# Sistemas De Información Del Control De Inventarios Mundo Accesorios Style

**PLAN ASEGURAMIENTO DE CALIDAD**

# BRAYAN CARDENAS DIANA YEPES VALERIA RUIZ FABIAN DIAZ

**2558108**

# ANALISIS Y DESARROLLO DE SOFTWARE

**CENTRO DE ELECTRICIDAD, ELECTRONICA Y TELECOMUNICACIONES CEET**

# 2024

## Aprobación del plan de SQA:

**Administrador de SQA Fecha**

## Administrador del proyecto Fecha

**PREFACIO**

Este documento contiene el Plan de garantía de calidad del software (SQA) para el Jucar Autopartes SAS. Las actividades de SQA descritas en este plan son coherentes con el Plan de desarrollo de software de Jucar Autopartes SAS y otros documentos de planificación del proyecto. Este documento se ha adaptado a partir de la plantilla del plan SQA, TM-SQA-01, v2.0.

El Proyecto asume la responsabilidad de este documento y lo actualiza, según sea necesario, para satisfacer las necesidades de Jucar Autopartes SAS. Los usuarios de este documento pueden reportar deficiencias o correcciones utilizando la Solicitud de cambio de documento que se encuentra al final del documento. Las actualizaciones de este documento se realizarán, al menos una vez al año

SOLICITUD DE CAMBIO DE DOCUMENTO (DCR)

|  |  |
| --- | --- |
| Título del Documento: **Plan de aseguramiento de calidad de Software** | Numero de Rastreo |
| **Nombre de la organización que presenta: Sistemas De Información Del Control De Inventarios Mundo**  **Accesorios Style** | |
| **Contacto de la Organización: BRAYAN CARDENAS** | Teléfono: 3137331835 |
| **Dirección de envió: Carrera 153#133-10** | |
| **Titulo Corto:Mundo Accesorios Style** | Fecha: |
| **Cambio Propuesto** | |
| **Racional para el Cambio** | |

Nota:

## REGISTRO DE CAMBIOS

**A+** AGREGADO, **M**- MODIFICADO, **D-** ELIMINADO

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NUMERO DE  VERSION | FECHA | NUMERO DE FIGURA, TABLA O PARRAFO | UNI°  D | TITULO O BREVE DESCRIPCION | CAMBIO PETECION  NUMERO |
|  |  |  |  |  |  |

Contenido

[1.1 ALCANCE 11](#_bookmark0)

* 1. [Vista general del sistema 12](#_bookmark1)
  2. [Relación con otros planes 13](#_bookmark2)
  3. [Referencias 13](#_bookmark3)

[Administración 14](#_bookmark4)

* 1. [Organización 14](#_bookmark5)
  2. [Recursos 15](#_bookmark6)
     1. [Instalaciones y equipos 15](#_bookmark7)
     2. [personal 16](#_bookmark8)

[Tareas de SQA 16](#_bookmark9)

* 1. [Revisar los productos de software 16](#_bookmark10)
  2. [Evaluar las herramientas de software 16](#_bookmark11)
  3. [Evaluar las instalaciones 17](#_bookmark12)
  4. [Evaluar los productos 17](#_bookmark13)
  5. [Evaluar los planes 17](#_bookmark14)
  6. [Evaluar los requerimientos 17](#_bookmark15)
  7. [Evaluar el diseño del software 17](#_bookmark16)
  8. [Evaluar el desarrollo del software y el proceso de pruebas de unidad 17](#_bookmark17)
  9. [Pruebas de integración 18](#_bookmark18)
  10. [Evaluar el proceso de acciones correctivas 18](#_bookmark19)
  11. [Llevar a cabo las revisiones del proyecto y las auditorias 18](#_bookmark20)
      1. [Llevar a cabo revisiones técnicas 18](#_bookmark21)
      2. [Verificar avances del proyecto 19](#_bookmark22)

[Documentación 19](#_bookmark23)

[Estándares, métricas y practicas 20](#_bookmark24)

[5.1 Métricas SI 20](#_bookmark25)

[Estándares de diseño y codificación 20](#_bookmark26)

[Pruebas 21](#_bookmark27)

[Reportar problemas y resolución 22](#_bookmark28)

* 1. [Reporte del proceso de auditorias 22](#_bookmark29)
     1. [Presentación y disposición del reporte del proceso de auditoria 22](#_bookmark30)
     2. [Procedimiento de escalamiento para la resolución de problemas de no acuerdo en el proceso del](#_bookmark31) [reporte de auditorias 25](#_bookmark31)
  2. [Problemas de registro en código 25](#_bookmark32)
  3. [Reporte de evaluación 25](#_bookmark33)
  4. [Reporte de la evaluación de instalaciones 25](#_bookmark34)

[Herramientas 27](#_bookmark35)

[Control de código 29](#_bookmark36)

[Sección 10. Entrenamiento 29](#_bookmark37)

[Sección 11. Control de medios 30](#_bookmark38)

[Sección 12. Control de proveedores 31](#_bookmark39)

SECCION 1.

OBJETIVO

Con base en el análisis de las problemáticas y necesidades encontradas en el establecimiento de ventas a partir de un levantamiento de información desarrollado en la empresa “Mundo accesorios Style” en los cuales se encontraron problemas en su gestión administrativa y específicamente en los procesos de inventario, El presente documento muestra el desarrollo de un sistema de información en su fase inicial, partiendo desde la concepción de los requisitos del sistema hasta los diagramas que modelan la solución de las necesidades de la empresa.

**Modelo de calidad plateado ISO 9000**

Se utilizo la norma ISO 9000 y buscar la certificación ISO 9001 puede proporcionar a una organización una serie de beneficios significativos, incluida la mejora de la calidad, el cumplimiento de requisitos contractuales, el acceso a nuevos mercados, la mejora de la eficiencia operativa, el aumento de la satisfacción del cliente y la facilitación de la toma de decisiones basadas en datos.

Enfoque al cliente: La norma requiere que las organizaciones comprendan y cumplan con los requisitos y expectativas de los clientes. Esto implica identificar y satisfacer las necesidades del cliente, así como monitorear continuamente la satisfacción del cliente.

Liderazgo: La alta dirección de la organización debe demostrar liderazgo y compromiso con el sistema de gestión de calidad, estableciendo una visión clara, políticas y objetivos de calidad, así como asignando roles y responsabilidades.

Participación del personal: Todos los miembros del personal deben estar involucrados en el logro de los objetivos de calidad de la organización. Esto implica proporcionar capacitación y recursos adecuados, así como promover la comunicación y la colaboración entre los equipos.

Enfoque basado en procesos: La norma requiere que las organizaciones adopten un enfoque basado en procesos para la gestión de calidad, identificando, entendiendo y gestionando los procesos interrelacionados como un sistema coherente.

Mejora continua: La organización debe buscar continuamente la mejora de la eficacia de su sistema de gestión de calidad a través de la identificación de oportunidades de mejora y la implementación de acciones correctivas y preventivas.

Enfoque basado en evidencias para la toma de decisiones: Las decisiones deben basarse en el análisis de datos y la información relevante, asegurando la precisión y la eficacia de las decisiones tomadas.

Gestión de relaciones con los proveedores: La organización debe establecer relaciones mutuamente beneficiosas con sus proveedores, comunicar claramente los requisitos del cliente y monitorear el desempeño del proveedor para garantizar la calidad de los productos y servicios suministrados.

