

**INFORME N.º R.J.P.G-AUD-IT-0001-2020**  
21 de agosto, 2020

**INFORME DE AUDITORIA SISTEMA TAILS & PAWS**

**2020**

## Contenido

<b>Resumen Ejecutivo.....</b>	<b>3</b>
<b>1.Introducción.....</b>	<b>7</b>
OBJETIVOS .....	7
ALCANCE.....	8
ASPECTOS POSITIVOS QUE FAVORECIERON LA EJECUCIÓN DE LA AUDITORÍA .....	8
LIMITACIONES QUE AFECTARON LA EJECUCIÓN DE LA AUDITORÍA .....	8
GENERALIDADES ACERCA DEL OBJETO AUDITADO .....	8
<b>2. Resultados.....</b>	<b>9</b>
ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES Y EXCEPCIONES.....	9
DEFINICION Y CRITERIOS DEL ALCANCE DEL SISTEMA.....	10
PLANIFICACIÓN DE LA ARQUITECTURA DONDE EL SISTEMA SERÁ ALOJADO .....	10
PLAN DE PRUEBAS.....	12
MANEJO DE COMUNICACIONES Y CONTROL DE VERSIONES.....	12
<b>3. Conclusiones.....</b>	<b>14</b>
ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES Y EXCEPCIONES.....	14
DEFINICION Y CRITERIOS DEL ALCANCE DEL SISTEMA.....	14
PLANIFICACIÓN DE LA ARQUITECTURA DONDE EL SISTEMA SERÁ ALOJADO .....	15
PLAN DE PRUEBAS.....	15
MANEJO DE COMUNICACIONES Y CONTROL DE VERSIONES.....	15
<b>4. Recomendaciones.....</b>	<b>17</b>

# Resumen Ejecutivo

## ¿Qué se examinó?

La Auditoria se enfocó en determinar si la documentación del sistema informático Tails & Paws fue desarrollada de conformidad con las mejores prácticas y procedimientos apropiados para garantizar el buen desempeño de los sistemas de información debido a que una correcta implementación proporciona los controles necesarios para que los sistemas sean confiables y con un buen nivel de seguridad.

La Auditoría realizada al sistema informático Tails & Paws durante el período comprendido entre el 19 de junio al 21 de agosto valoro aspectos como la definición de requerimientos, la gestión del alcance del proyecto.

Uno de los puntos que se audito sobre la importancia del documento DN01 Documento de Arquitectura de Software 2020 del sistema fue la evaluación del plan de pruebas si el mismo cumplía con los requerimientos necesarios para la implementación del producto desarrollado, dentro de los requerimientos necesarios para cumplir las pruebas y documentarlos se examinó que realmente se indicara los alcances de las pruebas, elementos de pruebas funcionales, enfoque de las pruebas, criterios de aceptación o rechazo, planificación y organización de las pruebas, dependencias y riesgos de las pruebas funcionales, mitigación de riesgos en pruebas funcionales e informe de pruebas todos puntos fueron evaluados y examinados con el fin de poder validar si realmente el sistema están listo para la implementación. Dentro de este mismo documento se realizó la evaluación sobre la planificación de la arquitectura que estará siendo utilizada una vez el sistema se encuentre en producción y todo lo que en ello conlleva, seguridad, actualizaciones, donde se encontrará de manera física o que proveedor va a tener este sistema.

Finalmente, se evaluó la gestión de las comunicaciones tanto internas como con interesados fuera del equipo de desarrollo, así como registros de control de cambios, con el fin de conocer el grado de disponibilidad de la información en caso de necesitar compartirse o realizar consultas de datos históricos.

### **¿Por qué es importante?**

Los sistemas informáticos manejan datos de diversa índole entre ellos información de clientes, proveedores, organizaciones que por su carácter privado requieren ser desarrollados siguiendo estándares que facultan al equipo de desarrollo a seguir una serie de lineamientos que aseguren un manejo óptimo, seguro y confiable de esta información de tal manera que se mitiguen impactos, se logren los objetivos y se dé un aseguramiento de la calidad de la información. La importancia de las pruebas en desarrollo de software en un proyecto de desarrollo de software puede aparecer errores en cualquiera de las etapas del ciclo de vida, algunos de ellos incluso permanecen sin ser descubiertos, de ahí la importancia de las pruebas en desarrollo de software.

