

Parkly

PLAN DE CALIDAD

Tabla de Contenidos

1	OBJETIVOS	4
2	INTRODUCCIÓN	4
2.1	Objetivos de SQA	5
2.2	El Rol de SQA	5
2.3	Responsabilidades de SQA	5
2.4	Funciones de SQA	5
3	DOCUMENTOS RELACIONADOS	6
4	DESTINATARIOS	6
5	ADMINISTRACIÓN	7
5.1	Organización	7
5.2	Responsabilidades	7
5.3	Análisis de Riesgos	8
6	ESTÁNDARES, PRÁCTICAS, CONVENCIONES Y MEDICIONES	9
6.1	Estándares	9
6.2	Templates	9
6.2.1	Planificación de Proyectos	9
6.2.2	Control y Seguimiento de Proyectos	9
6.2.3	Administración de Requerimientos	9
6.2.4	Administración de la Subcontratación de Software	9
6.2.5	Administración de la Configuración de Software	10
6.2.6	Aseguramiento de Calidad de Software	10
6.3	Checklists	10
6.3.1	Planificación de Proyectos	10
6.3.2	Control y Seguimiento de Proyectos	10
6.3.3	Administración de Requerimientos	10
6.3.4	Administración de la Subcontratación de Software	10
6.3.5	Administración de la Configuración de Software	11
6.3.6	Aseguramiento de Calidad de Software	11
6.4	Procesos, Guías y Procedimientos	11
6.4.1	Planificación de Proyectos	11
6.4.2	Control y Seguimiento de Proyectos	11
6.4.3	Administración de Requerimientos	11
6.4.4	Administración de la Subcontratación de Software	12
6.4.5	Administración de la Configuración de Software	12
6.4.6	Aseguramiento de Calidad de Software	12
6.5	Mediciones	12
7	TAREAS DE SQA	13
7.1	Tareas de Infraestructura	13
7.2	Productos bajo Control de Calidad	13
7.3	Procesos bajo Control de Calidad	15
7.4	Cronograma de Actividades de SQA	16
8	SOPORTE DE SQA A CLIENTES, PROVEEDORES Y SUBCONTRATISTAS	17
9	REGISTROS Y REPORTES DE SQA	18
9.1	Registro de SQA	18

PLAN DE CALIDAD

9.2 Reporte de SQA	20
10 REPORTES DE PROBLEMAS Y ACCIONES CORRECTIVAS	21
11 APÉNDICES	22
11.1 Glosario	22
11.2 Historia de Cambios	22

{

NOTA PARA EL RESPONSABLE DE SQA

Este estándar de Plan de SQA (SQAP) especifica los contenidos mínimos de un Plan de SQA para un proyecto de desarrollo.

Debajo de cada sección encontrará, encerradas entre llaves ({ , }) y en color azul, las guías para completar el contenido de cada una de ellas. Borre estas guías cuando lo crea necesario.

Recuerde modificar las propiedades del documento con el nombre del proyecto, y modificar el fondo de las páginas con la leyenda "confidencial" y "preliminar" de acuerdo a su criterio. Para esto último debe editar el encabezado/pie de página y seleccionar el recuadro de texto ubicado en la zona media de la página para editar sus contenidos o eliminarlo.

}

1 OBJETIVOS

El objetivo del Plan de Aseguramiento de la Calidad (SQA) para Parkly es garantizar que tanto el producto como el proceso de desarrollo cumplan con los criterios de calidad establecidos y que el sistema final sea confiable, seguro, eficiente y apto para su operación en entornos reales.

Este plan está alineado con los requisitos documentados en el ERS-830 IEEE del proyecto Parkly, donde se describen las funcionalidades, restricciones, características de los usuarios y el alcance del sistema. Toda la referencia técnica y conceptual se encuentra detallada en el documento ERS correspondiente.

A continuación se presentan los objetivos de calidad del producto y del proceso, expresados en términos cuantificables:

1. **Calidad de Software:** Garantizar que el 100% de los requisitos funcionales y no funcionales establecidos en el ERS estén implementados y verificados.
2. **Calidad de los Componentes IoT:** Verificar que el 100% de los sensores infrarrojos instalados funcionen correctamente durante las pruebas de campo.
3. **Calidad del Proceso de Desarrollo:** Realizar revisión y aprobación del 100% de la documentación (ERS, diseño, manuales, diagramas), así como realizar revisiones de código con un 80% de cobertura mínima por parte del equipo técnico.
4. **Calidad Operativa en la Aplicación Móvil:** Garantizar que la aplicación tenga una disponibilidad mínima del 99% durante pruebas internas.
5. **Seguridad del Sistema:** Cumplir al 100% con las prácticas de manejo seguro de datos sensibles (usuarios, pagos, sensores).
6. **Calidad en la Instalación de Hardware:** Asegurar que el 95% de los sensores mantengan mediciones estables y sin lecturas erróneas durante períodos de prueba prolongados.

2 INTRODUCCIÓN

El Aseguramiento de la Calidad del Software (SQA, por sus siglas en inglés) es una disciplina fundamental dentro del proceso de desarrollo de Parkly, ya que garantiza que cada fase del proyecto se realice siguiendo estándares definidos y con un enfoque orientado a la calidad, confiabilidad y eficiencia del sistema final.

La función principal del SQA es proporcionar confianza a los responsables del proyecto, a los clientes y a los futuros usuarios, asegurando que el producto se construya correctamente y cumpla con los niveles de calidad esperados. Su enfoque no se limita al software únicamente, sino que también abarca componentes esenciales del proyecto, como la instalación y funcionamiento del hardware eléctrico y sensores IoT, que forman parte crítica del sistema Parkly.

PLAN DE CALIDAD

La aplicación del SQA permite asegurar que:

- Se emplea una metodología de desarrollo adecuada, basada en buenas prácticas de ingeniería de software.
- El equipo sigue estándares, procesos y procedimientos formales, lo que previene errores y asegura consistencia en el trabajo.
- Se llevan a cabo revisiones y auditorías independientes, garantizando objetividad en la evaluación del producto y del proceso.
- La documentación necesaria (requisitos, diseño, pruebas, manuales) se genera de manera oportuna, permitiendo su uso para mantenimiento futuro.
- Existen mecanismos, y son usados, para controlar los cambios.
- El testing se enfatiza en las áreas de producto de alto riesgo.
- Cada tarea en el proceso de desarrollo se completa satisfactoriamente antes de que la siguiente comience.
- Las desviaciones de los estándares y procedimientos se exponen lo antes posible.
- El proyecto es auditável por profesionales externos en cualquier momento.

- La tarea de control de calidad se ejecuta siguiendo estándares previamente establecidos.
- El plan de SQA y el plan de desarrollo son compatibles.

2.1 Objetivos de SQA

Los principales objetivos del Aseguramiento de la Calidad del Software son los siguientes:

- Mejorar la calidad del software monitoreando apropiadamente tanto los productos de software como el proceso de desarrollo que los genera.
- Asegurar el cumplimiento de los estándares y procedimientos establecidos para el software y el proceso de software establecidos.
- Asegurar que cualquier desviación en el producto, el proceso, o los estándares son elevados a la gerencia para poder resolverlas.

