

**DISEÑO E IMPLEMENTACIÓN DE UN SOFTWARE PARA  
LA VALIDACIÓN DE REPORTES TÉCNICOS EN PRUEBAS  
DE CAMPO EN HUAWEI TECHNOLOGIES S.A.S**

(TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE INGENIERO DE  
TELECOMUNICACIONES)

CATALINA ALBA FORTICH

DIRECTOR:

ING. GERALD BREEK FUENMAYOR RIVADENEIRA, MTR.

UNIVERSIDAD SANTO TOMÁS  
FACULTAD DE INGENIERÍAS TIC  
INGENIERÍA DE TELECOMUNICACIONES  
BOGOTÁ D.C, 2025

A mi mamá Ruth Fortich González, por su amor, paciencia y por estar siempre dispuesta a acompañarme en cada etapa de este proceso. A mi hermano Miguel Eduardo Alba Fortich, por su apoyo incondicional y por impulsarme a seguir adelante. A mi familia, por creer en mí incluso en los momentos más difíciles; y a mis amigos de la universidad, por los aprendizajes, valores y experiencias compartidas que hicieron este camino más significativo.

## **AGRADECIMIENTOS**

Expreso mi sincero agradecimiento a quienes contribuyeron al desarrollo de este proyecto.

En primer lugar, a Miguel Eduardo Alba Fortich, por su apoyo en el diseño y programación del software ScanDoc, así como por su disposición para aportar conocimientos técnicos fundamentales durante el desarrollo de la solución.

A Gerald Breek Fuenmayor Rivadeneira, por su acompañamiento como tutor a lo largo del semestre, brindando orientación académica, seguimiento constante y aportes clave para la correcta estructuración de la monografía.

A Michael Luna Gómez, por su rol como supervisor durante la práctica en el área de Radiofrecuencia en Huawei Technologies S.A.S, y por compartir su experiencia y conocimientos, los cuales fueron fundamentales para comprender los procesos reales del entorno laboral y dar enfoque práctico al proyecto.

Asimismo, agradezco a todas las personas del área de Radiofrecuencia que, de manera directa o indirecta, aportaron a la realización de este trabajo, ya sea mediante asesoría técnica, retroalimentación o apoyo durante las pruebas del sistema.

## TABLA DE CONTENIDO

ACRÓNIMOS.....	6
RESUMEN.....	1
ABSTRACT.....	1
INTRODUCCIÓN.....	2
1 MARCO GENERAL DEL PROYECTO.....	3
1.1 PROBLEMA.....	3
1.2 OBJETIVOS.....	4
1.3 ALCANCE.....	4
1.4 METODOLOGÍA.....	5
1.5 CRONOGRAMA.....	8
2 MARCO REFERENCIAL.....	11
2.1 MARCO TEÓRICO.....	12
2.1.1 REDES MÓVILES 3G (UMTS).....	12
2.1.2 REDES MÓVILES 4G (LTE).....	13
2.1.3 REDES MÓVILES 5G (NR).....	13
2.1.4 PRUEBAS DE CAMPO (DRIVE TESTS).....	14
2.1.5 ESTRUCTURA DOCUMENTAL TÉCNICA EN RADIOFRECUENCIA.....	14
2.2 MARCO CONCEPTUAL.....	15
2.2.1 ARCHIVO .DOCX.....	16
2.2.2 XML (EXTENSIBLE MARKUP LANGUAGE).....	18
2.2.3 PANDOC.....	21
2.2.4 VALIDACIÓN DOCUMENTAL.....	23
2.2.5 CRITERIOS DE VALIDACIÓN.....	24
2.2.6 DRIVE TEST REPORT.....	25
2.2.7 INTERFAZ GRÁFICA (GODOT ENGINE).....	26
2.2.8 AUTOMATIZACIÓN.....	28
2.3 ESTADO DEL ARTE.....	31
3 ANÁLISIS.....	35