Hay una gran probabilidad de que el código final tenga errores tanto de requerimientos, como de diseño o de funcionalidad. Para identificar estos problemas antes de que ocurran en un entorno crítico, es necesario realizar pruebas de software, una parte muy importante del proceso, pero también muy costosa; sin embargo, debemos tener en cuenta que el coste debido a un fallo mientras está el software en funcionamiento puede llegar a ser mucho mayor.

Por último, el manejo de registros históricos tanto de versiones de los documentos y del sistema como tal, así como de las comunicaciones, tiene una alta importancia en caso de que el equipo de desarrollo necesite respaldarse respecto a algún evento que suceda y donde se deba registrar algún cambio que se haya llevado a cabo y la razón por la cual se ejecutó.

### **¿Qué encontramos?**

Una vez se finalizó la presente auditoría, la empresa de consultoría R.J.P.G Consulting ha determinado que existen algunas debilidades en el análisis de requerimientos como por ejemplo la definición de algunas excepciones de los requerimientos funcionales, la extensión del alcance del sistema presenta algunas áreas grises donde no se pueden medir adecuadamente los impactos ante el no cumplimiento de los objetivos planteados, lo cual pone de manifiesto un riesgo desde el punto de vista de una adecuada operación del sistema, alcance de las metas propuestas y la estrategia ante cualquier desviación durante el proceso de desarrollo. Cuando se finalizó la audición del plan de pruebas se determinó que al no existir un plan de mitigación de pruebas, el programa corre un alto riesgo ante un eventual problema de funcionamiento los Ingenieros del Sistema Tails & Paws deben tener la capacidad de conocer y aplicar las diferentes normas,

procesos y procedimientos para garantizar la calidad de los productos software, aplicando las pruebas de calidad de software necesarias para que con ellas se pueda ayudar a reducir los riesgos en las aplicaciones, logrando que se identifiquen los defectos antes de que se ejecuten, así de forma proactiva tomar decisiones que permitan hacer las actividades necesarias para mejorar las condiciones del software y ofertar un producto que satisfaga las necesidades del cliente. En cuanto al registro de las comunicaciones se puede determinar una estructura y un orden correcto para las comunicaciones internas, pero no tan explícito y más bien muy limitado para las comunicaciones con interesados externos. Respecto al control de versiones, se puede determinar que los documentos disponibles contienen historiales de versiones, incluyendo fechas y detalles, sin embargo, el acceso a las diferentes versiones no es fácil y puede limitar la capacidad de volver a alguna versión anterior en caso de ser necesario.

### ¿Qué sigue?

Con base en los hallazgos y observaciones que se encontraron en la presente auditoría se formulan una serie de recomendaciones al equipo de desarrollo del sistema Tails & Paws dentro de los que se resaltan: la actualización de la documentación relativa al documento de requerimientos de manera que permita realizar una revisión de las excepciones, casos de uso, dependencias. A su vez, revisar los impactos planteados en el alcance del proyecto e implementar una matriz FODA para determinar los elementos que pueden representar amenazas, fortalezas y oportunidades de mejora. Continuando sobre las recomendaciones para un plan de pruebas es necesario definir los objetivos y atributos de las pruebas de este modo, los objetivos principales al realizar pruebas serán los siguientes: detectar un error específico, descubrir errores no descubiertos antes y tener un buen caso de prueba. Además, los atributos que debería tener una buena prueba son: Intentar obtener la más alta probabilidad de encontrar un error, no debe ser redundante y no debe ser ni demasiado sencilla ni demasiado compleja dentro de las mismas es importante clasificar los tipos de pruebas para el sistema ya sean pruebas manuales: las que realiza el usuario paso a paso y pruebas automáticas: consiste en el uso de otro software y la comparación de los resultados obtenidos y los esperados. Permite adicionar pruebas cuya ejecución manual sería tediosa y/o difícil. También podemos probar módulos individuales hasta pruebas de todo el sistema en su conjunto y elegir cual es el más adecuado de implementar. Relacionado a las comunicaciones, se recomienda mantener registros separados para comunicaciones internas y con interesados externos para facilitar su acceso y consulta.