2.2 El Rol de SQA

Las personas responsables del proyecto de software (desarrollo y cliente) son las únicas que pueden ser responsables por la calidad. El rol de SQA es monitorear la manera en que estos grupos ejecutan sus responsabilidades. Por lo tanto existen los siguientes peligros latentes:

- Es un error asumir que el personal de SQA puede por sí solo hacer algo por la calidad del proyecto.
- La existencia de una función de SQA no asegura que se siguen los estándares y los procedimientos.
- Sólo si la gerencia demuestra periódicamente su soporte a SQA, siguiendo sus recomendaciones, SQA podrá ser efectiva.
- A menos que la gerencia de línea requiera que SQA trate de resolver sus no-conformidades con la gerencia del proyecto antes de elevarlas, SQA y desarrollo no trabajarán efectivamente.

Todo lo que puede hacer SQA es alertar a la gerencia sobre las desviaciones a los estándares y procedimientos establecidos. La gerencia debe entonces insistir acerca de que los problemas de calidad se solucionen antes de que el producto sea liberado para su uso, sino SQA se transforma en un ejercicio burocrático y costoso.

2.3 Responsabilidades de SQA

Las principales responsabilidades del rol de SQA son las siguientes:

- Verificar la completitud en los planes de desarrollo y de calidad del proyecto.
- Participar como moderador en inspecciones de diseño, de código u otros productos.
- Revisar los planes de testing verificando el cumplimiento de los estándares.
- Revisar una muestra significativa de los resultados del testing para determinar el cumplimiento de los planes.
- Auditarse periódicamente la performance de SCM para determinar el cumplimiento de los estándares
- Participar en todas las revisiones a fin de cada fase del proyecto y registrar formalmente si los estándares y procedimientos no se alcanzaron satisfactoriamente.

2.4 Funciones de SQA

Las principales funciones del rol de SQA, a través de todo el ciclo de vida, son las siguientes:

PLAN DE CALIDAD

- Prácticas de QA: se definen y están disponibles herramientas, técnicas, métodos y estándares de desarrollo adecuados para ser usados como estándares de las revisiones de QA.
- Evaluación de la planificación del proyecto de software: si no se planifican prácticas de calidad adecuadas desde el inicio y sincronizadas con el plan del proyecto, luego no serán implementadas.
- Evaluación de los requerimientos: como es extremadamente inusual que se desarrolle productos de alta calidad a partir de requerimientos de baja calidad, los requerimientos iniciales deben ser revisados contra los estándares de calidad establecidos.
- Evaluación del proceso de diseño: se definen los medios para asegurar que el diseño sigue las metodologías planificadas, que implementa los requerimientos y que la calidad del diseño propiamente dicha es revisada independientemente.
- Evaluación de las prácticas de codificación: las prácticas apropiadas de codificación deben ser establecidas y usadas.
- Evaluación del proceso de integración y testeo del software: se establece un programa de testing de calidad, el testing es ejecutado por un grupo independiente que es tanto capaz como está motivado para encontrar problemas, la planificación del testing comienza en las primeras etapas del proyecto, y se revisa la calidad del testing propiamente dicho.
- Evaluación del uso del proceso de control y gerenciamiento del proyecto: asegurando que los procesos de gerenciamiento están funcionando, SQA ayuda a garantizar que todo el grupo de proyecto está orientado a producir resultados de calidad.
- Adaptación de los procedimientos de SQA: El plan de SQA debe ser adaptado a las necesidades específicas del proyecto.

3 DOCUMENTOS RELACIONADOS

{

Indique todos los documentos utilizados para construir este plan, y los que son referenciados en alguna parte del mismo. (Ej.: Plan de Proyecto, Plan de SCM, Estándar de SQAP, estándares, guías y procedimientos de la organización utilizados por SQA, Proceso de Administración de Riesgos, etc.).

}

- Definición del Proceso de SQA
- Template de Plan de SQA
- Estudio de Viabilidad
- Cronograma del Proyecto
- Plan de SCM

4 DESTINATARIOS

El presente documento está destinado a las siguientes personas:

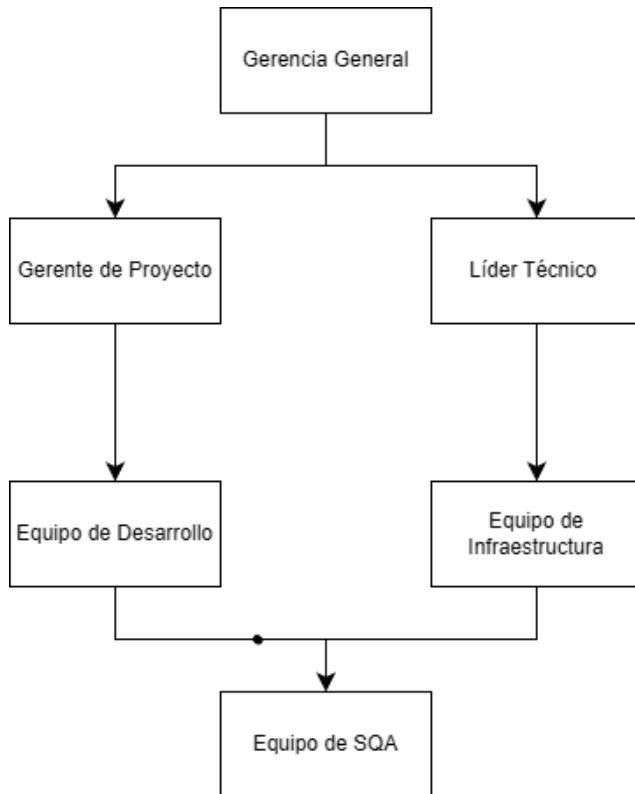
Lector	Sector o Rol
Gerente de Proyecto	Dirección y toma de decisiones estratégicas
Líder Técnico	Coordinación del desarrollo y arquitectura del sistema
Equipo de Desarrollo	Programadores, diseñadores y responsables del código
Equipo de SQA	Encargados de revisión, auditoría y aseguramiento de calidad
Equipo de Infraestructura	Encargados del hosting, circuitos y hardware del sistema

5 ADMINISTRACIÓN

Esta sección del Plan de SQA describe aspectos relacionados con el management del equipo de SQA del proyecto. Se describen la organización del equipo de SQA, los roles, responsabilidades y tareas, el cronograma de actividades y los riesgos que pueden amenazar los objetivos de este plan.

5.1 Organización

A continuación se describe la estructura organizativa del proyecto Parkly y la ubicación del equipo de SQA dentro del flujo general. El equipo de SQA interactúa de manera transversal con todas las áreas, verificando que cada fase del proyecto cumpla con estándares, procedimientos y métricas de calidad previamente establecidos.



- **Gerencia General:** Toma decisiones estratégicas y aprueba el avance del proyecto.
- **Gerente de Proyecto:** Coordina actividades, asigna tareas y da seguimiento al cumplimiento del plan.
- **Líder Técnico:** Supervisa el desarrollo, arquitectura, integración con circuitos IoT y decisiones técnicas.
- **Equipo de Desarrollo:** Implementa la funcionalidad del sistema Parkly.
- **Equipo de Infraestructura:** Administra servidores, redes, sensores, cámaras y hardware del sistema.
- **Equipo de SQA:**
 - Trabaja de forma transversal con todos los equipos.
 - Revisa documentación, código, pruebas e implementación.
 - Garantiza que se sigan estándares, lineamientos técnicos y buenas prácticas.
 - Reporta incumplimientos o riesgos directamente al Gerente de Proyecto.