## ALCANCE

* Módulos a desarrollar:

1. Cliente: Registrar cliente, consultar cliente, actualizar cliente y eliminar cliente.
2. Usuario: Registrar usuario, consultar usuario, actualizar usuario y eliminar usuario.
3. Proveedor: Registro de proveedor, consultar proveedor, actualizar proveedor y eliminar proveedor.
4. Factura proveedor: Registrar pedido, consultar pedido, actualizar pedido y eliminar pedido.
5. Factura cliente: Registrar venta, consultar venta, actualizar venta y eliminar venta.
6. Detalle factura proveedor: Registrar detalle de factura, consultar detalle de factura, actualizar detalle de factura y eliminar detalle de factura.
7. Detalle factura cliente: Registrar detalle de factura, consultar detalle de factura, actualizar detalle de factura y eliminar detalle de factura.
8. Categoría: Registrar categoría, consultar categoría, actualizar categoría y eliminar categoría.
9. Stock: Registrar inventario, consultar inventario, actualizar inventario y eliminar inventario.
10. Producto: Registrar producto, consultar producto, actualizar producto y eliminar producto.
11. Precio: Registrar precio, consultar precio, actualizar precio y eliminar precio.
12. Método de pago: Registrar método de pago, consultar método de pago, actualizar método de pago y eliminar método de pago.
13. Entidad bancaria: Registrar entidad bancaria, consultar entidad bancaria, actualizar entidad bancaria y eliminar entidad bancaria.
14. Efectivo: Registrar efectivo, consultar efectivo, actualizar efectivo y eliminar efectivo.
15. Reportes: Generación de reportes acerca de las ventas según lo que sugiere el administrador como fechas, valor, cantidad, etc.
    * Actores con acceso:
16. Caja
17. Bodega
18. Administración

Permisos por actor:

Caja: Tiene permisos de Registrar el cliente, consulta al cliente, actualizar cliente, registrar venta, consultar venta, actualizar venta, registrar detalle de factura, consultar detalle de factura, actualizar detalle de factura, registrar método de pago, consultar método de pago, actualizar método de pago, registrar entidad bancaria, consultar

entidad bancaria, actualizar entidad bancaria/ registrar efectivo, consultar efectivo, actualizar efectivo.

Bodega: tiene permisos de registro de proveedor, consultar proveedor, actualizar proveedor, registrar pedido, consultar pedido, actualizar pedido, registrar detalle de factura, consultar detalle de factura, actualizar detalle de factura, registrar categoría, consultar categoría, actualizar categoría, registrar inventario, consultar inventario, actualizar inventario, registrar producto, consultar producto, actualizar producto, registrar precio, consultar

precio, actualizar precio.

Administrador: tiene permisos de ingresar a cualquier entidad, realizar las eliminaciones requeridas y afirmadas por su autorización por lo que él solo las podrá eliminar y generar reportes.

## Identificación

Los elementos que están a continuación son a los que se les aplicara el plan de aseguramiento de la configuración.

Al cliente

* Especificación de requerimientos
* Manuales de usuario y de ayuda Trabajos internos
* SQA
* SDP
* Especificación de diseño
* Planes y resultados de pruebas
* Estándares y procedimientos Herramientas de programación

## Vista general del sistema

“Sistema de inventarios”, es un sistema enfocado a la administración de inventarios para la empresa Mundo Accesorios Style para solucionar los problemas administrativos en la empresa y con el levantamiento de

información adecuada asegurar una satisfacción del cliente con el producto

## Vista general del documento

Este documento contiene toda la organización y procedimientos a ser usadas para llevar a cabo en el plan SQA del proyecto

En la sección 1 se identifica el sistema al que se le va a aplicar este plan, se hace una pequeña descripción del sistema, se identifican otros documentos de administración relacionados con el plan SQA y adicionalmente se describe el propósito del plan SQA

En la sección 2 se describe cada elemento de la organización que influye en la calidad del software Sección 3 se describen las tareas de aseguramiento de la calidad del software

Sección 4 lista los documentos que están en la línea base o van a estar en la línea base durante el desarrollo del proyecto

Sección 5 describe todos los convenios, estándares y practicas a ser usadas en el desarrollo del proyecto Sección 6 describe el rol que tiene SQA en las pruebas

Sección 7 describe los reportes de errores y las acciones correctivas Sección 8 describe herramientas de SQA, técnicas y metodologías

Sección 9 describe la herramienta usada en la administración de la configuración para el control del código Sección 10 describe los entrenamientos requeridos para SQA

Sección 11 contiene los checklist a ser usados durante el desarrollo del proyecto, esto será usado para verificar que todos los procesos sean realizados con el aseguramiento de la calidad

## Relación con otros planes

El SQA está implementado de acuerdo con el plan de desarrollo del software y el plan de configuración del software

Todos los procesos definidos en el plan de aseguramiento de calidad que se usaran durante el ciclo de vida del proyecto están basados en los procesos definidos en el plan de desarrollo de software

## Referencias

* + 1. Plan de desarrollo de software

B) Plan de administración de configuración del software

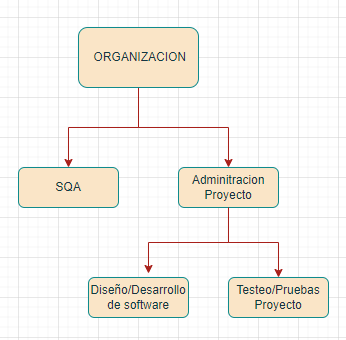
## 

## Administración

Esta sección describe cada elemento o área de organización que influye en la calidad del software

## Organización

En la siguiente figura se muestra la organización y los elementos que lo conforman, todos estos deben tener una cierta influencia en el plan de aseguramiento de la calidad



A continuación, se describe el rol que tiene cada elemento en el plan de aseguramiento de la calidad del software

* + 1. La gerencia es la responsable de:
       1. Todo el aspecto administrativo y económico
       2. Es a quien el equipo de SQA debe informar en caso de errores graves en el producto de trabajo
       3. Aprobar el documento de SQA
    2. SQA
       1. Es el encargado de definir el plan SQA
       2. Es el encargado de ejecutar el plan
       3. Verifica que todo lo establecido se siga al pie de la letra
       4. Es el responsable de verificar que todos los productos liberados cumplan con los requisitos de calidad establecidos
       5. Realizan las inspecciones
       6. Revisiones técnicas de los productos de software establecidos
    3. Administración de proyecto es responsable de:
       1. Implementar el plan de calidad establecidos en el plan de SQA
       2. Identificar las actividades de SQA a ser llevadas a cabo por el equipo de SQA
       3. Revisar y aprobar el plan de SQA
       4. Identificar una persona o grupo de personas para llevar a cabo las tareas de SQA
       5. Identificar y darle seguimiento a cualquier problema de calidad reportado por SQA
       6. Identificar y asegurar todos los factores a ser implementados en el sistema
       7. Identificar, desarrollar y dar mantenimiento documentos de planeación tales como: el plan de desarrollo de software y el plan de aseguramiento de calidad del software
    4. Diseño/desarrollo de software
       1. Revisar y comentar acerca del plan de SQA
       2. Identificar y darle seguimiento a cualquier problema de calidad reportado por SQA, que esté relacionado con el diseño y desarrollo del producto
       3. Implementar las practicas, procesos y estándares de desarrollo y diseño especificados en el plan de desarrollo del software y en el plan SQA