Finalmente, para el control de versiones se recomienda la utilización de un repositorio o recurso compartido para el almacenamiento tanto de documentos como de código de desarrollo para consulta posterior.

## **INFORME N. ° R.J.P.G-AUD-IT-0001-2020**

### **ÁREA DE AUDITORÍA DE SISTEMAS R.J.P.G CONSULTING**

### **INFORME DE AUDITORÍA DE SISTEMA INFORMÁTICO TAILS & PAWS**

## **1.Introducción**

#### **ORIGEN DE LA AUDITORIA**

En la actualidad, en el panorama de las Tecnologías de la Información y la Comunicación - TIC en especial el desarrollo de software, surgen dos grandes interrogantes relacionados con ¿Cómo desarrollar software? y ¿Cómo medir la calidad del software desarrollado?, de tal forma que cumpla con las características de eficiencia, flexibilidad, corrección, confiabilidad, mantenibilidad, portabilidad, usabilidad, seguridad e integridad; que determinaran su usabilidad y existencia en el tiempo.

Esto implica la utilización de metodologías o procedimientos estándares para desarrollar cada una de las etapas o fases del ciclo de vida del software, sin dejar de lado cada una de las personas que intervienen dentro del proceso y que juegan un papel trascendental en toda esta situación.

#### **OBJETIVOS**

La Auditoria se enfocará en determinar los controles y procedimientos utilizados en el sistema Tails & Paws dada su importancia para el buen desempeño de los sistemas de información debido a que proporciona los controles necesarios para que los sistemas sean confiables y con un buen nivel de seguridad.

Así mismo, evaluar el cumplimiento de la metodología aplicable a la documentación del proyecto Tails & Paws, de forma tal que se pueda establecer con claridad el logro de los objetivos del curso y la observación de las sanas prácticas en esa materia. La evaluación que se va a realizar tiene como objetivo fundamental detectar todas las falencias que se puedan encontrar a través de una auditoría con el fin de mejorar la calidad con la que se ejecutan los proceso Diseño y ejecución de testing de la plataforma tecnológica que tiene el sistema.

### **ALCANCE**

La auditoría interna se desarrollará en el tiempo estimado de tres meses iniciando 19 de junio y finalizando el 21 de agosto del 2020, considerando los detalles de los entregables que fueron establecidos en el sistema de Tails & Paws los documentos de especificación de requerimientos a continuación:

- IN01 Carta de aceptación del proyecto
- IN02 Estudio de factibilidad técnica tecnológica, económica y operacional
- IN03 Documento de alcance del proyecto (Project Charter)
- AN01 Documento de especificación de requerimientos
- DN01 Documento de Arquitectura de Software
- DN03 Prototipo del Proyecto, en “Pencil”, Minutas y Cronograma del Proyecto, Documentación de Pruebas y Manual de usuarios del sistema.

### **ASPECTOS POSITIVOS QUE FAVORECIERON LA EJECUCIÓN DE LA AUDITORÍA**

Para la realización de este estudio de auditoria se contó con la colaboración brindada por los desarrolladores del sistema Tails & Paws y así también con la ayuda de la profesora de curso de Auditoria de Sistemas (SC-704), el apoyo logístico ofrecido y el suministro oportuno de la información requerida.

### **LIMITACIONES QUE AFECTARON LA EJECUCIÓN DE LA AUDITORÍA**

Durante en el desarrollo de la auditoría, se encontró falta de oportunidad en el suministro de alguna información requerida, además de información incompletas y documentos antiguos donde no se había actualizado en su momento indicado. Otra limitante importante para la auditoría es el estado actual del proyecto, donde solo se dispone de acceso a la documentación, pero al no haber sido desarrollado aún, no se puede auditar sobre el sistema de manera general.