PLAN DE CALIDAD**5.2 Responsabilidades**

Rol	Responsabilidades
Responsable de Calidad (RQA/QC)	<ul style="list-style-type: none"> • Asegurar que los objetivos de calidad del proyecto se cumplan. • Planificar las actividades de SQA. • Obtener los recursos para la ejecución del Plan de SQA. • Realizar el monitoreo y el seguimiento de las actividades planificadas de SQA. • Informar a los Miembros del Equipo de proyecto y al Líder de Proyecto acerca de los resultados de las actividades de SQA. • Reportar periódicamente al Líder de Proyecto y a los Gerentes Ejecutivos sobre las actividades de SQA. • Elevar los casos de no-conformidad en la calidad de los productos y en la ejecución de los procesos a la Gerencia Ejecutiva. • Proveer al Líder de Proyecto elementos para el análisis del origen de los problemas cuando se detectan diferencias entre los resultados esperados y reales de las mediciones de Calidad.
Miembro de Calidad (MQA/QC)	<ul style="list-style-type: none"> • Construir los Activos de Proceso que los Miembros del Equipo de Proyecto deben utilizar para construir los productos. • Ejecutar las tareas definidas dentro del Plan de SQA. • Realizar auditorías sobre los procesos. • Realizar revisiones a los productos para verificar su conformidad a los estándares. • Coordinar y facilitar revisiones.
Líder de Proyecto (LP)	<ul style="list-style-type: none"> • Definir junto con el Responsable de Calidad los objetivos de calidad del proyecto en el Plan de SQA. • Realizar ajustes en el proyecto para cumplir con el proceso definido, según las no-conformidades detectadas por el equipo de Calidad. • Compensar los desvíos detectados ó ajustar el Plan.
Gerente Ejecutivo (GE)	<ul style="list-style-type: none"> • Proveer y garantizar los recursos para el desempeño del rol de Calidad. • Definir los casos de no-cumplimiento no resueltos en el ámbito del proyecto. • Realizar revisiones periódicas de las actividades y los resultados de SQA.
Equipo Técnico (ET)	<ul style="list-style-type: none"> • Realizar los ajustes necesarios y/o rehacer el trabajo para cumplir con el producto definido.
Equipo de Infraestructura	<ul style="list-style-type: none"> • Asegurar el correcto funcionamiento del entorno técnico para pruebas e implementación.

5.3 Análisis de Riesgos

De acuerdo con la estructura metodológica del proyecto, el Análisis de Riesgos completo se encuentra descrito en el **Estudio de Viabilidad**, donde se evalúan los riesgos técnicos, operativos, económicos y de gestión asociados al desarrollo e implementación de Parkly.

6 ESTÁNDARES, PRÁCTICAS, CONVENCIONES Y MEDICIONES

El propósito de esta sección es definir los estándares, prácticas, convenciones y mediciones utilizadas para lograr los objetivos definidos en este Plan.

6.1 Estándares

Para garantizar la consistencia, calidad y mantenibilidad del software desarrollado para **PARKLY**, se adoptarán los siguientes estándares. Cada uno funciona como una referencia formal que guiará las actividades de análisis, diseño, implementación, documentación y pruebas del sistema.

Estándares de Documentación

- **Estándar de Especificación de Requerimientos (ERS-830 IEEE):** IEEE 830 – Software Requirements Specification (SRS).
- **Estándar de Especificación de Arquitectura:** IEEE 42010 – Systems and Software Architecture Description.
- **Estándar de Diseño Detallado:** IEEE 1016 – Software Design Description (SDD).

Estándares de Modelado

- **UML 2.5 para modelado de casos de uso, clases y secuencias:** OMG UML 2.5 Specification.
- **Estándar de Modelado de Datos:**
 - ISO/IEC 11179 – Metadata Registry (MDR)
 - ISO/IEC 19501 – UML para modelado estructural de datos
 - Normas de Normalización (1FN, 2FN, 3FN, BCNF)
- **Estándar de Modelado de Procesos:**
 - OMG BPMN 2.0 Specification
 - UML Activity Diagrams (OMG UML 2.5)

Estándares de Nomenclatura

- **Nomenclatura para componentes del sistema:**
 - Uso de nombres descriptivos en inglés, formato *UpperCamelCase* para clases y *lowerCamelCase* para atributos y métodos.
- **Nomenclatura para bases de datos:**
 - Tablas en singular, en mayúsculas.
 - Campos en minúsculas con guion bajo.

Estándares de Código Fuente

- **Guías de estilo para Java y JavaScript/TypeScript:** Basado en Google Java Style Guide y Airbnb JavaScript Style Guide.
- **Control de versiones:** Uso obligatorio de Git con flujo GitFlow.
- **Buenas prácticas de programación segura:** Validación de datos, manejo de excepciones, uso responsable de APIs, protección de credenciales y tokens.

Estándares de Pruebas

- **Estándar de Casos de Prueba:** Cada prueba debe incluir: ID, nombre, precondiciones, entradas, pasos, resultados esperados y obtenidos.
- **Estándar de Pruebas de Integración y Sistema:** Deben documentarse conforme a la estructura definida por IEEE 829.
- **Cobertura mínima:** 80% de pruebas unitarias para módulos críticos (sensores, pagos, disponibilidad de cajones).

Estándares de Seguridad

- **OWASP Top 10:** Como base para evitar vulnerabilidades en la aplicación móvil, API y panel administrativo.
- **Manejo seguro de datos sensibles:** Cifrado TLS para comunicación, hash seguro para contraseñas y políticas de acceso basadas en roles.

6.2 Templates

{

Definir los templates definidos para el proyecto. Estos templates usualmente serán documentos independientes referenciados en esta sección.

Ejemplos de templates a utilizar pueden ser:

- Template de Documentos del Proyecto
- Template de Minuta de Reunión
- Template de Informe de Avance
- Template de Manuales de Usuario
- etc.

}

6.2.1 Planificación de Proyectos

- Template de Plan de Proyecto
Incluye objetivos, alcance, entregables, cronograma, riesgos y recursos.
- Template de Cronogramas
Basado en la estructura de tareas WBS del proyecto.
- Template de conteo de PF
Utilizado para la estimación formal del tamaño del software.

6.2.2 Control y Seguimiento de Proyectos

- Template de Informe de Avance de Proyectos.
Incluye estado, hitos, riesgos, métricas y desvíos.

6.2.3 Administración de Requerimientos

- Template de Especificación de Requerimientos de Software.
Basado en IEEE 830; define requisitos funcionales, no funcionales y reglas del negocio.

6.2.4 Administración de la Subcontratación de Software

- Template de Plan de Adquisición.
Para planear compras, servicios externos y criterios de evaluación.
- Template de Solicitud de Propuesta.
Para proveedores candidatos.
- Template de Encuesta a Cliente del Proveedor.
Para validar antecedentes y referencias.
- Template de Encuesta a Proveedor Candidato.
Para evaluar experiencia y capacidades.

PLAN DE CALIDAD

- Checklist de Análisis de Riesgos.
Matriz de riesgos orientada a subcontratación.
- Template de Evaluación de productos enlatados.
Para soluciones preexistentes comparadas.
- Template de Matriz de Evaluación.
Para ponderación cuantitativa de alternativas.
- Template de Informe de Evaluación Proveedor – Producto.
Documento final de recomendación.
- Modelos de Contrato.
Plantillas formales para contratación aprobadas por el área legal.

6.2.5 Administración de la Configuración de Software

- Template de Plan de SCM.
Define repositorio, lineamientos de versionado, integración, release y control de cambios.

6.2.6 Aseguramiento de Calidad de Software

- Template de Checklist.
Para revisiones de documentos, procesos y código.
- Template de Registro de SQA.
Para documentar hallazgos, desviaciones y conformidades.
- Template de Reporte de Actividades de SQA.
Resumen periódico de auditorías, revisiones y métricas.
- Template de Plan de SQA.
Documento base para todos los proyectos que implementen el proceso de SQA.