## Recursos

### Instalaciones y equipos

El equipo de SQA tendrá acceso a las instalaciones y equipos definidos en el plan de desarrollo del software, el equipo de SQA tendrá acceso a los recursos computacionales para realizar las funciones tales como evaluar los productos o realizar las auditorias

### personal

El perfil de los integrantes de SQA es el siguiente:

* + - 1. Debe estar familiarizado con las pruebas de software
      2. Debe conocer las partes de un plan de aseguramiento de la calidad del software
      3. El conocimiento en diseño, código, análisis estructural y pruebas de software
      4. Debe conocer los planes de:
         1. Desarrollo de software
         2. Administración de la configuración de software
      5. Trabajar en equipo

El administrador de SQA debe dominar los temas ya mencionados

**Planes Desarrollados**

Aquí hay una descripción básica de cada uno de los planes que mencionaste:

1. Plan de Instalación: Este plan detalla los pasos necesarios para instalar un sistema, software o infraestructura. Incluye la preparación del entorno, la instalación real del sistema o software, la configuración inicial y cualquier prueba necesaria para garantizar que todo funcione correctamente. También puede abordar aspectos como requisitos de hardware, compatibilidad de software, configuraciones de red, seguridad y cualquier otro detalle relevante para la instalación.

2. Plan de Respaldo: Este plan describe cómo se realizarán las copias de seguridad de los datos importantes para garantizar su disponibilidad en caso de pérdida, daño o corrupción. Incluye la frecuencia de las copias de seguridad, los métodos de almacenamiento utilizados (por ejemplo, almacenamiento en la nube, unidades externas, servidores dedicados), los procedimientos para realizar y verificar las copias de seguridad, así como la política de retención de datos.

3. Plan de Migración: Este plan se refiere al proceso de trasladar datos, aplicaciones, sistemas o infraestructura de un entorno a otro. Puede ser necesario cuando se actualiza o reemplaza la tecnología existente, se cambia de proveedor de servicios, se fusionan empresas, entre otras situaciones. El plan de migración describe los pasos necesarios para minimizar el tiempo de inactividad, garantizar la integridad de los datos durante el proceso y asegurar una transición suave hacia el nuevo entorno.

4. Plan de Capacitación: Este plan se centra en cómo preparar a los usuarios finales, administradores de sistemas u otros miembros del equipo para utilizar eficazmente un nuevo sistema, software o tecnología. Puede incluir la creación de materiales de capacitación, la programación de sesiones de capacitación, la identificación de los temas a cubrir, la selección de instructores y la evaluación del progreso de los participantes. El objetivo es asegurar que todos los usuarios estén familiarizados y cómodos con la nueva tecnología para maximizar su eficacia y minimizar los errores.

**Manuales Desarrollados**

1. Manual de Instalación: Este manual proporciona instrucciones detalladas sobre cómo instalar y configurar un sistema, software o hardware. Suele incluir pasos específicos, requisitos de sistema, configuraciones recomendadas, configuraciones de red (si es aplicable), solución de problemas comunes durante la instalación, y cualquier otro detalle técnico relevante para llevar a cabo la instalación de manera correcta y eficiente.

2. Manual Técnico: Este tipo de manual está dirigido a usuarios técnicos o administradores de sistemas y proporciona información detallada sobre el funcionamiento interno y la configuración del sistema, software o hardware. Puede incluir detalles sobre la arquitectura del sistema, los protocolos de comunicación, la configuración avanzada, los componentes internos, las API (Interfaz de Programación de Aplicaciones), la integración con otros sistemas, y cualquier otra información técnica relevante para aquellos que necesitan comprender en profundidad el funcionamiento del producto.

3. Manual de Usuario : Este manual está diseñado para los usuarios finales del sistema, software o hardware y proporciona instrucciones paso a paso sobre cómo utilizarlo de manera efectiva. Suele incluir información sobre la interfaz de usuario, funciones básicas y avanzadas, configuraciones, ajustes, solución de problemas comunes, y cualquier otra información que los usuarios necesiten para aprovechar al máximo el producto. El manual de usuario se redacta en un lenguaje claro y accesible para garantizar que los usuarios puedan seguir las instrucciones sin dificultad.

**Pruebas Desarrolladas**

## Aquí te describo brevemente cada uno de los términos que has mencionado relacionados con pruebas de software:

## 1. Casos de Prueba: Son documentos que describen las condiciones de entrada, acciones a realizar y resultados esperados para verificar el cumplimiento de una determinada función del software. Los casos de prueba son utilizados por los testers para ejecutar pruebas de forma sistemática y garantizar que el software funcione correctamente.

## 2. Reporte de Bugs: Documento que recopila información sobre los defectos encontrados durante las pruebas de software. Incluye detalles como la descripción del bug, pasos para reproducirlo, sistema operativo y versión del software, prioridad, estado actual del bug, entre otros. Este reporte es utilizado por el equipo de desarrollo para corregir los errores.

## 3. Pruebas Unitarias: Son pruebas realizadas a nivel de componentes individuales o unidades de código (como funciones o métodos) para verificar su correcto funcionamiento de manera aislada. Las pruebas unitarias suelen ser escritas por los propios desarrolladores y se ejecutan automáticamente durante el proceso de desarrollo.

## 4. Pruebas de Integración: Se realizan para probar la interoperabilidad entre diferentes módulos, componentes o subsistemas del software. El objetivo es garantizar que todas las partes del sistema funcionen correctamente juntas después de ser integradas.

## 5. Pruebas a la API: Son pruebas que se realizan para verificar el comportamiento de una interfaz de programación de aplicaciones (API). Se centran en probar las solicitudes, respuestas y funcionalidades expuestas por la API, asegurando su correcto funcionamiento y comportamiento esperado.

## 6. Pruebas Automatizadas: Son pruebas que se ejecutan automáticamente mediante el uso de herramientas de prueba automatizadas. Estas pruebas pueden incluir pruebas unitarias, pruebas de regresión, pruebas de integración, entre otras. La automatización de pruebas ayuda a mejorar la eficiencia y la calidad del proceso de prueba.

## 7. Pruebas de Rendimiento: Son pruebas diseñadas para evaluar el rendimiento del sistema bajo diferentes condiciones de carga y uso. El objetivo es identificar cuellos de botella, tiempos de respuesta lentos o problemas de escalabilidad que puedan afectar el rendimiento del software en producción.

## 8. Pruebas de Aceptación de Usuarios: También conocidas como pruebas de aceptación del usuario (UAT, por sus siglas en inglés), son pruebas realizadas por usuarios finales o representantes del cliente para verificar si el software cumple con los requisitos y expectativas del usuario final. Estas pruebas suelen realizarse en un entorno similar al de producción antes de la implementación final del software.

## Tareas de SQA

En esta sección se listan todas las tareas que el equipo de SQA realizara, estas tareas serán durante todo el ciclo de vida del proyecto y se realizaran según la calendarización descrita en el plan de desarrollo de software.

Una tarea se considerará completa si se ha levantado un reporte acerca de esa tarea.

Las siguientes tareas requieren de la coordinación y cooperación del equipo de desarrollo para ser llevadas a cabo de forma satisfactoria por el equipo de SQA.

