviaciones encontradas. | Documentación de las desviaciones |

*Tabla 3. Actividades a realizar para el plan de SQA*

## Responsabilidades

|  |  |
| --- | --- |
| Rol | Actividad |
| Administrador | * Plan de Proyecto * Gestión de Riesgos * Plan de Iteración |
| Analista | * Modelos de Casos de Uso * Alcance del Sistema * Pautas para la interfaz de usuario |
| Arquitecto | * Descripción de la arquitectura |
| Responsable de la Gestión de la Configuración | * Plan de Gestión de la Configuración * Informa de la Línea Base del Proyecto. |
| Responsable de Verificación y Validación | * Informe de Verificación Unitaria. * Plan de Verificación y Validación. |
| Responsable de SQA | * Control de calidad |
| Asistente de SQA | * Control de calidad |

*Tabla 4. Tabla de roles con sus respectivas actividades*

# 

# Documentación

## ~~Propósito~~

~~Identificación de la documentación relativa a desarrollo, Verificación & Validación, uso y mantenimiento del software.~~

~~Establecer como los documentos van a ser revisados para chequear consistencia: se confirman criterio e identificación de las revisiones.~~

## ~~Documentación mínima requerida~~

|  |  |  |
| --- | --- | --- |
| Documento | Entregable | Descripción |
| Especificación de requerimientos de software | ERS | Describe los requisitos del software: funcionales, atributos, y restricciones |
| Plan de aseguramiento de calidad | SQAP | Describe el plan, roles, que adoptará cada integrantes en el desarrollo del software |
| Plan de pruebas | PT | Descripción de los módulos que sera probados, el tipo de pruebas, entradas, salidas esperadas, aplicado a cada prueba a realizar |
| Administración de la configuración | MC | Describe la nomenclatura utilizada en el proyecto. |
| Plan de desarrollo de Software | PDS | Describe la implementación, calendarios, actividades, roles y su responsabilidades. |

*Tabla 5. Documentación mínima requerida*

# Estándares, prácticas, convenciones y métri*cas.*

## Estándares

Como estándar para la construcción del documento se aplica el estándar:

Std 1012-1986 IEEE Standard for Software Verification and Validation Plans

## Métricas

* Tiempo estimado
* Tiempo real invertido
* Esfuerzo planeado
* Esfuerzo realizado
* Costo planeado
* Costo real
* Número de inclumplimeitnos sin arreglar
* Número de inclumplimientos arreglados
* Número total de incumplimientos

# Revisiones y auditorías

## Objetivo

Definir las revisiones y auditorías técnicas y de gestión que se realizarán; asimismo, se especifica el cómo serán llevadas a cabo dichas revisiones y auditorías. Las revisiones y auditorías de los resultados del desarrollo se realizan a medida que se terminan cada una de las fases del ciclo de vida de desarrollo de software, con el fin de:

* Conocer el progreso alcanzado en el desarrollo.
* Evaluar el ajuste a los requerimientos del sistema.
* Evaluar la eficiencia en el trabajo

## Requerimientos mínimos

Como mínimo deberán revisarse todas las entregas semanales, basado en los estándares definidos anteriormente. Estas revisiones serán realizadas por el Responsable de SQA y/o el Asistente de SQA.

Se deben llevar a cabo, al menos, las siguientes revisiones y auditorías:

## Revisión de los Requisitos de Software (RRS):

La RRS se genera para:

* Evaluar las especificaciones de requerimientos del software (ERS).
* Asegurar que los requerimientos establecidos en la ERS, sean los correctos y estén completos.
* Garantizar la calidad, viabilidad e integridad de los requerimientos establecidos.

Los requerimientos de revisiones de ERS en la RRS son los siguientes:

* Fiable
* Completo
* Depurable
* Modificable
* .Consistente
* Libre de ambigüedades
* Utilizable durante la fase de operación y mantenimiento
* Inspeccionar que la relación entre los requerimientos y sus derivados sea la adecuada

## Revisión del Diseño Preliminar (PDR)

La PDR es realizada para evaluar la suficiencia técnica del DDS preliminar, antes de comenzar con el diseño detallado, define los siguientes puntos:

* Evaluar el progreso, consistencia y suficiencia técnica del alcance de diseño con los requerimientos funcionales de la ERS.
* Verificar la existencia y compatibilidad de las interfaces entre el software, el hardware y los usuarios finales.
* Determinar un diseño de software que cumpla con los requerimientos.