6.3 Checklists

{

Definir los checklists que serán utilizados en el proyecto. Estos checklists serán utilizados durante las revisiones a lo largo del proyecto. Dado que las inspecciones necesitan de checklists para ser eficaces, debe haber un checklist para cada clase de documento sobre el cual se planea realizar inspecciones. Usualmente existe un checklist para cada estándar del proyecto. Los checklists usualmente son items de configuración independientes de este plan, en cuyo caso deben incluirse las referencias a ellos en esta sección.

Ejemplos de Checklists:

- Checklist para Plan de Proyecto
- Checklist para Especificación de Requerimientos
- Checklist para Especificación de Diseño Detallado
- Checklist para Código Fuente
- etc. (ver ejemplos de estándares)

}

6.3.1 Planificación de Proyectos

- Checklist del Plan de Proyecto.
 - Verificar que se incluye alcance, objetivos y entregables del proyecto.
 - Confirmar que el cronograma está definido con fechas y responsables.
 - Asegurar que se documentan los riesgos, su análisis y plan de respuesta.
 - Validar que el plan contiene recursos asignados y estimaciones completas.
 - Confirmar que se aplica el template oficial de planificación.

- Checklist del Cronograma.
 - Verificar que el cronograma tiene estructura WBS.
 - Asegurar que las dependencias entre tareas están correctamente definidas.
 - Confirmar que se asignan responsables por actividad.
 - Validar que se incluyen los hitos del proyecto.
 - Revisar que se actualice conforme al avance real.

6.3.2 Control y Seguimiento de Proyectos

- Checklist de Revisión de ERS
 - Confirmar que los requisitos cumplen el estándar IEEE 830/29148.
 - Revisar que los requisitos son medibles, verificables y trazables.
 - Verificar que se clasifican en funcionales y no funcionales.
 - Validar que existe matriz de trazabilidad.
 - Revisar que todos los términos relevantes están definidos en el glosario.

6.3.3 Administración de Requerimientos

- Checklist de Especificación de Requerimientos de Software.
 - Verificar que los requisitos están completos y sin ambigüedades.
 - Validar que cada requisito posee identificador único y criterio de aceptación.
 - Confirmar que los requisitos funcionales y no funcionales están claramente diferenciados.
 - Revisar que los requisitos estén alineados con los objetivos del sistema.
 - Confirmar que se cumple el template oficial del proyecto y el estándar IEEE 830.
 - Verificar consistencia entre requisitos y casos de uso.

6.3.4 Administración de la Subcontratación de Software

- Checklist de Aceptación de Producto.
 - Verificar que el producto entregado coincide con los criterios de aceptación.
 - Validar cumplimiento de los requisitos funcionales y técnicos acordados.
 - Confirmar entrega de documentación, licencias y manuales requeridos.
 - Revisar que se ejecutaron pruebas de aceptación y que fueron aprobadas.
- Checklist de Estrategia de Contratación.
 - Verificar definición de alcance de la contratación.
 - Confirmar criterios de evaluación de proveedores.
 - Validar requerimientos técnicos mínimos para el proveedor.
 - Revisar claridad en plazos, entregables y condiciones comerciales.
- Checklist de Evaluación del Proveedor.
 - Verificar experiencia previa relacionada con el tipo de proyecto.
 - Revisar referencias de clientes anteriores.
 - Confirmar solvencia técnica y capacidad de entrega.
 - Evaluar cumplimiento de normativas y certificaciones relevantes.
- Checklist de Monitoreo de proveedores.
 - Comprobar cumplimiento de hitos contractuales.

PLAN DE CALIDAD

-
- Revisar entregables parciales y su calidad.
 - Validar comunicación continua y seguimiento del avance.
 - Registrar incidentes o desviaciones del proveedor.
 - Checklist de Prueba de Aceptación.
 - Confirmar ejecución del plan de pruebas acordado.
 - Validar resultados de pruebas funcionales y técnicas.
 - Revisar resolución de defectos críticos antes de la aceptación.
 - Documentar aprobación formal por parte del cliente.

6.3.5 Administración de la Configuración de Software

- Checklist de Plan de SCM.
 - Verificar que se definan claramente los ítems de configuración.
 - Confirmar lineamientos de versionado para código, documentos y plantillas.
 - Validar que exista estrategia de ramas, integración y release.
 - Revisar procedimientos de control de cambios y autorizaciones.
 - Confirmar mecanismos de auditoría de configuración.
 - Validar ambiente de repositorio y accesos definidos.

6.3.6 Aseguramiento de Calidad de Software

- Checklist de Revisión del Plan de SQA.
 - Validar que el plan define roles, responsabilidades y alcance.
 - Confirmar que se incluyen los estándares, procesos y templates aplicables.
 - Revisar que estén definidas las actividades de auditoría y puntos de control.
 - Verificar que se establecen métricas de calidad y formas de medición.
 - Confirmar alineación con el Plan de Proyecto y el ERS.
 - Validar que se considera la calidad tanto de software como de hardware IoT.

6.4 Procesos, Guías y Procedimientos

{ Definir los procesos, guías y procedimientos a seguir durante el transcurso del proyecto. Estas definiciones usualmente son independientes al SQAP y se hará referencia a los documentos correspondientes dentro de esta sección. Los procesos, guías y procedimientos definidos en esta sección serán tomados por el equipo del proyecto para realizar su trabajo y SQA debe verificar que estos sean realizados como se indica en ellos. Incluir también la definición de los procesos, guías y procedimientos que el equipo de SQA debe seguir para realizar su trabajo.

Ejemplos de procesos, guías y procedimientos son:

- Guía para realizar reuniones efectivas
 - Ciclo de Vida de Desarrollo del Proyecto
 - Proceso de Ingeniería de Requerimientos
 - Proceso de Testing
 - Procedimientos para la administración de datos
 - Procedimientos para la administración de ambientes de desarrollo y testing
 - Procesos de revisión (Recorrida, Revisión Técnica, Inspección, Post-Proyecto, de Avance, etc.)
 - etc.
- }

6.4.1 Planificación de Proyectos

Este proceso define cómo se planifica un proyecto dentro del control de SQA:

- Definición del Plan de Proyecto:
Elaborar el plan siguiendo el template oficial, asegurando que se documentan alcance, objetivos, entregables, riesgos y cronograma.
- Identificación de Actividades de SQA:
Incluir todas las tareas de auditoría, revisiones y aseguramiento definidas en el Plan de SQA.
- Estructura del Cronograma:
Crear una WBS con actividades, responsables, dependencias y fechas planificadas.

PLAN DE CALIDAD

-
- Validación del Plan:
El Responsable de SQA revisa y aprueba que el plan cumpla con estándares y cubra todo lo requerido para el aseguramiento de la calidad.

6.4.2 Control y Seguimiento de Proyectos

Este proceso establece cómo se monitorea el avance del proyecto:

- Elaboración del Informe de Avance:
Registrar hitos logrados, entregables completados, problemas surgidos, riesgos y acciones correctivas.
- Reuniones de Seguimiento:
Realizar reuniones periódicas con Líder de Proyecto, equipo técnico, SCM y SQA para revisar el estado actual.
- Actualización del Cronograma:
Ajustar porcentajes de avance y reprogramar actividades cuando sea necesario.
- Control de Desviaciones:
Identificar diferencias entre avance real y planificado, documentarlas y definir acciones correctivas.