## Revisar los productos de software

El plan de desarrollo de software identifica todos los productos de software a ser desarrollados y evaluados e igual que los estándares o reglas a seguir. Es necesario que el equipo de SQA ayude a la administración del proyecto, los estándares y los lineamientos a seguir en el desarrollo del proyecto

## Evaluar las herramientas de software

El SQA deberá evaluar las herramientas, tanto existentes como las que se planean obtener, usadas para el desarrollo y soporte, durante el desarrollo del proyecto.

Las herramientas deben ser evaluadas teniendo en cuenta si realizan las funciones para las que son consideradas, si en verdad pudieran ayudar al desarrollo ya sea disminuyendo tiempos o haciendo que los procesos de

desarrollo sean más fáciles.

Las herramientas de software planeadas para su adquisición deben cumplir los requisitos mencionados

anteriormente, es decir que en realidad sean útiles para el desarrollo del proyecto y que además sean soportados por los sistemas de cómputo actuales en los que se desarrolla el proyecto.

## Evaluar las instalaciones

SQA debe evaluar las instalaciones actuales y futuras, estos deben proveer el equipo necesario y el espacio para el desarrollo y soporte.

## Evaluar los productos

Esta tarea se encargará de verificar que los procesos de calidad estén presentes en todos los productos del software.

SQA hará una revisión a todos los productos que estén listos para su revisión y además verificará que los resultados sean reportados y los problemas descubiertos y resueltos acorde a los tiempos acordados

## Evaluar los planes

El equipo de SQA deberá evaluar todos los planes para el proyecto, además de ayudar a identificar los estándares y lineamientos para los planes del proyecto nombrados en la sección 1.6.

## Evaluar los requerimientos

El análisis de los requisitos establece un único entendimiento del problema entre el cliente y el equipo de

desarrollo. Un acuerdo con el cliente es establecido con el fin de definir los requisitos para el software y el mismo es mantenido durante el desarrollo del proyecto.

Las tareas SQA para los requerimientos son las siguientes:

* + 1. Verificar que todos los participantes correctos estén involucrados en el análisis de los requerimientos para identificar todas las necesidades del usuario.
    2. Revisar que todos los requerimientos sean posibles de implementar.
    3. Verificar que todos los requerimientos fueron comunicados, documentados y revisados.
    4. Verificar que todos los requerimientos que pudieran tener un error fueran revisados y corregidos por el equipo de requerimientos.
    5. Darle seguimiento a los cambios que pudieran tener los requerimientos.

SQA usara una checklist para realizar la evaluación, todos los resultados se reportarán a la administración del proyecto con el fin de darle seguimiento.

## Evaluar el diseño del software

Las tareas de SQA en el proceso de diseño son:

* + 1. Verificar que los procesos de diseño de software sigan los estándares determinados.
    2. Verificar que todos los elementos que no cumplen con la calidad requerida sean procesados de acuerdo con los estándares y procedimientos establecidos.
    3. Verificar que todos los requerimientos estén presentes en el diseño.

Se generará un reporte, así como las acciones correctivas, después será decisión de la administración del proyecto si son llevadas a cabo o no.

## Evaluar el desarrollo del software y el proceso de pruebas de unidad

SQA llevara a cabo 3 diferentes tareas.

* + 1. Verificar que los procesos de codificación y pruebas de unidad están siendo implementados de acuerdo con los estándares y procesos establecidos.
    2. Verificar que los elementos encontrados en las revisiones de código sean procesados mediante los estándares y procedimientos establecidos.
    3. Verificar que las pruebas sean realizadas al pie de la letra.

## Pruebas de integración

Se tienen en cuenta las siguientes tareas para llevar a cabo las pruebas de integración.

* + 1. Verificar que las actividades de pruebas de software estén identificadas y el ambiente donde se van a llevar a cabo las pruebas ya estén identificadas.
    2. El equipo SQA llevara a cabo las pruebas.
    3. Los lineamientos de las pruebas ya están definidos.
    4. Se ejecutaron las pruebas de unidad con anterioridad.
    5. El equipo SQA ejecutara las pruebas de integración.
    6. Se deben analizar los resultados de las pruebas.

## Evaluar el proceso de acciones correctivas

A continuación, se describe el proceso de acción correctiva

* + 1. Identificar el problema.
    2. Reportar el problema al equipo correspondiente.
    3. Analizar el problema y dar una solución.
    4. Realizar la acción correctiva. Actividades de SQA:

1. Revisar el proceso de acción correctiva constantemente para estar seguro de que se está aplicando con los estándares establecidos.
2. Realizar análisis periódicos a todos los problemas reportados para identificar tendencias que puedan generar problemas genéricos en las áreas. Este análisis debe tener el estudio de las causas, la magnitud del impacto, la frecuencia en la que ocurre y medidas de prevención.

Se genera un reporte en el que se ven reflejadas las acciones correctivas y la administración del proyecto decide si se aplicaran o no.

## Llevar a cabo las revisiones del proyecto y las auditorias

El equipo de SQA será el encargado de realizar revisiones y auditorias constantes a las diferentes áreas

relacionadas al desarrollo del proyecto, esto con el fin de prevenir errores y asegurar la calidad del software.

### Llevar a cabo revisiones técnicas

El equipo de SQA es el encargado de lleva a cabo todas las revisiones técnicas, todo el equipo de SQA debe tener conocimientos acerca del producto que se evaluara.

Los elementos a los que se les aplicara la revisión técnica son los siguientes:

* + - 1. **Especificación de requisitos de software:** El objetivo es determinar que los requisitos están listos, para que se pueda pasar al diseño arquitectónico.
      2. **Arquitectura del software:** El objetivo es determinar que los requisitos se encuentran contemplados en el diseño de alto nivel.
      3. **Diseño de software:** El objetivo es determinar que todos los requisitos que están presentes en el diseño y que el diseño es consistente.
      4. **Desarrollo de software:** El objetivo es determinar que todos los productos de software desarrollados cumplen con los estándares establecidos

### Verificar avances del proyecto

SQA verificará periódicamente el estado del proyecto, el progreso, los problemas y los riesgos proporcionando una valoración individual de cada aspecto, SQA le proveerá la siguiente información a la administración del proyecto:

Cumplimiento: Identificara el nivel de desarrollo que se presenta en el proyecto teniendo en cuenta que cosas se llevan a cabo y que cosas no

Problemas: Identificara los problemas potenciales o actuales que pueden afectar en el desarrollo del proyecto. Riesgos: Identificara los riesgos basándose en el estado del proyecto.

Posteriormente los resultados se le comparten a la administración del proyecto y al equipo de desarrollo.

## Documentación

En esta sección se encuentra la documentación que describe y da soporte al sistema SI en el desarrollo de este o en su fase final, este fue creado y actualizado a lo largo del ciclo de vida del desarrollo.

En la siguiente tabla se listan los documentos relacionados con el SI.

|  |  |
| --- | --- |
| **Nombre del Documento** | **Descripción del Documento** |
| **IEEE 830** | En este documento se describen todos los requisitos del  producto que serán implementados |
| Plan de desarrollo de software | Este documento indica todo lo que se va a implementar del producto, las actividades a realizar y la asignación de responsabilidades |
| SI JUCAR AUTOPARTES SAS | En este documento se describen todos los planes y roles que tendrá cada  elemento del grupo de desarrollo en el proceso de aseguramiento de la calidad del software |
| Plan de administración de configuración del software | En ese plan se establece la forma de determinar la línea base y así la nomenclatura de cada producto de trabajo |
| **Diseño de bajo nivel** | En este documento se encuentra el diseño de bajo nivel del sistema |
| **Plan de Pruebas** | Este documento contiene el esquema acerca de que se va a probar y  como se va a probar |
| **Arquitectura** | Este documento contiene el diseño del SGI |

## Estándares, métricas y practicas

En esta sección se describen los estandartes, prácticas y procesos que hay que realizar para que el proyecto tenga éxito.