Para la PDR se toman como requerimientos de revisiones los siguientes puntos:

* Revisar que se detallen todas las interfaces con otro software, sistemas de comunicación, etc. Para una adecuada identificación de interfaces y de un diseño óptimo.
* Revisar que exista un análisis del diseño para verificar la compatibilidad con los requerimientos críticos.
* Revisar que se establezcan los requerimientos del factor humano.

## Revisión del Diseño Crítico (CDR)

La CDR es generada para determinar la aceptabilidad de cómo la DDS cumple con la ERS. Evalúa la suficiencia técnica, integridad del diseño detallado del software, antes de comenzar a codificar para establecer que el diseño detallado satisface los requerimientos de la SRS. Para la CDR se toman como requerimientos de revisión los siguientes puntos:

* Evaluar la compatibilidad del diseño detallado con la ERS.
* Examinar la representación de datos en forma de diagramas lógicos, algoritmos, almacenamiento y representación de datos.
* Determinar la compatibilidad e integridad de requerimientos de interfaces.
* Establecer que todas las interfaces internas y externas incluyendo interacciones con la base de datos sean expresadas.

## Revisión Técnica Formal (RTF)

Tiene el objetivo de descubrir errores en la función, la lógica ó la implementación de cualquier producto del software, verificar que satisface sus especificaciones, que se ajusta a los estándares establecidos, señalando las posibles desviaciones detectadas. Dicha revisión es un proceso de revisión riguroso, su objetivo es llegar a detectar lo antes posible, los posibles defectos o desviaciones en los productos que se van generando a lo largo del desarrollo. Por esta característica se adopta esta práctica para productos que son de especial importancia. En la reunión participan el responsable de SQA e integrantes del equipo de desarrollo.Se debe convocar a la reunión formalmente a los involucrados, informar del material que ellos deben preparar por adelantado, llevar una lista de preguntas y dudas que surgen del estudio del producto a ser revisado.Como salida se obtiene el Informe de RTF.

Los elementos mínimos que deberán ser revisados son:

* Especificación de Requerimientos
* Modelo de Diseño y Descripción de la Arquitectura
* Plan de Verificación y Validación
* Plan de Gestión del Proyecto
* Plan de Gestión de Configuración Diseño vs. Especificación de requerimientos
* Implementación vs. Diseño
* Verificación vs. Especificación de requerimientos

## Revisiones a realizar

El tipo de revisiones que deberán ser realizadas se describen a continuación:

|  |  |
| --- | --- |
| **Tipo de revisión** | **Descripción** |
| Revisión de requerimientos | Esta revisión se realiza para asegurar que se cumplió con los requerimientos especificados por el cliente. |
| Revisión de diseño preliminar | Esta revisión se realiza para asegurar la consistencia y suficiencia técnica del diseño preliminar del software. |
| Revisión de diseño crítico | Esta revisión se realiza para asegurar la consistencia del diseño detallado con la especificación de requerimientos. |
| Revisión del Plan de Verificación y Validación | Esta revisión se realiza para asegurar la consistencia y completitud de los métodos especificados en el Plan de Verificación y Validación. |
| Revisiones de gestión | Estas revisiones se realizan periódicamente para asegurar la ejecución de todas las actividades identificadas en este Plan. Deben realizarse por una persona ajena al grupo de trabajo (en caso de que sea posible). |
| Revisión del Plan de Gestión de la Configuración | Esta revisión se realiza para asegurar la consistencia y completitud de los métodos especificados en el Plan de gestión de configuración. |
| Revisión Post Mortem | Esta revisión se realiza al concluir el proyecto para especificar las actividades de desarrollo implementadas durante el proyecto y para proveer recomendaciones. |