6.4.3 Administración de Requerimientos

Este proceso describe cómo se gestionan y controlan los requisitos:

- Documentación de Requisitos:
Elaborar la Especificación de Requerimientos usando el template oficial y aplicando el estándar IEEE 830.
- Revisión Formal de Requisitos:
Realizar inspecciones apoyadas en checklists para validar claridad, consistencia, completitud y verificabilidad.
- Matriz de Trazabilidad:
Generar y mantener la trazabilidad entre requisitos, diseño, casos de uso, pruebas y cambios.
- Gestión de Cambios:
Registrar, aprobar e implementar cambios de forma controlada, evaluando impacto en cronograma, arquitectura y pruebas.

6.4.4 Administración de la Subcontratación de Software

Este proceso regula el trabajo con proveedores externos:

- Estrategia de Contratación:
Definir criterios técnicos, comerciales y de calidad para seleccionar proveedores.
- Solicitud de Propuesta (RFP):
Elaborar y enviar la RFP utilizando el template oficial.
- Evaluación de Proveedores:
Aplicar encuestas, análisis comparativos y matriz de evaluación para seleccionar la opción más adecuada.
- Monitoreo del Proveedor:
Revisar avances, entregables, cumplimiento de plazos y calidad del producto subcontratado.
- Aceptación del Producto:
Ejecutar pruebas de aceptación y validar criterios antes del cierre contractual.

6.4.5 Administración de la Configuración de Software

Este proceso define cómo se controlan las versiones y configuraciones:

- Plan de SCM:
Implementar el plan utilizando el template oficial para definir repositorios, estrategias de versionado y control documental.
- Identificación de Ítems de Configuración:
Registrar código, documentos, plantillas y artefactos críticos bajo control de SCM.
- Control de Versiones:
Aplicar prácticas de branching, integración y liberación según lo definido en el Plan de SCM.
- Auditorías de Configuración:
SQA verifica que las versiones, cambios y entregables se encuentran correctamente identificados y almacenados.

6.4.6 Aseguramiento de Calidad de Software

Este proceso establece cómo se garantiza la calidad del producto final:

- Revisiones y Auditorías:
Ejecutar inspecciones de documentos, código y diseño usando los checklists correspondientes.
- Pruebas y Validaciones:
Preparar las actividades de pruebas funcionales, de integración, de humo y de regresión según los estándares del proyecto.
- Registros de SQA:
Documentar hallazgos, conformidades, desviaciones y acciones correctivas en los formatos oficiales.
- Reporte de Actividades:
Generar reportes periódicos con resultados de auditorías, métricas de calidad y recomendaciones.
- Verificación de Cumplimiento:
Asegurar que los entregables cumplan los estándares definidos y que los mecanismos de control funcionan correctamente.

6.5 Mediciones

{

Las mediciones del proyecto usualmente se definen en el Plan de Proyecto. Las mediciones de SQA pueden también ser incluidas en el Plan de Proyecto, pero deben aparecer definidas al menos en el Plan de SQA y ser referenciadas en el Plan de Proyecto o viceversa.

En esta sección defina las mediciones de calidad que serán recolectadas durante el proyecto y describa su uso.

Las mediciones del producto deben ser reportadas y utilizadas para administrar el desarrollo y entrega del producto. Ejemplos de mediciones sobre el producto son: número de defectos encontrados, cantidad de nuevas líneas de código, cantidad de líneas de código reusadas, etc.

Las mediciones sobre el proceso de desarrollo deben reportarse y utilizarse para determinar la calidad del proceso. Ejemplos de mediciones sobre el proceso son: duración prevista y real de cada etapa de desarrollo, costos estimados y reales del proyecto, y esfuerzo estimado y real de cada fase.

Las mediciones sobre el desempeño de SQA también deben ser definidas aquí y deben abarcar como mínimo a las siguientes: realización de los hitos comparados contra el SQAP, trabajo completado, esfuerzo realizado, presupuesto gastado por las actividades de SQA comparado con el SQAP, cantidad de revisiones comparadas con el SQAP, cantidad de desviaciones observadas por SQA.

}

Las mediciones referidas a la calidad del software se definen en el **Plan de Proyecto**, en la sección **Mediciones**.

Las mediciones permiten evaluar el desempeño del proyecto, la calidad del producto y la efectividad de los procesos utilizados. Para el proyecto Parkly, se utilizarán métricas cuantitativas y cualitativas que permitan dar seguimiento al avance, detectar desviaciones y asegurar la mejora continua.

Las mediciones consideradas en este Plan SQA incluyen:

- Métricas de Proceso

Estas métricas permiten evaluar la eficiencia con la que se ejecutan las actividades del proyecto:

- Cumplimiento del cronograma:
Porcentaje de actividades completadas según lo planificado.
- Índice de retrabajo:
Proporción de actividades que requieren correcciones o rehacerse completamente.
- Cumplimiento de auditorías SQA:
Número de actividades de aseguramiento realizadas respecto a las planificadas.
- Número de desviaciones detectadas en auditorías:
Permite medir la conformidad con los procesos.

- Métricas de Producto

Estas métricas evalúan la calidad del software y su grado de cumplimiento con los requisitos:

- Densidad de defectos:
Número de defectos encontrados por módulo, sprint o fase.
- Cobertura de requisitos:
Porcentaje de requisitos funcionales y no funcionales implementados y verificados.
- Índice de estabilidad de requerimientos:
Cantidad de cambios en los requisitos a lo largo del proyecto.

- Métricas de Calidad del Código

- Complejidad ciclomática

PLAN DE CALIDAD

-
- Permite evaluar la mantenibilidad del software
 - Porcentaje de código revisado
Evalúa el grado de inspección aplicada antes de la integración.
 - Porcentaje de defectos encontrados en revisión vs. pruebas
Mide la efectividad de las revisiones tempranas.
 - Métricas de Pruebas
 - Cobertura de pruebas:
Porcentaje de casos de prueba ejecutados respecto a los planificados.
 - Tasa de fallas:
Número de fallos detectados durante las pruebas por unidad de tiempo
 - Defectos abiertos vs. cerrados:
Permite evaluar el avance en la resolución de problemas.
 - Tiempo promedio de resolución de defectos:
Mide la eficiencia del equipo en solucionar incidencias.
 - Métricas de Satisfacción del Usuario / Cliente
 - Encuestas de satisfacción:
Evaluación de la percepción del cliente respecto a estabilidad, usabilidad y desempeño.
 - Índice de aceptación del producto:
Porcentaje de entregables aprobados sin necesidad de correcciones mayores.
 - Reporte y Seguimiento de Métricas
 - Todas las métricas se recopilarán en los Reportes de Actividades SQA.
 - Los resultados serán analizados en reuniones de seguimiento.
 - Cualquier desviación significativa deberá generar acciones correctivas.
 - Las métricas se actualizarán de manera periódica según lo establecido en el Plan de Proyecto.

7 TAREAS DE SQA

Se describen las tareas que SQA debe realizar para armar su propia infraestructura y las tareas de control de los productos y procesos críticos respecto a la calidad del producto.

7.1 Tareas de Infraestructura

Esta sección detalla las tareas iniciales que el equipo de SQA deberá llevar a cabo para establecer la infraestructura operativa necesaria para ejecutar el Plan de Calidad para el sistema Parkly.

1. Obtención y capacitación del personal SQA

Definir los perfiles requeridos para el equipo de SQA y ejecutar el proceso de integración. Dada la naturaleza híbrida (software/hardware) de Parkly, se requiere experiencia especializada.