Los estándares (de como un producto puede ingresar a la línea base), procesos relacionados con administración de la configuración están definidos en el plan de administración de la configuración.

Los estándares (normas a seguir en el desarrollo del software), métricas (acerca de en qué estado se encuentra el proyecto) se encuentra el plan de desarrollo de software.

Los estándares (de como identificar riesgos), y las prácticas para mitigar los riesgos están descritos en el plan de administración de riesgos.

Los estándares de codificación y diseño se encuentran en la sección 5.2

## Métricas SI

* + 1. Calendario de trabajo de SI. (Planeado)
    2. Calendario de trabajo de SI. (Actual)
    3. Trabajo de SI. (Completo)
    4. Gastos de SI. (Planeados)
    5. Gastos de SI. (Actual)
    6. Gastos de reserva de SI. (Planeado)
    7. Gastos de reserva de SI. (Actual)
    8. Número de elementos no resueltos abiertos actualmente
    9. Número de elementos no resueltos cerrados
    10. Número de elementos no resueltos

## Estándares de diseño y codificación

* + 1. Lista de estándares de diseño.
       1. El desarrollo de la API se realizará en C#
       2. El desarrollo de la interfaz gráfica utilizando React
       3. El nombre de las clases y métodos estará en ingles
       4. El nombre del método será descriptivo sobre la función a realizar
    2. Lista de estándares de codificación
       1. Se hará uso de DTOs los cuales deben seguir la siguiente estructura de forma respectiva segun los datos del DTO:

public record ProductoDTO

{

public Guid ProductoId { get; init; } public string? Nombre { get; init; } public int Cantidad { get; init; }

public float Precio { get; init; }

public string? Estado { get; init; } public string? Lugar { get; init; }

public Guid StockId { get; init; } public Guid IdCategoria { get; init; }

}}

* + - 1. Todas las clases especificadas en el diseño se deben encontrar en el código.
      2. La codificación será en ingles exceptuando nombres de entidades.
      3. Las interfaces mantendrán el mismo nombre del servicio al que se relaciona con la única diferencia de tener una (I) al comienzo.

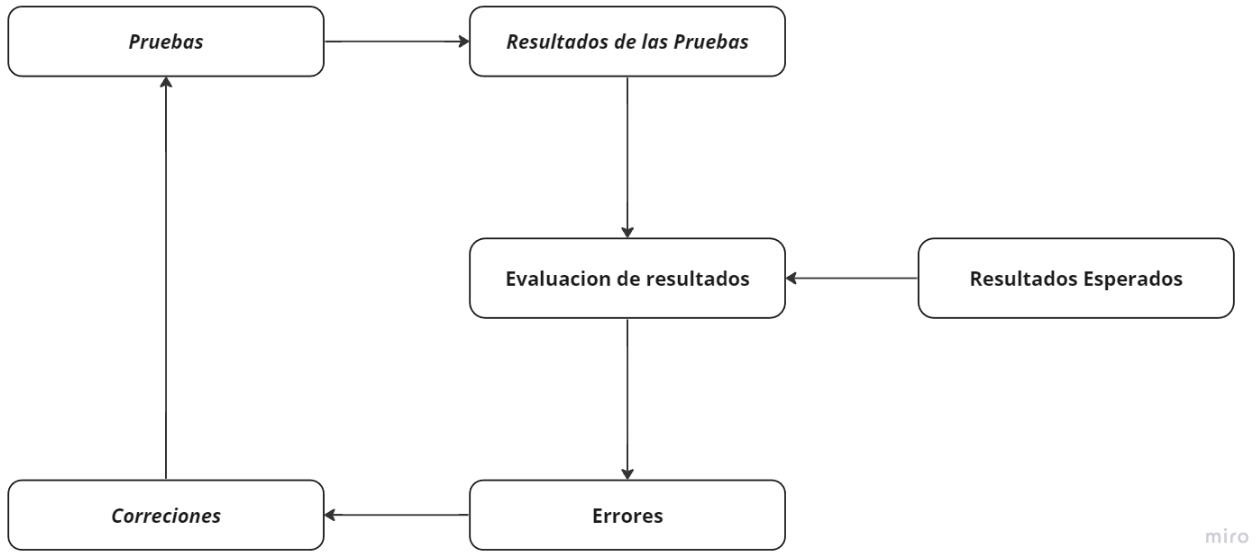
# 6 Pruebas

Las actividades de pruebas en el SI incluyen las: pruebas de unidad, pruebas de integración, pruebas de performance y pruebas de aceptación.

El equipo de SQA serán los encargados de ejecutar las pruebas, de anotar los resultados de las pruebas, así como analizar el resultado de las pruebas y de recomendar acciones correctivas si en los elementos probados se

encuentra algún defecto.

En el documento de pruebas se describe con más detalle el proceso de las pruebas. En la siguiente figura se muestra el proceso de las pruebas



## Reportar problemas y resolución

En esta sección se describen los procesos a seguir para reportar, dar seguimiento y resolver problemas en los productos de software y desarrollo

## Reporte del proceso de auditorias

SQA reporta los resultados del proceso de auditoría y provee recomendaciones, si es necesario debe usar el proceso de reporte de auditorías. El proceso de reporte de auditorías es usado para reportar lo siguiente:

* + - Se realiza un seguimiento y el proceso funciona
    - Se realiza un seguimiento, pero el proceso no funciona
    - No se realiza seguimiento

### Presentación y disposición del reporte del proceso de auditoria

El equipo de SQA le entregara dicho reporte a:

* + - 1. Al administrador del proyecto: Funciones:
         1. Para saber si los procesos de desarrollo son acatados y si estos son efectivos para cumplir con las metas del proyecto
         2. Para indicar su acuerdo, desacuerdo o aplazamiento de las recomendaciones mencionadas en el reporte de la auditoria

|  |
| --- |
| Reporte de procesos de auditoria |
| Numero de reporte: |
| Líder de auditoria: |
| Fecha de reporte: |
| Equipo de auditoria: | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre del proyecto: | | | | |
| Fecha de la auditoria: | | | | |
| Procesos/Procedimientos auditados: | | | | |
| Checklist usado(s): | | | | | | | |
| Resultados de la auditoria: (Seleccionar una)   * Procesos/Procedimientos aprobados * Procesos/Procedimientos condicionalmente aceptados Condiciones: * Procesos/Procedimientos no aceptados Condiciones: | | | | |
| Elemento (Ele) | | | | |
| ELE # | Titulo | Asignado a | Fecha de asignación | Fecha de fin |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Acción de corrección: | | | | |
| Estado:   * Aprobado * Cancelado * Aplazado | | | | |
| Administrador de proyecto: Fecha: | | | | |

### Procedimiento de escalamiento para la resolución de problemas de no acuerdo en el proceso del reporte de auditorias

Al encontrarse con un problema de calidad en algún elemento de trabajo ya sea documento, código o producto de software, primero se tratará con el creador de ese elemento, si entonces existen problemas, ya sea que el dueño de ese producto no quiera corregir el error, o problemas de entendimiento, entonces el equipo de SQA le notificara al administrador del proyecto para que este tome cartas en el asunto y de una solución al problema.