*Tabla 6. Tabla de las revisiones a realizar dentro del Plan SQA*

Para el caso de auditorías a realizar, se toman a consideración las siguientes:

|  |  |
| --- | --- |
| **Auditoría** | **Descripción** |
| Auditoría funcional | Esta auditoría se realiza previa a la liberación del software, para verificar que todos los requerimientos especificados en el documento de requerimientos fueron cumplidos. |
| Auditoría física | Esta revisión se realiza para verificar que el software y la documentación son consistentes y están aptos para la liberación. |
| Auditorías internas al proceso | Estas auditorías son para verificar la consistencia: del código versus el documento de diseño, especificaciones de interfaz, implementaciones de diseño versus requerimientos funcionales, requerimientos funcionales versus descripciones de testeo. |

*Tabla 7. Tabla de las auditorías a realizar dentro del Plan SQA*

# 

# Pruebas

Las pruebas ~~de verificación~~ se harán conforme a lo expresado en el documento: Plan de verificación y validación.

Tipos de pruebas ~~sugeridas~~:

* Pruebas de integración
* Pruebas de aceptación
* Pruebas de Cajas Negras

Todo personal que desarrolle algún componente del proyecto realizará las siguientes actividades de pruebas necesarias del software y estas son responsables de:

* Realizar las pruebas Unitarias
* Realizar las pruebas de acuerdo al plan de pruebas del software
* Anotar los resultados de las pruebas
* Recomendar acciones correctivas si se encontraron defectos en los módulos probados

# Informe de problemas y acción correctiva

En esta sección se describe el reporte y control del sistema utilizado por el personal de calidad para registrar y analizar las discrepancias encontradas, así como para monitorear la implementación de las acciones correctivas. Los formatos utilizados para realizar los reportes se describen más adelante en esta sección.

## Reporte del proceso de auditorías

El personal de calidad reportará el resultado de las auditorías y las recomendaciones proporcionadas. Este reporte se usa para asegurarse que el proceso:

* Se está siguiendo de manera correcta y se está trabajando de forma efectiva
* Se está siguiendo, pero no se está trabajando de manera efectiva
* No se está siguiendo

## Presentación del reporte del proceso auditoría

El proceso de Reporte de Auditorías está dirigido hacia el Administrador del Proyecto el cual utilizará los reportes de las siguientes maneras:

* Para saber si los procesos de desarrollo son acatados y si son efectivos para el cumplimiento de las metas del proyecto. Cuando sea necesario el administrador del proyecto puede iniciar cambios a los procesos, mediante los procedimientos establecidos, para que los procesos queden estables.
* Para indicar el acuerdo, desacuerdo, o el aplazamiento de las recomendaciones hechas en el Proceso de Reporte de Auditoría. En caso de que el administrador del proyecto indique desacuerdo con las recomendaciones registradas en el proceso de reporte de auditoría, la disposición final de recomendaciones del informe se hace por el patrocinador del proyecto.

El formato para el reporte del proceso de auditorías puede ser consultado en el anexo 1.

## Procedimiento de escalamiento para la Resolución de no ocurrencia en el Proceso de Reporte de Auditoría

Al encontrarse un problema de calidad en algún elemento de trabajo ya sea documento, código o producto de software se tendrá que realizar lo siguiente:

* Primero se tratará con el creador de ese elemento:

1. Si existen problemas de desacuerdos en la resolución del problema, el personal de calidad tendrá que notificar al administrador del proyecto para que este tome cartas en el asunto y de una solución al problema;

2. Si el administrador del proyecto no da una solución, se escalará el problema al patrocinador del proyecto y él tomará la decisión final.

El personal de calidad conservará el registro original de las conclusiones tomadas y la resolución posterior en sus expedientes de auditoría.

Para el caso de las herramientas de software, consulte el anexo 2 para el formato de Reporte de Evaluación de Herramientas de Software.