Procedimiento.

- Definición del Perfil. Elaborar un perfil que incluya competencias en pruebas de backend (API REST) y conocimientos fundamentales en diagnóstico de fallos (troubleshooting) de sistemas de conectividad IoT.
- Capacitación Específica. Proporcionar entrenamiento en la utilización de la plataforma de recolección de datos del gateway IoT y en la interpretación de los protocolos de comunicación utilizados por los sensores de ocupación.

2. Recopilación y estandarización de artefactos SQA

Generar, formalizar y difundir el conjunto de documentación (estándares, templates, procedimientos) que regirá las actividades de calidad.

Procedimiento.

- Desarrollo de Templates. Crear plantillas obligatorias para el registro de casos de prueba, planes de prueba y reportes de defectos. Estos deben incluir campos específicos para la trazabilidad de los componentes de hardware.
- Definición de Estándares. Adoptar estándares de la industria y documentar los procedimientos internos para las pruebas de regresión y pruebas de humo previas a cada release.

3. Definición e Implementación de Métricas de SQA

Seleccionar e implementar los indicadores clave de rendimiento para la medición objetiva del producto y la eficiencia del proceso QA.

Procedimiento.

- Métricas del Producto. Automatizar la recolección de densidad de defectos, así como las métricas críticas del sistema en el entorno operativo como la latencia de actualización de datos.
- Métricas del Proceso. Establecer métricas de eficiencia interna como el tiempo promedio de ciclo de defectos.

4. Configuración e instalación de Herramientas de Soporte

Seleccionar, adquirir y configurar el conjunto de herramientas de software que soportarán la gestión y la automatización de las pruebas.

Procedimiento.

- Gestión QA. Instalar y configurar una herramienta centralizada para la gestión de requisitos, casos de prueba y defectos.
- Automatización. Establecer el entorno de ejecución de scripts para la automatización de pruebas funcionales y de API.
- Pruebas de Carga. Instalar herramientas de rendimiento para simular la concurrencia de usuarios y el volumen de tráfico de datos del entorno de despliegue real.

5. Integración del SQA con el Ciclo de Desarrollo

Integrar las actividades de SQA dentro del ciclo de Integración Continua y Despliegue Continuo.

Procedimiento:

- Definición de Quality Gates: Establecer los criterios de entrada y salida de cada fase.
- Concientización del Equipo. Realizar sesiones informativas para capacitar al equipo de desarrollo y gestión sobre el rol de SQA, el uso adecuado de los

PLAN DE CALIDAD

templates de defectos y los procedimientos de escalación.

Las tareas a realizar por SQA para desarrollar su propia infraestructura dentro del proyecto se definen en el **Plan de Proyecto** y en el cronograma asociado correspondiente.

7.2 Productos bajo Control de Calidad

En esta sección se indican, para aquellos productos del proyecto considerados críticos con respecto a la calidad, los estándares, guías, procedimientos y templates que deben seguirse en su construcción y cómo será su verificación, haciendo referencia al proceso de revisión a seguir y checklists utilizados

PLAN DE CALIDAD

Documento	Activos de Proceso a utilizar	Proceso a seguir – Participantes	Rol SQA
Planificación de Proyectos			
Plan de Proyecto	Template de Plan de Proyecto	Revisión – Líderes, Gerente Ejecutivo, Auspiciante, Cliente, Resp. SQA	Verificar la inclusión de todas las actividades y recursos de SQA, y la trazabilidad con los hitos del proyecto.
Cronograma	Template de Cronogramas	Revisión – Líderes, Gerente Ejecutivo, Auspiciante, Cliente, Resp. SQA	Validar que se ha asignado tiempo adecuado para las tareas de QA antes del despliegue.
Conteo de PF	Template de conteo de PF	Revisión – Líderes, Analistas Funcionales	
Control y Seguimiento de Proyectos			
Informe de Avance de Proyectos.	Template de Informe de Avance de Proyectos.	Reunión de Avance de Proyecto – Líder, Equipo Técnico, SQA, SCM.	
Administración de Requerimientos			
Especificación de Requerimientos de Software.	Template de Especificación de Requerimientos de Software. Estandar IEEE 830.1998	Revisión – Analistas Funcionales, Miembros de Calidad, Miembros del Equipo Técnico (Diseño),	Asegurar la verificabilidad y consistencia de los requisitos funcionales y no funcionales de Parkly.
Especificación de Arquitectura	Template de Arquitectura de Software.	Revisión - Ingeniero de IoT y despliegue de Harware, Miembros de Calidad	Supervisar la revisión conjunta entre el arquitecto y el desarrollador de código. Verificar el diseño de la capa de comunicación y el procesamiento de la data cruda del sensor.
Código Fuente	Estándar de Codificación interno de Parkly.	Revisión - Arquitecto de Software/Desarrollad	Verificar la cobertura de pruebas unitarias y de integración de

PLAN DE CALIDAD

		or, Miembros de calidad	los módulos críticos.
Administración de la Subcontratación de Software			
Plan de Adquisición.	Template de Plan de Adquisición.	Revisión – Gerente Ejecutivo, Líderes, Auspiciante, Cliente, Resp. SQA, Resp. SCM	
Solicitud de Propuesta.	Template de Solicitud de Propuesta.	Revisión – Gerente Ejecutivo, Líderes, Auspiciante, Cliente, Resp. SQA, Asesor Legal, Resp. SCM	
Encuesta a Cliente del Proveedor.	Template de Encuesta a Cliente del Proveedor.	Revisión – Líder, Analistas Funcionales	
Encuesta a Proveedor Candidato.	Template de Encuesta a Proveedor Candidato.	Revisión – Líder, Analistas Funcionales	
Evaluación de productos enlatados.	Template de Evaluación de productos enlatados.	Revisión – Líder, Analistas Funcionales	
Matriz de Evaluación.	Template de Matriz de Evaluación.	Revisión – Líder, Analistas Funcionales	
Evaluación Proveedor – Producto.	Template de Informe de Evaluación Proveedor – Producto.	Revisión – Líder, Analistas Funcionales	
Contrato.	Modelo de Contrato.	Revisión – Gerente Ejecutivo, Líderes, Auspiciante, Cliente, Resp. SQA, Resp. SCM, Asesor Legal	

PLAN DE CALIDAD

Administración de la Configuración de Software			
Plan de SCM.	Template de Plan de SCM.	Revisión – Líder, Resp. SQA, Resp. SCM, Analistas, Testing	Asegurar que el plan de SCM define el control de versiones para los activos críticos (código y templates de pruebas). Testing debe validar la estrategia de control de versiones del software.
Aseguramiento de Calidad de Software			
Checklists	Template de Checklist	Revisión – Miembros SQA, Resp. SQA, Expertos en la materia	Crear y validar los checklists específicos para la prueba de humo (sanity check) de la App móvil y la verificación inicial del sensor (IoT).
Reporte de Actividades de SQA	Template de Reporte de Actividades de SQA.	Revisión – Miembros SQA, Resp. SQA	
Plan de SQA	Template de Plan de SQA.	Revisión - Miembros SQA, Resp. SQA, Líder, Gerente Ejecutivo	Documentar y obtener la aprobación formal de todos los stakeholders sobre el proceso de calidad.

7.3 Procesos bajo Control de Calidad

En esta sección se indican, para aquellos procesos del proyecto considerados críticos con respecto a la calidad, la definición de cada uno de ellos y la forma en que tales procesos serán controlados por el equipo de SQA.