### Problemas de registro en código

Los problemas que se encuentren en el código de software o en la documentación que se encuentra bajo gestión de configuración deben registrarse mediante un P/CR (o STR, según corresponda al proyecto)

independientemente de cómo o quién haya descubierto el problema. Los P/CR generados por SGI se prepararán y procesarán de acuerdo con la referencia (f). SGI analizará los P/CR en

busca de tendencias de problemas en un esfuerzo por evitar discrepancias recurrentes.

## Reporte de evaluación

La tabla 7.2 provee el formato para el reporte de las herramientas de software.

## Reporte de la evaluación de instalaciones

La tabla 7.3 provee el formato para el reporte de instalaciones.

|  |
| --- |
| **Evaluación de herramienta de software** |
| SQA: Fecha: |
| Herramienta de software evaluada: |

|  |
| --- |
|  |
| Métodos o criterio usados en la evaluación: |
| Resultado de la evaluación: |
| Acciones correctivas recomendadas: |
| Acciones correctivas tomadas: |

|  |
| --- |
| **Evaluación de las instalaciones** |
| SQA: Fecha: |
| Instalación evaluada (Equipo, espacio) |

|  |
| --- |
|  |
| Métodos o criterio usados en la evaluación: |
| Resultado de la evaluación: |
| Acciones correctivas recomendadas: |
| Acciones correctivas tomadas: |

Herramientas

Las herramientas por usar en el desarrollo del producto son:

**Visual Studio Code:** Entorno de desarrollo Integrado que soporta varios lenguajes de programación como: PHP, Java, Python, C, CSS, C++, and C#, contiene control integrado de Git

**Visual Studio:** Entorno de desarrollo integrado que soporta varios lenguajes de desarrollo como: C++, C#, Fortran, Visual Basic, permite control de Git

**SQL Server:** Es un sistema de gestión de base de datos relacional. El lenguaje de desarrollo utilizado es Transact- SQL, utilizado para manipular y recuperar datos, crear tablas y definir relaciones entre ellas.

**Android Studio:** Es el entorno de desarrollo integrado oficial para la plataforma Android. Reemplazó a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android

**Oracle Datamodeler:** Permite a los usuarios crear, navegar y editar modelos lógicos, relacionales, físicos, multidimensionales y de tipo de datos. Proporciona capacidades de ingeniería directa e inversa y apoya el

desarrollo colaborativo a través del control de código fuente integrado. Puede ser utilizado tanto en entornos tradicionales como en la nube.

**Balsamiq:** Es una herramienta rápida wireframing que le ayuda a trabajar más rápido y más inteligente.

Reproduce la experiencia de dibujar en una pizarra, pero utilizando un ordenador. Hacer maquetas es rápido. Vas a generar más ideas, para que puedas tirar las malas y descubrir las mejores soluciones.

**Bootstrap:** Es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web.

**Postman:** Es una aplicación que permite probar API web. Permite resolver el problema de compartir pruebas API con otras personas.

**Git:** Es un software de control de versiones, pensando en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente.

**NodeJS:** Es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google.

**PowerShell:** Es una interfaz de consola con posibilidad de escritura y unión de comandos por medio de

instrucciones. Esta interfaz de consola está diseñada para su uso por parte de administradores de sistemas con el propósito de automatizar tareas o realizarlas de forma más controlada.

**Discord:** Es un servicio de mensajería instantánea y chat de voz VolP. En esta plataforma, los usuarios tienen la capacidad de comunicarse por llamadas de voz, videollamadas, mensajes de texto, o con archivos y contenido multimedia en conversaciones privadas o como parte de comunidades llamadas servidores.

**Jira:** Es un software para la gestión de proyectos, seguimiento de errores e incidencias, es increíblemente flexible y se puede personalizar para trabajar con el flujo de trabajo específico de tu equipo.

**Draw io:** Es un software de dibujo de gráficos multiplataforma. Su interfaz se puede utilizar para crear diagramas como diagramas de flujo, esquemas alámbricos, diagramas UML, organigramas y diagramas de red.

**Miro:** Es una plataforma de colaboración digital diseñada para facilitar la comunicación en equipo remota y distribuida, la gestión de proyectos y definir productos.

**Idera SQL Diagnostic Manager:** Es una poderosa solución de monitoreo y diagnóstico de rendimiento en tiempo real que alerta de manera predictiva sobre problemas del estado de salud, rendimiento o disponibilidad de su

entorno de SQL Sever a través de una consola central, en entornos físicos o virtuales y en la nube, también permite una documentación rápida sobre la base de datos.

## Control de código

En la sección 5.2 se discuten los estándares que debe tener el código generado durante el desarrollo del proyecto.

A continuación, se listan un grupo de tareas a ser realizadas, la descripción más detallada del control de versiones del código se encuentra en el documento de administración de configuración.

1. Identificar, etiquetar y clasificar el software a ser controlado.
2. Identificar la ubicación física del software a ser controlado.
3. Identificar el lugar donde se encuentran las copias de seguridad.
4. Distribución de copias de código.
5. Identificar los documentos afectados por el cambio de código.
6. Establecimiento de versiones nuevas.
7. Administrar los usuarios que tienen acceso al código.

La tarea del equipo SGI es comprobar que estos procesos se están llevando a acabó de forma correcta por el equipo de administración de la configuración.

## Sección 10. Entrenamiento

En la siguiente tabla se listan las capacitaciones necesarias para el equipo de desarrollo.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Capacitación | Descripción | Dirigido  a/impartido por | Fecha | |
| Inicio | Fin |
| Introducción al SGI | Introducción rápida de lo tratado en el desarrollo del proyecto SGI:  requerimientos, casos de uso,  técnicas y cualquier otra información que resulte  relevante para el | Dirigido a:  Brayan Cardenas Diana Yepes Valeria Ruiz  Fabian Diaz  Impartido por: Isaura Suarez | **2023-05-15** | **2023-05-17** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | desarrollo del  proyecto. |  |  |  |
| Introducción a  .NET | Introducción al desarrollo de APIs  en C# utilizando el framework de  Microsoft ASP.NET Core. | Dirigido a:  Brayan Cardenas Diana Yepes Valeria Ruiz  Fabian Diaz  Impartido por: Victor Vladimir | **2023-09-02** | **2023-09-08** |
| Introducción a React | Introducción al uso de React para el desarrollo de  aplicativo web haciendo uso de JS y CSS. | Dirigido a:  Brayan Cardenas Diana Yepes Valeria Ruiz  Fabian Diaz  Impartido por: Helbert Rojas | **2023-10-30** | **2023-11-10** |
| Introducción a React native | Introducción al uso de React native para el desarrollo de aplicativo móvil utilizando JS, CSS y aplicaciones de  ejecución móvil como expo go y  Android Studio. | Dirigido a:  Brayan Cardenas Diana Yepes Valeria Ruiz  Fabian Diaz  Impartido por: Helbert Rojas | **2023-10-30** | **2023-11-10** |

## Sección 11. Control de medios

El control de medios incluye los siguientes elementos:

1. Copia de seguridad programada regularmente
2. Medios etiquetados e inventariados archivados en un área de almacenamiento de acuerdo con los requisitos de seguridad
3. Protección adecuada contra ingreso no autorizado

SQA llevará a cabo evaluaciones continuas del proceso de control de medios de software para verificar que el proceso de control de medios de software sea eficaz

## Sección 12. Control de proveedores

Antes de cualquier adquisición de software para respaldar el esfuerzo de desarrollo, SGI y los miembros del proyecto definirán y proporcionarán los requisitos completos por medio de un proceso de evaluación de las herramientas que dicho software nos brinda y las opciones de adquisición óptimas para el desarrollo.