# Técnicas, herramientas y metodologías

## Herramientas

Para esta sección se incluyen utilidades del sistema operativo, Debuggeadores, documentos de ayuda, checklist , analizadores de estructuras, analizadores de código, auditorías de estándares, monitoreo de rendimiento, software de desarrollo, matrices de seguimiento de software, pruebas de generadores de casos. A continuación, se mencionan las herramientas a ser empleadas y las condiciones en las cuales serán utilizadas:

* Lenguajes de programación: PHP, HTML, CSS, Java, Python.
* Herramientas para la elaboración diagramas UML: Jude, DB Designer, Star UML
* Herramientas de para el trabajo de bases de datos: MYSQL o Postgres
* Herramientas para elaborar Casos de Uso: Jude, Star UML
* Herramientas para procesamiento de texto: Microsoft Word
* Herramientas de apoyo: Internet, Excel, Photoshop, Figma, Github.
* Herramientas de desarrollo: Netbeans, Dreamweaver, Eclipse.

## Técnicas

En las técnicas se incluirá la revisión de uso de estándares, inspecciones de software, rastreo de requerimientos, verificación y validación de diseño y requerimientos, mediciones y evaluaciones de fiabilidad, análisis de lógica de negocio.

* Estándares: Codificación de Lenguajes, UML, Diseño de BD Lógicas y Físicas
* Programación en Pares.

Para las técnicas que ayudan a la evaluación de la calidad son:

* ANSI / IEEE –STD 830 Guide for Software Requirements Specifications
* ANSI / IEEE –STD 1016 Recommended Practice for Software Design Descriptions
* ANSI / IEEE –STD 1008 Standard for Software Unit Testing
* ANSI / IEEE –STD 1063 Standard for Software User Documentation
* ANSI / IEEE –STD 1028 Standard for Software Reviews and Audits

## Metodologías

Las metodologías de Aseguramiento de Calidad serán conjuntos integrados de técnicas, de entre los anteriores. Como principal metodología se emplearán:

* Programación orientada a objetos
* Programación orientada a eventos.

# Control del código

Para la revisión del código, se analizará el mismo para chequear que cumple

con los estándares definidos en este documento.

Para la documentación de cada entrega semanal, se validará que cumpla con los estándares definidos en este documento, revisando correctitud, completitud, y que cumpla con el formato definido.

Para revisar los atributos de calidad se tomarán como válidos los resultados de las pruebas realizadas por el Responsable de Verificación.

Para el control del código, se utilizarán las siguientes checklists:

|  |  |
| --- | --- |
| **Número** | **Pregunta** |
| **1** | ¿Definió los formatos de los distintos tipos de pantallas? |
| **2** | ¿Definió los controles y elementos de diseño de interfaz de pantalla, sus características, su disposición, y cómo se gestionan los eventos relacionados con ellos? |
| **3** | ¿Indicó la forma de navegar dentro de la pantalla y entre ellas? |
| **4** | ¿Definió los distintos tipos de formatos de impresión que se utilizarán en el sistema? |

Tabla 8. Revisión de las pautas de la interfaz de usuario

|  |  |
| --- | --- |
| **Número** | **Pregunta** |
| **1** | ¿La interacción es intuitiva de usar? |
| **2** | ¿Es fácil de aprender? |
| **3** | ¿Es correcta la rapidez de respuesta a las acciones del usuario? |
| **4** | ¿Se usó el color de forma efectiva? |
| **5** | ¿El usuario recibe un correcto feedback de la aplicación? |

Tabla 9. Revisión de la Interfaz de Usuario

|  |  |
| --- | --- |
| **Número** | **Pregunta** |
| **1** | ¿La arquitectura está compuesta de capas? |
| **2** | ¿Los paquetes están bien definidos, incluyen su funcionalidad e interfaz para otros módulos? |
| **3** | ¿Todas las estructuras de datos importantes están ocultas mediante funciones de acceso? |
| **4** | ¿Todos los objetos importantes están descritos y justificados? |
| **5** | ¿La interfaz con el usuario está modularizada tal que los cambios en ella no afecten el resto del programa? |
| **6** | ¿Se ha estimado el uso de memoria, se ha descrito y justificado una estrategia para la gestión de la memoria? |
| **7** | ¿Se ha incluido una estrategia coherente de manejo de errores? |
| **8** | ¿La arquitectura está diseñada para acomodar cambios? |
| **9** | ¿El diseño de alto nivel es independiente de la máquina y lenguaje que se usará para implementarlo? |

Tabla 10. Revisión de la Arquitectura

El programa a utilizar será uno utilitario, el cual para su identificación contará con un método de implementación de serie numérica única y versiones para su distinción.