### **GENERALIDADES ACERCA DEL OBJETO AUDITADO**

Se observa que se ha realizado un esfuerzo importante en la revisión y actualización de los documentos del proceso, lo que garantiza que las actividades ejecutadas retroalimentan el sistema de plan de pruebas, es de resaltar la nueva estructura y métodos definidos para el proceso de pruebas donde se debe implementar donde se puede determinar la calidad del sistema.



## 2. Resultados

### ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES Y EXCEPCIONES

En cuanto a la especificación de requerimientos se detalla que la organización de los requerimientos funcionales en su amplia mayoría cumple con los criterios de claridad y completitud, sin embargo, algunos de ellos necesitan un mayor nivel de detalle, como en el requerimiento funcional **RF-07 Caso de Uso de Inicio de Sesión**, en este apartado la definición de excepciones del requerimiento no especifica el comportamiento del sistema si un usuario olvida la contraseña y que acción toma tanto el cómo el sistema.

Con respecto a los Requerimientos No Funcionales no se detalla el método de respaldo periódico de la información como parte del aseguramiento de la Continuidad del Negocio en el requerimiento **RNFF-03 Respaldos de información**, lo cual pone de manifiesto una debilidad respecto a los planes de contingencia que podría tener el sistema

Por último, presencia de ambigüedad en la redacción del requerimiento no funcional **RNFS-07 Certificado TLS** ya que se mencionan medidas de seguridad, protección, pero no se especifican cuáles lo que no permite un adecuado entendimiento de las soluciones a implementar para este fin.

Ante lo anterior, el equipo de desarrollo del sistema fue consultado y se manifestó que el documento de requerimientos tenía pendientes algunas actualizaciones las cuales pretendían cubrir dichos puntos.

## **DEFINICION Y CRITERIOS DEL ALCANCE DEL SISTEMA**

El análisis de esta documentación permitió conocer los siguientes aspectos e hitos respecto a elementos de riesgo y controles:

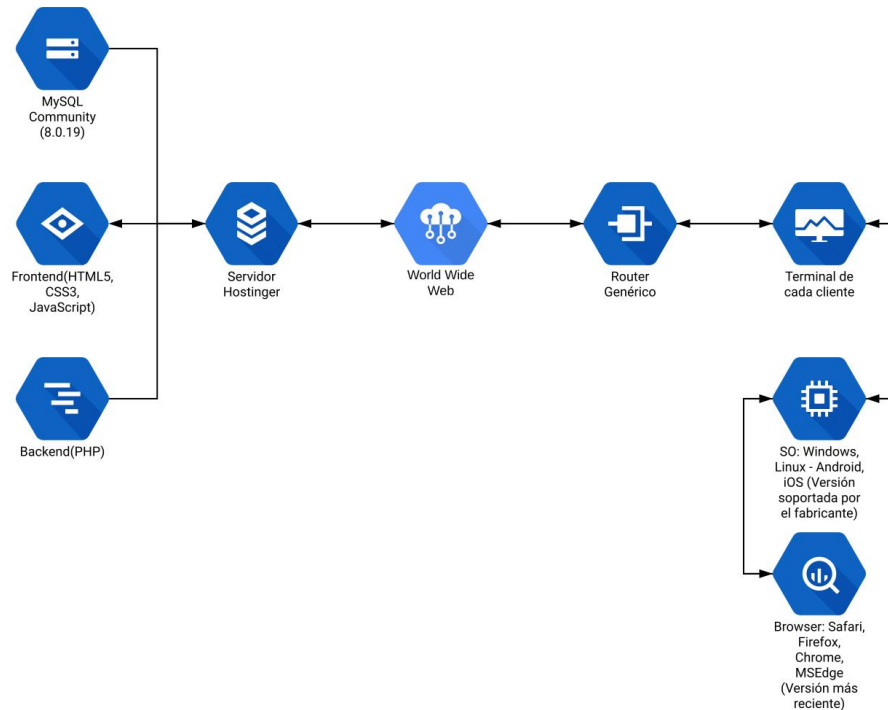
- La organización objetivo es sin fines de lucro y actualmente tiene gastos no planeados de alrededor de C100.000 colones, el sistema Tails & Paws espera reducir este monto a 40.000 colones mensuales.
- El seguimiento de adoptantes de animales es lento y poco preciso, el sistema Tails & Paws pretende manejar esto en 1 hora aproximadamente.
- Los casos de abandono o maltrato tardan 3 días en presentar una acción respecto a su reporte. El sistema Tails & Paws espera reducirlo a un tiempo de acción menor a 3 días.
- Inexistencia en el control de elementos tales como animales, productos y adopciones. Tails & Paws pretende generar un control y manejo histórico de la información.
- El Proyecto espera concretarse dentro de 1 año calendario