Parte del proceso de evaluación requerirá que el proveedor o vendedor describa su soporte técnico, el manejo de las preguntas, problemas de los usuarios, las actualizaciones del producto de software y las diferentes limitaciones que presentaran los distintos softwares según si se opciones por medio de pagos normales o por medio de llaves contempladas en medidas educativas o de apoyo a estudiantes.

## Recopilación, mantenimiento y retención de registros

**Gestión de riesgos**

El Proyecto Jucar Autopartes SAS ha desarrollado un plan de gestión de riesgos. SQA revisará y evaluará el análisis técnico de riesgos y cualquier plan de reducción de riesgos. Los informes de SQA confirmarán que los riesgos

identificados se gestionan de acuerdo con las disposiciones de los planes de gestión de riesgos del proyecto, y que los elementos de acción asociados se informan, gestionan y siguen hasta el cierre.

## Riesgos Ocurridos Durante el desarrollo

* Se debe aclarar que uno de los integrantes se retiró del proyecto en mitad del desarrollo

## Posibles Riesgos

* Perdida de código
* Mal versionamiento y posible pérdida de versiones
* Mal manejo de la Base de Datos y posible pérdida de datos importantes
* Cambios en los requisitos
* Tecnología obsoleta
* Falta de comunicación entre el equipo
* Dependencia de terceros
* Falta de experiencia
* Mala gestión del proyecto
* Software obsoleto

## Lista de acrónimos

|  |  |
| --- | --- |
| SCM | Software Configuration Management |
| SQA | Software Quality Assurance |
| DTO | Data Transfer Object |
| SI | Sistema Inventarios |
| E/S | Entrada/Salida |
| VoIP | Voz sobre protocolo de internet |
| SDP | Software Development Plan |
| P/CR | Informe del problema/Cambio |
| B2B | Business-to-business |

Checklist

|  |
| --- |
| **Checklist para el Proceso de Auditoría a la Planeación** |
| **Proyecto: Fecha:**  **Preparado por:** |
| **Procedimientos:**   * Los planes del proyecto existen y están documentados en el SDP. * Los estándares están documentados y están presentes en el SDP. * El contenido del SDP tiene consistencia en la implementación de los estándares de procesos de software. |

|  |
| --- |
| * El SDP está bajo la administración de la configuración. * Los requerimientos del software son la base de los planes, productos de trabajo y las actividades * Existe el SCM ya sea en el SDP o en un documento aparte * El plan de SCM está bajo la protección de la configuración. * El plan de SQA existe. * Existen planes para las pruebas integración de software. |

*Ilustración 11.1: Checklist para el proceso de auditoría a la planeación*

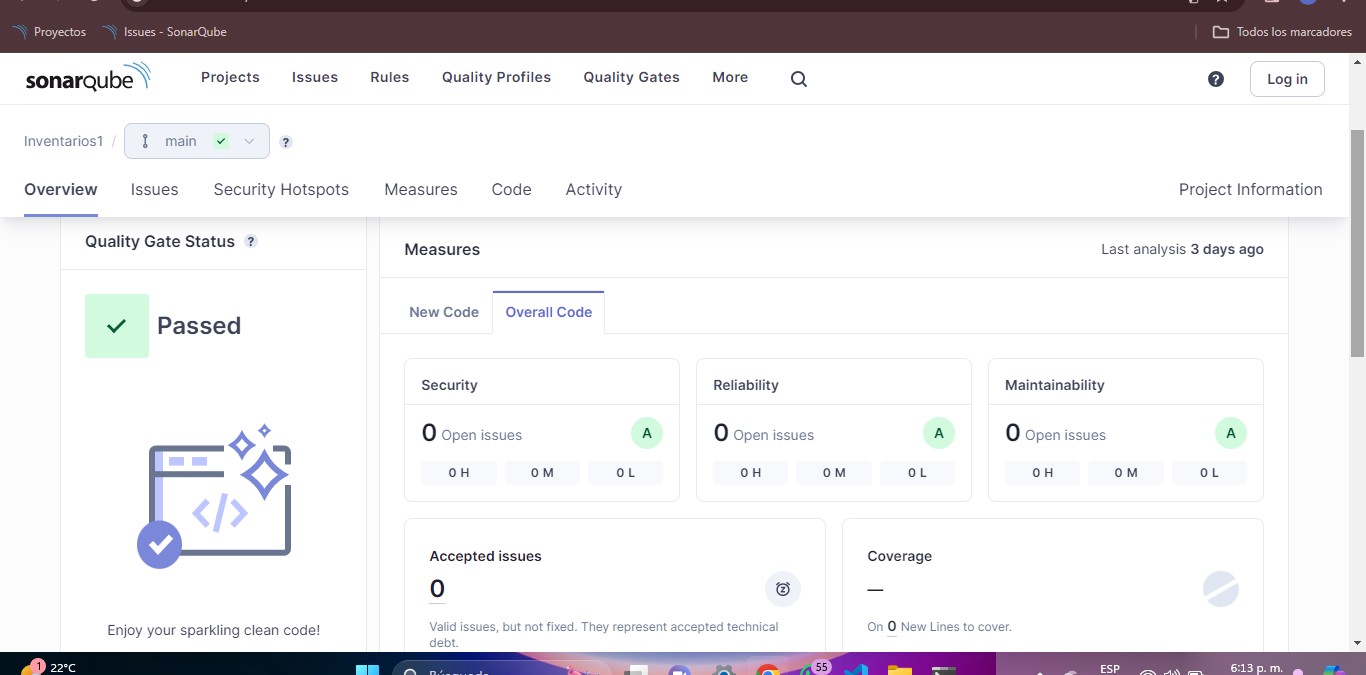
|  |
| --- |
| **Reporte de Análisis del Proceso de Auditoria a los Requisitos** |
| **Proyecto: Fecha:**  **Preparado por:** |
| **Procedimientos:**  **Parte 1. Requerimientos de software**   * Los requerimientos de software están documentados e incluyen la matriz de rastreo. * El SRS está bajo la administración de la configuración * Los planes de desarrollo de software y otros productos son modificados si se modifica el SRS para que estos sean consistentes. * Se han usado mediciones para determinar el estado de la administración de requisitos. |