El mismo software se encontrará en el área de desarrollo, donde el mismo equipo encargado de su desarrollo también cumplirá con la tarea de documentar y dar mantenimiento al software.

Para la distribución del software a todo el equipo se dispondrá del uso de la herramienta Github para el acceso del código fuente; para el caso de la documentación, esta podrá ser solicitada para su consulta por los miembros del equipo autorizados o quienes cuenten con una cuenta de Github para ver la documentación.

Para los cambios e implementación de versiones en la documentación se seguirán los procedimientos mencionados en la sección de “revisiones y auditorías”.

# Control de medios

En esta sección se definen los métodos y facilidades que se van a utilizar para proteger el medio físico de accesos no autorizados y daños y degradaciones inesperadas, y las organizaciones responsables para realizar este control.La organización responsable por esta tarea es la organización de desarrollo, con la supervisión de la organización de la SQA. Se debería asegurar que:

* Está garantizado el almacenamiento y recuperación de software.
* El software está accesible únicamente para aquellos que lo necesitan.
* Se controla el entorno para que no se degrade el medio físico en el que se almacena el software.
* Se almacenan copias del software crítico y del código en línea base fuera de las instalaciones de la organización

## Medio de almacenamiento

El medio del programa de computadora se define como aquellos medios sobre los cuales los datos son almacenados. Se utilizarán los siguientes medios:

* Los discos duros como dispositivos primarios.
* Los CD ́s y dispositivos USBs como almacenamiento secundario, para guardar las copias de seguridad.
* La documentación respectiva sobre el desarrollo de software (papel).
* Github como medio de respaldo para el código fuente y documentación.

## Proceso de copias de seguridad

Las copias de seguridad serán realizadas a la finalización de cada sesión de trabajo, registrándose la fecha y hora de copia de seguridad.

## Puntos de control

Para el acceso no autorizado, se debe asignar cuentas privilegiadas, cada usuario que interactúa con el software tendrá su propia cuenta de acuerdo al cargo que desempeñe y para el manejo del programa se debe contar con una cuenta de correo electrónico en Github. Se cuenta con la integridad de la Base de Datos para la protección de los datos. Se realizará una revisión periódica del software con el fin de que funcione óptimamente

# Recopilación, mantenimiento y retención de registros

~~Para esta sección se identifica aquella documentación que se debe retener, y se especifican los métodos y facilidades que se utilizarán para recolectar, proteger y mantener la documentación. También se especificará el período de retención para cada tipo de registro.~~

Los documentos que son requeridos son los siguientes:

* Plan de garantía de calidad del software.
* Especificación de requerimientos del software.
* Descripción del diseño del software.
* Plan de verificación y validación del software.
* Documentación del usuario.

Para el mantenimiento de los registros del software, será realizado por versiones sucesivas de actualizaciones de las mismas, para esto se lleva un registro de actualizaciones de la documentación. Los documentos verificados y validados, deben ser documentados en libros impresos, con tres copias de cada documento y almacenado en lugares diferentes y ambiente adecuados como Github y en disco duros. La retención de registros se realizará en cada finalización de las fases del ciclo de vida de desarrollo de software y según los puntos de verificación y validación.

# Entrenamiento

Existen diferentes disciplinas bajo las cuales los miembros del equipo de desarrollo de software tienen opción para formarse. En esta sección a través de la Tabla 11, se indican los conocimientos o habilidades deseables para realizar efectivamente las tareas orientadas a aseguramiento de calidad.