## **PLANIFICACIÓN DE LA ARQUITECTURA DONDE EL SISTEMA SERÁ ALOJADO**

Dentro del documento de arquitectura se encuentra el diagrama de infraestructura que estará siendo una vez la aplicación se encuentre desarrollada y la misma se encuentre en la fase de producción, alguno de los detalles que en ellos se pueden apreciar son;

- Se hará uso de MySQL como el sistema de gestión de base de datos para el sistema.
- El lenguaje de programación a utilizar será PHP

A continuación, se mostrará el diagrama de infraestructura que se encuentra en el documento:



En relación con el caso de uso **RNFF-04 Disponibilidad** se indica que el Sistema debe de contar con un sistema disponible al menos un 95% del tiempo, pero con el diagrama mostrado anteriormente no cumple con los requisitos para ser un sistema puede no contar con una disponibilidad donde se cumpla ese porcentaje de tiempo funcionando.

Se realizó una entrevista de manera virtual **PT\_06 R.J.P.G Consulting-006-2020 Entrevista** donde se capturo información importante del sistema, entre ellos cabe destacar los siguientes:

- Se estará haciendo del servicio en la nube para almacenar la información del sistema. Sin embargo, no tienen la información sobre el proveedor que se estará utilizando.
- Las actualizaciones en los servidores se estarán haciendo de manera anual.
- Cada desarrollador tendrá un acceso de administrador al sistema.

## **PLAN DE PRUEBAS**

Se asegura, que el personal de auditoria con el que cuenta el sistema está debidamente calificado para realizar una auditoría interna. Se indica que se toman medidas periódicas, de mantenimiento preventivo para asegurar el correcto funcionamiento del sistema en todo momento. Alegan que el sistema Tails & Paws se guían bajo los mejores estándares de calidad. En un comentario de autocrítica, se indicó que uno de los aspectos más vulnerables de su sistema son las faltas de plan de pruebas para así evitar caídas del sistema o monitoreos el mismo. El modelo debe facilitar la creación, uso y el compartir en forma óptima la información por parte del negocio de tal manera que se mantenga su integridad, sea flexible, funciona rentable, oportuna, segura y tolerante a fallos y poder conformar una buena documentación sobre plan de pruebas esperadas. Al no existir un plan de mitigación de riesgos, el programa corre un alto riesgo ante un eventual problema de funcionamiento los Ingenieros del Sistema Tails &Paws deben tener la capacidad de conocer y aplicar las diferentes normas, procesos y procedimientos para garantizar la calidad de los productos software, aplicando las pruebas de calidad de software necesarias para que con ellas se pueda ayudar a reducir los riesgos en las aplicaciones, logrando que se identifiquen los defectos antes de que se ejecuten, así de forma proactiva tomar decisiones que permitan hacer las actividades necesarias para mejorar las condiciones del software y ofertar un producto que satisfaga las necesidades del cliente.

## **MANEJO DE COMUNICACIONES Y CONTROL DE VERSIONES**

Como parte de la documentación brindada por el equipo de desarrollo se obtuvo acceso a las 18 minutos registradas en el periodo de Planeación del proyecto de los cuales se puede extraer la siguiente información:

- Correcta estructura de fechas y participantes de cada reunión
- Se respetan las fechas establecidas para las reuniones que se planearon en ese momento
- Existen limitantes en cuanto a las acciones pendientes asignadas a cada miembro de la reunión
- La información respecto a las comunicaciones con interesados externos es muy limitada y de difícil acceso

- Con base en la entrevista realizada al equipo de trabajo **PT\_07 R.J.P.G Consulting-007-2020 Entrevista**, se conoció que las comunicaciones externas fueron documentadas como parte de las mismas minutas internas
- Basado en el alcance del proyecto, como se puede apreciar en el documento IN03 Documento de alcance del proyecto (Project Charter), debido al tipo de proyecto que se está realizando, es normal que las comunicaciones externas sean limitadas y de poca frecuencia.