3.1	REQUERIMIENTOS DEL CLIENTE .....	35
3.2	REQUISITOS DEL SOFTWARE.....	36
3.2.1	REQUISITOS FUNCIONALES (RF).....	36
3.2.2	REQUISITOS NO FUNCIONALES (RNF).....	37
3.3	RESTRICCIONES Y SUPUESTOS.....	37
3.4	CRITERIOS GENERALES DE ACEPTACIÓN.....	38
4	DESARROLLO DE LA SOLUCIÓN .....	38
4.1	MÉTODO DE DESARROLLO.....	39
4.1.1	ETAPAS DE LA METODOLOGÍA RUP EN EL PROYECTO .....	40
4.2	DISEÑO DE LA SOLUCIÓN .....	41
4.2.1	ARQUITECTURA GENERAL DEL SISTEMA.....	42
4.2.2	FLUJO DE PROCESAMIENTO DE DOCUMENTOS.....	47
4.2.3	MODELO DE DATOS .....	48
4.2.4	CRITERIOS DE VALIDACIÓN .....	50
5	IMPLEMENTACIÓN DEL SISTEMA.....	51
5.1	PRUEBAS Y VALIDACIÓN.....	56
5.1.1	ESCENARIOS DE PRUEBA .....	56
5.1.2	RESULTADOS OBTENIDOS .....	62
6	CONCLUSIONES.....	63
6.1	TRABAJOS FUTUROS.....	65
	REFERENCIAS .....	67
	LISTA DE FIGURAS .....	70
	LISTA DE TABLAS .....	70

## ACRÓNIMOS

**UMTS** (Universal Mobile Telecommunications System): representa la tercera generación (3G) de comunicaciones móviles.

**RSCP** (Received Signal Code Power): medida de la potencia recibida de la señal piloto de una celda específica, expresada típicamente en dBm.

**Ec/Io** (Energy per-Chip to Interference Ratio): Relación entre la energía de la señal piloto deseada y la densidad total de interferencia, expresada en dB.

**RSRP** (Reference Signal Received Power): Potencia promedio recibida de las señales de referencia específicas de celda, medida en dBm.

**SINR** (Signal to Interference plus Noise Ratio): Relación entre la potencia de la señal útil y la suma de interferencia más ruido térmico, expresada en dB.

**Throughput**: Velocidad efectiva de transferencia de datos alcanzada en pruebas, medida típicamente mediante servidores FTP o iPerf. Este parámetro integra múltiples factores como SINR, carga celular, configuración de MIMO, agregación de portadoras (Carrier Aggregation en LTE-Advanced) y scheduling.

**Eventos de handover**: Registros de trasposos entre celdas, clasificados en intra-frequency (misma frecuencia), inter-frequency (diferente frecuencia de la misma tecnología) e inter-RAT (entre diferentes tecnologías radio como LTE-UMTS). El análisis de estos eventos permite identificar problemas de movilidad, parámetros mal configurados o necesidades de expansión.

## RESUMEN

Este documento presenta el diseño e implementación de ScanDoc, un software para la validación automatizada de reportes técnicos de pruebas de campo en el área de Radiofrecuencia de Huawei-Colombia. La necesidad surge de la revisión manual de documentos Word (LTE y UMTS), un proceso repetitivo y propenso a errores. La solución se sustenta en un marco teórico que integra redes móviles, estructura del formato .docx, procesamiento XML, conversión con Pandoc y principios de automatización, materializándose en una arquitectura modular desarrollada en Godot Engine con capacidad de análisis simultáneo de múltiples documentos. Las pruebas realizadas evidencian reducciones significativas en los tiempos de validación, confirmando la viabilidad técnica del sistema y su impacto positivo en la eficiencia y estandarización del proceso documental.

**Palabras clave:** Automatización; Validación documental; Drive Test; LTE/UMTS; DOCX

## ABSTRACT

This document presents the design and implementation of ScanDoc, a software tool for the automated validation of technical field test reports in the Radio Frequency division of Huawei-Colombia. The need for this solution arose from the manual review of Word documents (LTE and UMTS), a repetitive and error-prone process. The solution is based on a theoretical framework that integrates mobile networks, the structure of the .docx format, XML processing, conversion using Pandoc, and automation principles, resulting in a modular architecture developed in Godot Engine with the capability to simultaneously analyze multiple documents. The tests conducted demonstrate significant reductions in validation times, confirming the system's technical viability and its positive impact on the efficiency and standardization of the document process.

**Keywords:** Automation; Document validation; Drive Test; LTE/UMTS; DOCX

## INTRODUCCIÓN

Las pruebas de campo en redes móviles (UMTS y LTE) generan reportes técnicos en Word (.docx) que consolidan tablas, gráficas y secciones estandarizadas, usados como insumo para tareas de optimización y aseguramiento de calidad en Radiofrecuencia (RF). En Huawei-Colombia, la validación preliminar de estos reportes es un paso crítico que actualmente se realiza de forma manual, lo que implica tareas repetitivas, alto consumo de tiempo y riesgo de errores, afectando la agilidad en la toma de decisiones técnicas.