*Ilustración 11.2: Reporte de análisis del proceso*

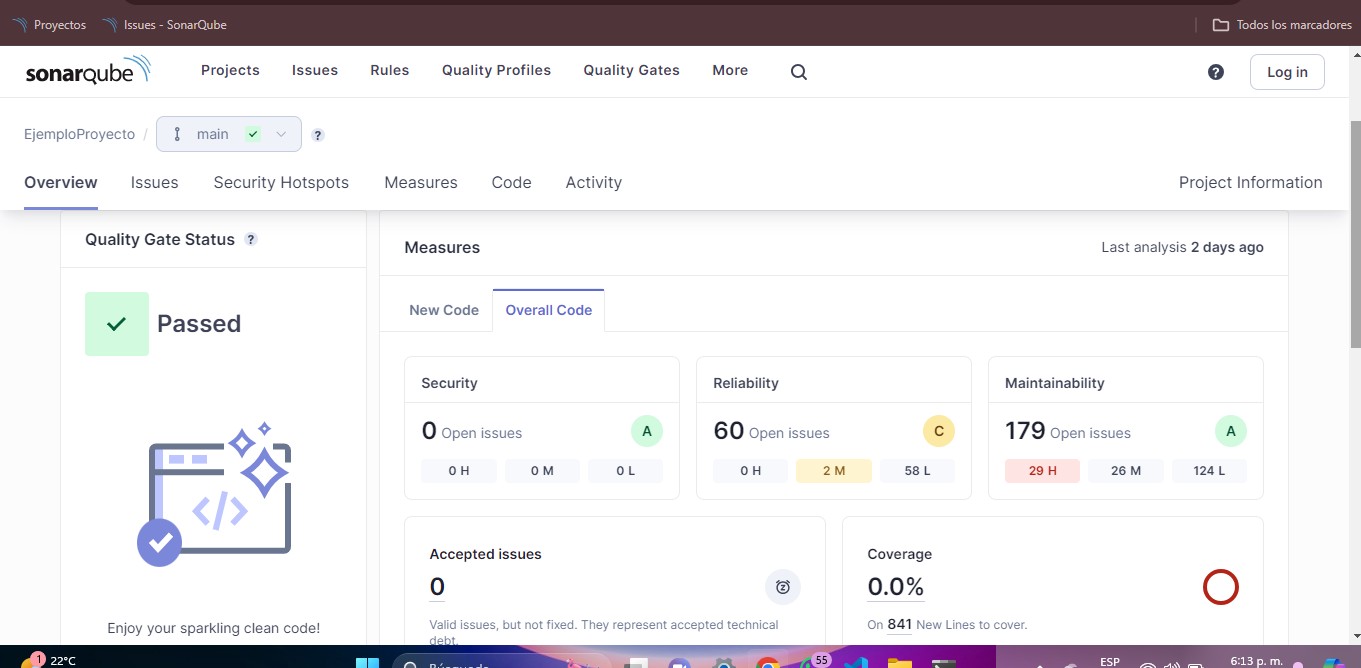
|  |
| --- |
| **Checklist para el Proceso de Auditoría al Diseño de Software** |
| **Proyecto: Fecha:**  **Preparado por:** |
| **Procedimientos:**  **Parte 1. Diseño**   * Los documentos de diseño y la matriz de rastreo estas listas. * Las caminatas evalúan si todos los requerimientos están plasmados en el diseño, así como el de tratar de encontrar errores. * El diseño esta actualizado con respecto a los últimos cambios en los requisitos. * Los cambios en el diseño seguidas, evaluadas. * El diseño de software es consistente con la metodología de diseño propuesta en el SDP. |

*Ilustración 11.3: Checklist para el proceso de auditoría a la planeación*

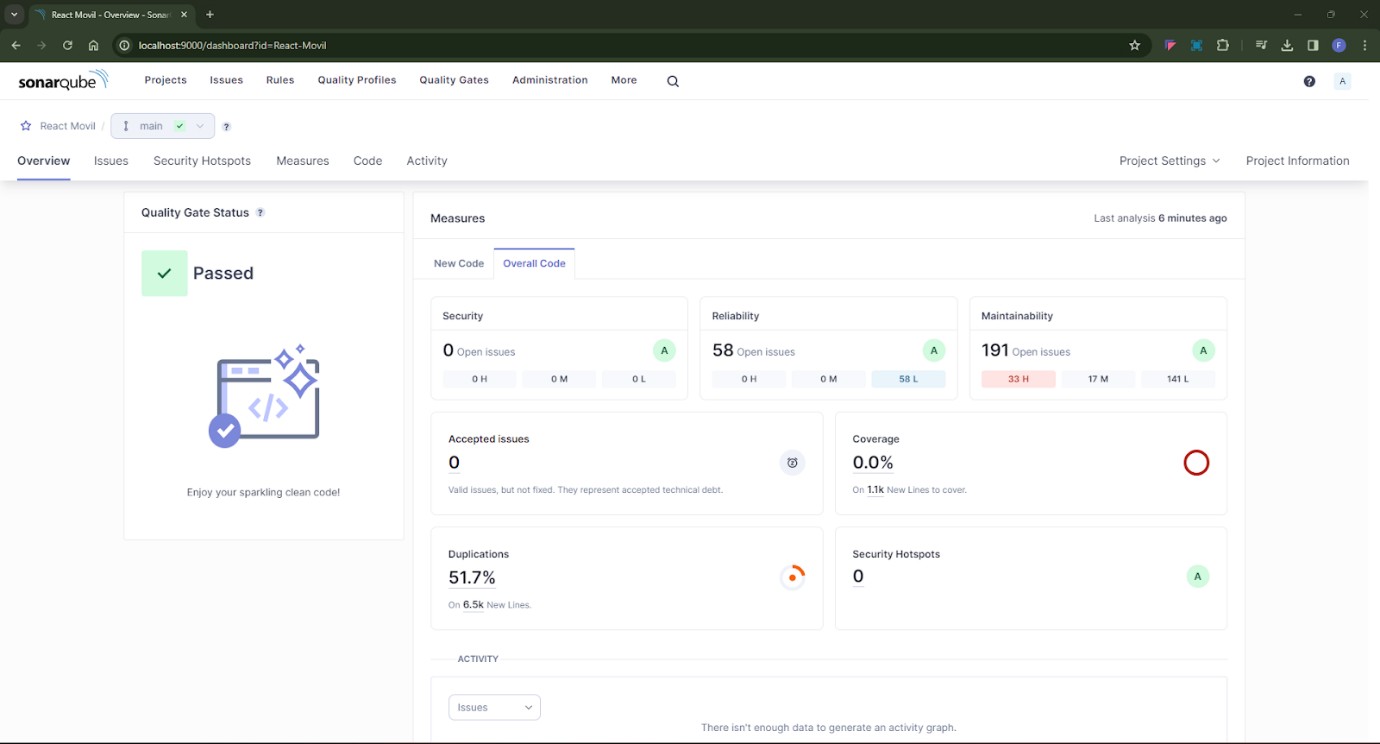
Anexos API



FRONT END -WEB



FRONT END - APP MÓVIL



Pruebas de Usabilidad

Pruebas De Usabilidad Administrador:

https://docs.google.com/document/d/1B5M-OCWEGVs4edZUvPH8ZG-Gy8N- DkbZ/edit?usp=sharing&ouid=112510879730265058597&rtpof=true&sd=true

Pruebas de usabilidad Bodega:

https://docs.google.com/document/d/1YAdNz8ZwMuIVza-JKa19t6B-nl9mgtWlIjxtveqfCeU/edit?usp=sharing

Pruebas de usabilidad- Caja:

https://docs.google.com/document/d/1Dj8Bzn3GgN9E3eiD\_CQ\_JbmKR0MBfDgxR0lPuR0Xs8M/edit?usp=sharing