Con respecto al manejo del control de versiones del proyecto se realizó el análisis con base en los siguientes documentos:

- IN02 Estudio de factibilidad técnica tecnológica, económica y operacional
- IN03 Documento de alcance del proyecto (Project Charter)
- AN01 Documento de especificación de requerimientos
- DN01 Documento de Arquitectura de Software

Basado en la información colectada se puede entender los siguientes detalles respecto al manejo de versiones:

- Los documentos poseen tablas donde se registran los cambios realizados y las fechas en que fueron actualizados
- Existe registro de los miembros del equipo que fueron responsables de los cambios
- Basado en la documentación colectada y las diferentes comunicaciones con el equipo de trabajo, no se pudo obtener acceso a los documentos de versiones anteriores.
- Por medio de la entrevista **PT\_07 R.J.P.G Consulting-007-2020 Entrevista** realizada al equipo de trabajo se pudo conocer que, para el desarrollo del sistema, se planea utilizar un repositorio como Git para gestionar las versiones del código.

## 3. Conclusiones

### **ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES Y EXCEPCIONES**

De acuerdo con lo evaluado el Equipo de Desarrollo no cuenta con un marco metodológico idóneo para llevar a cabo algunos procedimientos de implementación, lo cual atenta contra las mejores prácticas del Ciclo de Vida de Desarrollo de Software que establece que se debe desarrollar y aplicar un marco metodológico que guíe los procesos de implementación y considere la definición de requerimientos, los estudios de factibilidad, la elaboración de diseños, la programación y pruebas, el desarrollo de la documentación, la conversión de datos y la puesta en producción, así como también el mantenimiento post implantación de la satisfacción de los requerimientos.

Si bien se observa que el sistema define de manera clara y concisa los requerimientos de forma textual, pero algunos aspectos requieren mayor claridad en su descripción, así mismos aspectos relativos a la seguridad, contención, seguridad necesitan ser expuestos de manera más clara y precisa con el fin de reducir el factor de riesgo ante situaciones que comprometan la operación del sistema.

### **DEFINICION Y CRITERIOS DEL ALCANCE DEL SISTEMA**

De acuerdo con el análisis de la información recabada el Equipo de Desarrollo del Sistema Tails & Paws, ante los anteriores retos y objetivos se enumeran como criterios de fallo:

- Falta de presupuesto
- Retiración inesperada por parte de alguno de los integrantes del proyecto
- Falta de tiempo para desarrollar el proyecto
- Falla en el software
- Falta de alguno de los proveedores
- No lograr alguno de los objetivos necesarios para completar el proyecto

Dada la anterior información extraída del Alcance (Project Charter) se concluye que la identificación de impactos e hitos por parte del equipo de desarrollo presenta elementos válidos dentro de este apartado, sin embargo, no es posible identificar la importancia de cada uno respecto a la función principal del sistema, por ejemplo, se podría interpretar dada la información suministrada que todos los riesgos representar igualdad de criticidad y que el impacto es igual.

Respecto a controles pese a que no se denotan como tales, si se mencionan objetivos estratégicos relacionados a elementos de tiempo, presupuesto, factibilidad operativa, eficiencia los cuales en el ejercicio de la función de desarrollo y posterior implementación deben ser cumplimentados en alineamiento con controles preventivos, detectivos y correctivos a la luz de los hitos declarados en el alcance.

### **PLANIFICACIÓN DE LA ARQUITECTURA DONDE EL SISTEMA SERÁ ALOJADO**

Se puede determinar que aún no se cuenta con una claridad de la infraestructura que se estará utilizando en el sistema, quizás esto se deba a que están en una primera fase de planificación pero si es importante que se realice una exhaustiva investigación sobre este tema, porque una vez el sistema se esté desarrollando se debe de ir haciendo de una arquitectura muy similar a la que quedará en producción, esto haciendo uso de un ambiente de desarrollo para luego pasarlo sin problemas a un ambiente de producción, para así evitar la menor afectación del sistema.