Proceso	Activos de Proceso a utilizar	Método de comprobación y Tarea específica de SQA	Rol SQA Asignado
Planificación de Proyectos	Proceso de Planificación de Proyectos	Participación en el Proceso – Resp. SQA. Tarea: Verificar que el Plan de Proyecto contemple la	Gerente de Proyecto o Analista de SQA

PLAN DE CALIDAD

		asignación de recursos y tiempo adecuados para todas las pruebas (integración, rendimiento, IoT).	
Control y Seguimiento de Proyectos	Proceso de Control y Seguimiento de Proyectos	Auditoría, Reportes de Administración de Proyecto – Resp. SQA. Tarea: Auditar que las reuniones de avance de proyecto se lleven a cabo según el cronograma y que los informes de avance reflejen el estado real de los defectos críticos y el cumplimiento de los objetivos planteados.	Gerente de Proyecto
Administración de Requerimientos	Proceso de Administración de Requerimientos	Auditoría – Resp. SQA. Tarea: Auditar el proceso de cambio de requisitos para asegurar que toda modificación a la SRS sea formal y que los casos de prueba se actualicen en consecuencia.	Analista de SQA
Administración de la Subcontratación de Software	Proceso de Administración de la Subcontratación de Software	Auditoría – Resp. SQA. Tarea: Auditar el proceso de selección y evaluación de proveedores (ej. plataforma de cloud) para garantizar el cumplimiento de los criterios técnicos de calidad.	Ingeniero de IoT y despliegue de Hardware
Administración de la Configuración de Software	Proceso de Administración de la Configuración de Software	Auditoría, Reportes de SCM – Resp. SQA. Tarea: Auditar que se sigan los procedimientos de check-in/check-out de código y que la versión de software liberada a pruebas y producción corresponda a la versión aprobada.	Arquitecto de Software
Aseguramiento de Calidad de Software	Proceso de Aseguramiento de Calidad de Software	Auditoría Externa, Reportes de SQA –	Gerente de Proyecto

PLAN DE CALIDAD

		Equipo SQA Externo al proyecto, Gerente Ejecutivo. Tarea: Revisar la conformidad con el Plan de SQA y reportar las no conformidades del proceso a la gerencia ejecutiva.	
Testing (Pruebas)	Proceso de Ejecución de Pruebas de Software	Muestreo de la Ejecución de Casos de Prueba. Tarea: Seleccionar aleatoriamente un subconjunto de casos de prueba críticos y auditar su ejecución por parte del equipo de Testing para validar la documentación de resultados.	Analista de SQA

7.4 Cronograma de Actividades de SQA

Actividad	02/02/26 - 06/02/26	16/02/26 - 20/02/26	02/03/26 - 06/03/26	09/03/26 - 13/03/26	23/03/26 - 27/03/26	06/04/26 - 10/04/26	13/04/26 - 17/04/26	20/04/26 - 24/04/26	27/04/26 - 01/05/26	11/05/26 - 15/05/26	18/05/26 - 22/05/26
Definición e implementación de Métricas de Calidad											
Infraestructura de SQA Completada (Setuo de herramientas y templates listos a utilizar)											
Revisión y Aprobación de la Especificación de Requerimientos											
Auditoría del Proceso de Administración de Requerimientos (verificación del control de cambios)											
Revisión de la Especificación de Arquitectura											
Auditoría del Plan SCM											
Creación y Aprobación de Casos de Prueba Críticos											
Auditoría continua del proceso de Testing (muestreo de ejecución de pruebas)											
Reporte de calidad de Producto											
Auditoría final del Plan del Proyecto											
Cierre Formal de Calidad											

Las tareas a realizar por SQA dentro del proyecto se definen en el **Plan de Proyecto** y en el cronograma asociado correspondiente.

8 SOPORTE DE SQA A CLIENTES, PROVEEDORES Y SUBCONTRATISTAS

Esta sección define las previsiones y mecanismos que el equipo de Aseguramiento de la Calidad de Software (SQA) implementará para garantizar que los productos y servicios proporcionados por entidades exteriores (proveedores, subcontratistas y clientes) cumplan con los requisitos de calidad del proyecto ParkIt, según lo establecido en la Especificación de Requerimientos (SRS).

Control de Calidad en Proveedores y Subcontratistas

1. Proveedores de Hardware/Tecnología
 - a. Objetivo. Asegurar la funcionalidad y confiabilidad del hardware que alimenta la plataforma.
 - b. Mecanismos de control SQA.
 - i. Revisión de Especificaciones. El ingeniero de IoT, con apoyo del SQA, revisará las especificaciones técnicas del proveedor para asegurar que se alineen con los requisitos del sistema Parkly.
 - ii. Auditoría de Muestreo de Productos Entregados. Se realizará una auditoría de calidad por muestreo a la recepción de lotes críticos de sensores (tasa de error y latencia de comunicación de sensores).
 - iii. Pruebas de Integración y Rendimiento. SQA someterá el hardware a pruebas de estrés en un entorno de laboratorio para verificar su rendimiento bajo condiciones operativas extremas antes de su aceptación formal.
2. Subcontratistas de Desarrollo de Software
 - a. Objetivo. Garantizar que el código y los módulos desarrollados externamente cumplen con los estándares de codificación y las especificaciones de arquitectura internas.
 - b. Mecanismos de Control SQA.
 - i. Revisión Formal de Productos Entregados. SQA exigirá al subcontratista la entrega de los artefactos críticos (código fuente, plan de testing) para una revisión conjunta con el Arquitecto de Software.
 - ii. Auditoría del Proceso SCM del subcontratista. El responsable SQA auditará el proceso de control de configuraciones (SCM) del subcontratista para asegurar la trazabilidad del código y el uso de un repositorio de versiones.
 - iii. Verificación de cobertura de pruebas. SQA requerirá y verificará la cobertura de pruebas unitarias (mínimo 80%) de todo el código entregado por el subcontratista antes de integrar el módulo al sistema principal de Parkly.

Soporte de SQA al Cliente. El cliente principal de Parkly es el administrador del estacionamiento. El SQA jugará un rol clave en la última fase de validación de calidad.

- Mecanismo. El equipo SQA organiza y supervisa las pruebas de aceptación del Usuario (UAT).
- Tareas del SQA.
 - Guía y coordinación. El SQA ayuda al administrador a ejecutar una serie de escenarios de negocios reales.
 - Registro de Fallos. SQA es el encargado de registrar cualquier problema o fallo que el administrador encuentre durante estas pruebas y de asegurarse de que sean corregidos antes de la entrega final.

Las previsiones de SQA para asegurar que los productos de clientes, proveedores y subcontratistas logren los requisitos de calidad establecidos, se definen en el documento **Plan de Contratación** del proyecto en la sección **Garantía de calidad del software SQA**

9 REGISTROS Y REPORTES DE SQA

{

Describa la documentación que se requiere que el grupo de SQA produzca. Identifique el método y frecuencia con que se reportará al grupo de desarrollo de software y a otros grupos relacionados de software acerca de las actividades de SQA.

Describa una lista de registros de calidad que serán mantenidos durante el proyecto.

Describa la identificación, recolección, clasificación, backup, almacenamiento, mantenimiento y depuración de los registros de calidad.

}

Se describen los registros e informes de SQA a utilizar en el proyecto.

9.1 Registro de SQA

Propósito Registrar formalmente cualquier evento evaluado por SQA, ya sea:

- Una actividad completada exitosamente,
- Una desviación respecto al plan,
- Un hallazgo en auditorías, revisiones o pruebas.