Para atender esta problemática, se presenta ScanDoc: un software de validación automatizada de la estructura de reportes de pruebas de campo en redes LTE y UMTS. La solución analiza documentos .docx mediante su estructura interna y conversión a formato intermedio con Pandoc, e integra una interfaz gráfica desarrollada en Godot Engine. El sistema verifica la presencia de secciones, tablas y gráficas obligatorias, actuando como filtro previo a la revisión técnica detallada y sin sustituir el criterio del ingeniero. Los resultados muestran reducciones significativas en los tiempos de validación y mayor consistencia en la aplicación de criterios.

El documento se organiza en seis capítulos: el Capítulo 1 establece el marco general del proyecto, incluyendo el problema, objetivos, alcance, metodología y cronograma; el Capítulo 2 desarrolla el marco referencial con los fundamentos teóricos y el estado del arte relevantes; el Capítulo 3 consolida los requerimientos del sistema y los criterios de aceptación definidos; el Capítulo 4 describe la arquitectura de ScanDoc, su flujo de procesamiento y modelo de datos; el Capítulo 5 presenta la implementación y los resultados obtenidos mediante escenarios de prueba; y el Capítulo 6 expone las conclusiones y posibles líneas de trabajo futuro.

## 1 MARCO GENERAL DEL PROYECTO

El presente proyecto se desarrolla en el contexto de las actividades del área de Radiofrecuencia de Huawei-Colombia, donde la validación de reportes técnicos generados durante las pruebas de campo constituye una etapa fundamental dentro del aseguramiento de la calidad de la red. Actualmente, este proceso se realiza de manera manual, lo que implica inversiones significativas de tiempo, riesgo de errores humanos y variaciones en los criterios de verificación. Ante esta necesidad, se propone el diseño e implementación de un software que permita automatizar la revisión preliminar de dichos documentos, asegurando que cumplan con los elementos estructurales requeridos antes de su análisis técnico detallado.

### 1.1 PROBLEMA

En el área de Radiofrecuencia (RF) de Huawei-Colombia se realizan constantemente pruebas de campo a nivel nacional para evaluar el desempeño y la calidad de la señal de los operadores Tigo y Telefónica en las tecnologías LTE y UMTS.

Tras la ejecución de estas pruebas, se generan múltiples reportes técnicos (por ejemplo: LTE Tigo, LTE Telefónica, UMTS Tigo, UMTS Telefónica) que deben ser revisados manualmente para verificar si contienen todos los parámetros y gráficas requeridas (potencia, señal, frecuencia, activación de MOCN/MORAN, entre otros).

Sin embargo, este proceso de revisión es repetitivo, propenso a errores humanos y consume aproximadamente revisar manualmente 4 documentos entre 20 y 30 minutos, especialmente considerando la cantidad de reportes generados a nivel nacional. La validación manual de cada documento retrasa la aprobación de resultados y la toma de decisiones en los proyectos de optimización o integración de redes.

Tanto el supervisor de área como la ingeniera responsable de este proyecto identificaron esta problemática como una de las principales causas de ineficiencia en el flujo de trabajo. En este sentido, el supervisor manifestó:

*“Se están demorando mucho en entregar la validación de los reportes”.*

*De manera similar, un integrante del equipo (Jaider De La Rosa Rondón) resaltó la necesidad de contar con un filtro previo que acelere la retroalimentación a quienes elaboran los informes:*

*“Sería bueno tener un preliminar para rechazar y solicitar correcciones de inmediato, en vez de descargar y revisar reportes uno por uno.”*

Adicionalmente, se advierte el riesgo operativo de escenarios en los que se acumulen entregas masivas (por ejemplo, más de 200 reportes en un solo día) mientras el equipo se encuentra asignado a otros proyectos, lo que podría dejar a una sola persona disponible para revisar y aprobar rápidamente un volumen crítico de documentos.

**Pregunta problema:** ¿Cómo puede un software automatizado mejorar el proceso de validación de reportes técnicos del área de Radiofrecuencia, mejorando la eficiencia y precisión de las revisiones?

## **1.2 OBJETIVOS**

Diseñar e implementar un software automatizado que mejore el proceso de validación de reportes técnicos generados en las pruebas de campo de las redes LTE y UMTS del área de Radiofrecuencia de Huawei-Colombia.

- Analizar los requerimientos técnicos y operativos necesarios para la automatización del proceso de validación de reportes técnicos.
- Diseñar el modelo de arquitectura de solución con base en el modelo de datos y los criterios de validación.
- Implementar la solución diseñada en el modelo propuesto para dar un cumplimiento a los requerimientos.