### **PLAN DE PRUEBAS**

Se puede determinar que uno de los riesgos encontrados fue la falta de documentación del plan de pruebas en documento de DN01 Documento de Arquitectura de Software 12020 donde se evidenció que la no le dieron importancia a hora de implementar este método para la calidad del sistema cabe agregar que es recomendable después de realizar la implementación de un sistema el realizar una auditoría, con el fin de asegurarse que todos los procedimientos para implementar un programa de pruebas están realmente cubiertos, una auditoria como estas no debe verse como una manera de encontrar errores en un sistema, por el contrario se trata de encontrar debilidades y corregirlas, para asegurar que el recurso invertido realmente cumpla su objetivo el cual es el de tener ganancias para la organización.

### **MANEJO DE COMUNICACIONES Y CONTROL DE VERSIONES**

Se puede concluir que las comunicaciones internas fueron manejadas apropiadamente en cuanto a tiempos y estructura, aunque la información de las minutas pudo haber sido más explícita en cuanto a las acciones asignadas a cada miembro. Caso contrario, las comunicaciones externas son de difícil acceso e interpretación y no favorecen un entendimiento claro de la retroalimentación e interacción en general con miembros fuera del equipo de desarrollo del proyecto.

Respecto al manejo de versiones, se puede determinar que existió un control estricto y bien documentado de las actualizaciones de los documentos y sus responsables, pero no un almacenamiento estructurado ni de fácil acceso para las versiones anteriores de la documentación. Por otra parte, se pudo conocer de la planificación existente y en progreso respecto a la utilización de repositorios y herramientas para manejo de versiones al momento del desarrollo del proyecto.



## 4. Recomendaciones

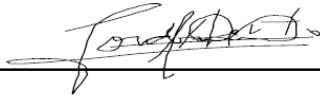
Dentro de las recomendaciones que se hacen destacan las siguientes:

- Generar un mecanismo de control que permita exponer los objetivos planteados en el alcance y cuales son lo son las debilidades, fortalezas, amenazas para posteriormente calificar los alcances basados en la prioridad dentro del sistema, a su vez divulgar esta información al cliente para una efectiva coordinación.
- Elaborar un plan de reevaluación de los requerimientos funcionales y no funcionales con el fin de concretar y especificar de manera más detallada aspectos de desviaciones, excepciones con el fin de contar con la información más actualizada además de definirse acciones a ser implementadas derivadas de esta revisión.
- Realizar un plan para definir si el aplicativo debe ser ejecutado con pruebas automatizadas.
- Detectar si los testers cuentan con el expertise acerca de la realización de pruebas automatizadas.
- Organizar que escenarios automatizar.
- Realizar un buen uso de la data generada durante pruebas de calidad anteriores y la actual.
- Los parches de seguridad y las actualizaciones no deben de revisarse de manera anual, esto porque en cualquier momento hay una vulnerabilidad y si el sistema no se actualiza como el fabricante lo esté recomendado para mitigar cualquier punto de seguridad.
- Revisar el diagrama de infraestructura para asegurar que el sistema con los requisitos para cumplir con una alta disponibilidad.
- Si todos los integrantes necesitan de acceso como administrador a los servidores está bien, pero quizás se puede revisar si realmente todos van a requerir ese acceso o se puede segregar los mismos.
- Utilización de un recurso compartido (drive, repositorio, etc) para almacenar las versiones anteriores de todos los documentos y recursos existentes en general.
- Separar las comunicaciones internas y externas o manejar una estructura diferente en las minutas que haga más explícito la interacción con interesados externos.
- Agregar nombres específicos de los responsables de las tareas asignadas en cada minuta y las acciones llevadas a cabo en cada caso



---

Guillermo Ramírez Chacón  
**Auditor**



---

Jordan Quesada Quesada  
**Auditor**



---

Randall Sánchez Rivera  
**Auditor**



---

Pedro Sánchez Esquivel  
**Auditor**