Es recomendable que las personas designadas a la unidad de aseguramiento de calidad tengan conocimiento de las herramientas utilizadas por el grupo de desarrolladores y preferentemente que hayan participado en el desarrollo de al menos un proyecto.

|  |  |  |
| --- | --- | --- |
| **Tarea** | **Habilidades Requeridas** | **Tipo de Revisión** |
| Revisión de código | * Lenguaje de programación aplicado * Marcos de trabajo aplicados | Revisión por pares |
| Revisión de documentación | * Modelo de calidad de referencia * Metodología de desarrollo de software * Guías institucionales de elaboración | Revisión técnica formal  Inspecciones |
| Auditoría del Proceso de Desarrollo | * Proceso de desarrollo de software aplicado * Estándares tomados como referencia | Auditoría del proceso realizado en sus diferentes fases |
| Pruebas de Software | * Niveles y técnicas de pruebas * Estrategia de pruebas * Herramientas de soporte a la realización de pruebas | Auditoría del proceso de pruebas |
| Administración | * Administración de proyectos * Técnica de valor ganado * Recopilación y análisis de métricas | Auditoría del proceso administrativo |
| Control de Cambios | * Administración de la configuración (registro, control de cambios, seguimiento y atención) | Auditoría al proceso de control de cambios |
| Control de Riesgos | * Administración de riesgos * Auditoría al proceso (identificación, análisis, control de riesgos, atención) | Auditoría al proceso de control de riesgos |

Tabla 11. Habilidades requeridas para realizar las revisiones

El plan de entrenamiento aplicado a las personas que formarán parte de la unidad de aseguramiento de calidad, como se mencionó previamente, debe iniciar con los desarrolladores. La transmisión de conocimiento hacia el grupo de desarrolladores pretende que realicen las diferentes fases del proceso de construcción de software de forma sistemática y alineada a un modelo de calidad.

Cada desarrollador júnior trabajará en conjunto con un desarrollador sénior quien será el responsable de conducir su entrenamiento de acuerdo a las actividades que le sean asignadas, cabe aclarar que no se aplica un plan de entrenamiento general, más bien se prepara al personal de acuerdo al objetivo que debe alcanzar, de tal forma que si su asignación es la generación del documento de especificación de requerimientos, conocerá el estándar o estándares de referencia y los lineamientos que debe seguir para generarlo.

Una vez que un desarrollador adquiere experiencia y consolida sus conocimientos su participación en el grupo de aseguramiento de calidad debe ser más efectiva.

# Anexos

Anexo 1. Formato del reporte del proceso de auditorías

|  |  |
| --- | --- |
| REPORTE DEL PROCESO DE AUDITORÍAS | |
| Identificador de seguimiento: | |
| Auditor: | Fecha de reporte: |
| Equipo: | Nombre del proyecto: |
| Fecha de la auditoría: | |
| Proceso/procedimiento auditado: | |
| Checklist de auditoría usado (agregar): | |
| Resultado de la auditoría: (Marque uno solo)  \_\_\_\_\_ Proceso/Procedimiento Aceptable  \_\_\_\_\_ Proceso/Procedimiento Condicionalmente aceptable  \_\_\_\_\_ Proceso/Procedimiento Inaceptable  (Sujeto a la finalización satisfactoria de los puntos mencionados arriba)  Notas de Condiciones:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  TITULO: ENCARGADO: FECHA DE VENCIMIENTO:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|
| Acciones correctivas:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| Disposición: (Marque el correspondiente)  \_\_\_\_ Aprobado  \_\_\_\_ Cancelado  \_\_\_\_ Aplazado | |
| Administrador del proyecto: Fecha:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| Firma del encargado de SQA: Fecha: | |

Anexo 2. Formato de Reporte de Evaluación de Herramientas de Software

|  |
| --- |
| **EVALUACIÓN DE HERRAMIENTAS DE SOFTWARE** |
| SQA: Fecha de evaluación: |
| Herramienta de software evaluada: |
| Métodos o criterios usados en esta evaluación: |
| Resultados de la evaluación: |
| Acciones correctivas recomendadas: |
| Acciones correctivas tomadas: |