Roles Gerencia Superior: destinatario
 Grupo de QA: originador del informe
 Copias a: Líderes del proyecto

Contenidos Cada registro deberá incluir:

- Fecha y hora del evento.
- Tipo de actividad (revisión, auditoría, prueba, entrega).
- Lista de participantes.
- Descripción clara del evento.
- Referencia a checklist o estándares aplicados.
- Resultado (conformidad / desviación).
- Recomendaciones o acciones sugeridas.

Seguimiento En caso de desviación:

- Se definen e implementan acciones correctivas.
- Se archivan los informes de problemas.
- SQA valida posteriormente la corrección.

Registro de SQA	
Fecha	15/01/2026
Para	Hernández Perrusquia Jhoel (Gerente de Proyecto)
Copia	Fernandez Romero Laura Lucia (Asesor Docente)
De	Díaz García Yaneli Guadalupe (Ejemplo de miembro de calidad)
Descripción	Se realizó la validación de detección de 5 sensores infrarrojos instalados en el estacionamiento de prueba al mediodía.
Conformidad	NO CONFORME (Hubo un fallo)
Observaciones	2 de los 5 sensores arrojaron "Ocupado" estando vacíos debido a la interferencia de la luz solar directa en el receptor infrarrojo.
Seguimiento	Se requiere instalar una cubierta física en los sensores o recalibrar el umbral de luz en el código del microcontrolador.
Necesario	Si
Esfuerzo Insumido	2 horas
Tiempo Insumido	13:00 - 15:00 horas

9.2 Reporte de SQA

Propósito Proporcionar un resumen mensual del trabajo de SQA, que incluya:

- Actividades realizadas,
- Procesos revisados,
- Auditorías ejecutadas,
- Métricas del proyecto,
- No conformidades abiertas.

Roles Gerente de TI: destinatario
Grupo de QA: originador del informe
Copias a: Jefes de Área adecuados, Gerencia Superior, Jefe de QA.

Reporte de Actividades de SQA	
Fecha	31/01/2026
Área	Desarrollo Backend
Actividades de QA del área durante este mes	<ol style="list-style-type: none"> 1. Revisión del documento ERS-830. 2. Pruebas unitarias del módulo de "Autenticación de Usuarios". 3. Inspección del diagrama de base de datos.
Estado de los procesos del área	Cumple parcial
Observaciones sobre el área	"Falta documentación del módulo de pagos."
Mediciones de QA	24 horas
Tiempo Insumido	12 horas
Esfuerzo Insumido	16 horas
Hitos logrados	"Diseño de arquitectura aprobado pruebas iniciales del sensor IR completadas."
Revisiones y Auditorías realizadas	"Auditoría ERS; Revisión API Parking; Auditoría de SCM."
Pruebas de Aceptación realizadas	"Prueba de flujo de login aceptada"
Número de Informes de no-conformidad por mes	3
Firma	Diaz Garcia Yaneli Guadalupe
Aclaración	Responsable de SQA

10 REPORTES DE PROBLEMAS Y ACCIONES CORRECTIVAS

{

Describa los procedimientos (o haga referencia a ellos) a ser utilizados para reportar, monitorear y resolver problemas identificados en los productos de software.

Igualmente describa los procedimientos (o haga referencia a ellos) a ser utilizados para reportar, monitorear y resolver problemas identificados en el proceso de desarrollo de software.

SQA debe documentar, o hacer referencia a, un procedimiento para manejar desviaciones de lo real respecto a lo esperado, que típicamente indica que:

- Las desviaciones relacionadas con Plan de Desarrollo de Software, y con los estándares y procedimientos designados para el proyecto, son documentados y resueltos con los líderes involucrados o el gerente del proyecto, cuando sea posible.
- Las desviaciones relacionadas con Plan de Desarrollo de Software, y con los estándares y procedimientos designados para el proyecto, y que no se resuelvan con los líderes o el gerente del proyecto, son documentados y presentados al gerente superior designado para recibir problemas de no cumplimiento.
- Los puntos de no cumplimiento presentados al gerente superior son revisados periódicamente hasta que sean resueltos.
- La documentación de puntos de no cumplimiento es administrada y controlada.

Describa también el proceso a ser usado para detectar y eliminar causas potenciales de problemas o defectos.

}

Ver arriba, en la sección **Registro de SQA**, cómo se tratan los problemas y no conformidades dentro del proyecto. Una descripción más detallada de los procedimientos para reportar, monitorear y resolver problemas identificados en los productos y procesos de software se encuentra en el documento **Proceso de SQA**.

10.1 Procedimiento para reportar problemas

Cuando se detecta un problema en un producto o proceso, se debe:

1. Registrarlo formalmente en el **Registro de SQA**.
2. Clasificarlo como desviación, defecto o no conformidad.
3. Notificarlo a los líderes correspondientes.
4. Archivar evidencia (capturas, logs, checklist utilizado).

(PlanSQA Parkly.docx)

10.2 Procedimiento para monitorear problemas

SQA debe dar seguimiento regular hasta su resolución:

- Validación de que se definieron acciones correctivas.
- Confirmación de que fueron implementadas.

PLAN DE CALIDAD

-
- Cierre formal del problema.
 - Elaboración de registro actualizado.
(PlanSQA Parkly.docx)

10.3 Tratamiento de desviaciones

El procedimiento oficial indica:

- Las desviaciones **se discuten y resuelven primero** con los líderes del proyecto o el gerente de proyecto.
- Si no se resuelven, se escalan a Gerencia Superior.
- La gerencia revisa periódicamente todos los puntos abiertos.
- Los documentos y evidencias de no conformidad quedan bajo control documental.
(PlanSQA Parkly.docx)

10.4 Acciones Correctivas

Las acciones correctivas buscan eliminar la causa raíz del problema.

Incluyendo:

- Análisis causa–efecto (diagrama Ishikawa o 5 porqués).
- Establecer responsable y fecha de corrección.
- Verificación por parte de SQA.
- Actualización de estándares o procesos si es necesario.
(PlanSQA Parkly.docx)

10.5 Referencias adicionales

Para detalle completo, el documento hace referencia al **Proceso de SQA**, encargado de guiar formalmente el tratamiento de no conformidades.

11 APÉNDICES

{

Incluya bajo esta sección aquella información adicional que considere relevante para este plan y que haya decidido mantenerla fuera del cuerpo principal del documento. Usualmente habrá al menos un glosario y la historia de cambios del documento.

}

Esta sección incluye información complementaria que facilita la comprensión del Plan de SQA, así como el seguimiento histórico de los cambios realizados al documento. Los apéndices sirven como referencia para términos clave utilizados en el proyecto y para garantizar el control de versiones del presente plan.

11.1 Glosario

SQA: Software Quality Assurance.

SCM: Software Configuration Management.

ERS: Especificación de Requerimientos de Software.

BPMN: Business Process Model and Notation.

Stakeholder: Parte interesada en el proyecto.

Defecto: Cualquier desviación respecto al comportamiento esperado del sistema.

Archivos relevantes (Modelo de negocio, Requisitos, Proyect Charter) :

https://drive.google.com/drive/folders/13Vsqjgn40hewBLPWDj4_1M8NW6-XmFEa?usp=sharing

11.2 Historia de Cambios

Revisión	Autor	Fecha	Descripción
1.0	Equipo SQA	2025-01-15	Versión inicial del documento
1.1	Equipo SQA	2025-01-22	Ajustes en sección 6 y definición de métricas