## **1.3 ALCANCE**

El proyecto se desarrollará en el área de Radiofrecuencia (RF) de Huawei-Colombia, centrado exclusivamente en el proceso de validación de reportes técnicos generados tras las pruebas de campo de las redes LTE y UMTS de los operadores Tigo y Telefónica.

El sistema automatizado tendrá como alcance funcional:

- Identificar automáticamente si un reporte está completo o si le faltan gráficas, tablas o parámetros esenciales.
- Generar un informe resumen que indique los reportes que requieren revisión o corrección.
- Permitir la integración futura con algoritmos de inteligencia artificial para análisis más profundo (sin que esto sea parte del alcance inicial).

El desarrollo contemplará únicamente un módulo de validación de reportes, el cual será probado y validado con tres profesionales del área: supervisor, tutor de trabajo y compañero desarrollador.

Este proyecto no abarca la revisión de reportes de otras áreas (como transporte o energía), ni la automatización de análisis de desempeño avanzado de red; su alcance se limita al control de completitud y formato de los documentos.

## **1.4 METODOLOGÍA**

El presente proyecto se enmarca en un estudio de caso descriptivo con variables de trabajo cuantitativas y cualitativas, orientado al análisis y mejora del proceso manual de revisión de reportes técnicos en el área de Radiofrecuencia (RF) de Huawei-Colombia. A partir del diagnóstico de la situación actual, se propone el desarrollo de una herramienta de automatización que optimice el flujo de validación de dichos reportes, reduciendo tiempos de revisión y aumentando la precisión en la detección de omisiones o fallas estructurales. Las fuentes de información utilizadas provendrán de los propios reportes técnicos generados por el área de RF, tanto en tecnologías LTE como UMTS para los operadores Tigo y Telefónica. Además, se contemplarán entrevistas y observaciones con el supervisor, el tutor de trabajo y un compañero desarrollador del área, quienes participarán activamente en la validación de los avances y resultados. También se empleará documentación técnica interna sobre formatos, parámetros de validación y manuales corporativos, junto con

fuentes académicas recientes relacionadas con la automatización de procesos documentales y la gestión de reportes técnicos.

La población objeto de estudio corresponde al personal del área de RF de Huawei-Colombia involucrado directamente en el proceso de validación de reportes, mientras que la muestra estará compuesta por tres participantes seleccionados de manera no probabilística por conveniencia: el supervisor de área, el tutor de trabajo y un compañero desarrollador. Su participación permitirá obtener una retroalimentación técnica constante durante las fases de desarrollo.

El proyecto seguirá una metodología de desarrollo iterativa, dividida en tres fases principales: **(1)** análisis de requerimientos y diagnóstico del proceso actual de validación, **(2)** diseño del modelo de arquitectura y definición de criterios de validación, y **(3)** implementación y pruebas del módulo automatizado. Cada una de estas fases será revisada y aprobada por los tres participantes mencionados, garantizando que las decisiones técnicas sean coherentes con las necesidades reales del área.

En cuanto a las herramientas de captura, análisis y presentación, la etapa de captura se basará en la lectura directa de reportes técnicos reales en formato Word, los cuales siguen plantillas internas de la empresa. Para el procesamiento, el sistema empleará Godot Engine (en su lenguaje GDScript) como interfaz principal y orquestador del flujo de trabajo. Desde Godot se gestionará la interacción con el usuario, la cola de archivos a validar y la visualización de resultados. Se integrarán o desarrollarán plugins específicos que faciliten la lectura y el análisis de los archivos XML embebidos en los documentos Word (.docx), con el fin de identificar estructuras internas como tablas, secciones o bloques gráficos.

Adicionalmente, se empleará Pandoc como herramienta de conversión intermedia, encargada de transformar los archivos Word o sus componentes XML en formato HTML, lo que permitirá realizar un parseo estructurado del contenido. A partir de la estructura HTML generada, el sistema podrá detectar la presencia o ausencia de los elementos requeridos—como gráficos, tablas o secciones específicas—, los cuales serán procesados mediante scripts y extensiones en Godot.

El flujo de trabajo propuesto se resume en las siguientes etapas: extracción del contenido XML desde los archivos .docx, conversión mediante Pandoc a HTML, análisis y detección de elementos dentro de Godot, y generación de resultados automáticos. Para

complementar el proceso, se implementarán scripts ligeros encargados de las operaciones de archivo, el registro básico de eventos y la generación de reportes resumen. Finalmente, la información validada será presentada a través de dashboards de control y reportes automáticos que permitirán visualizar de forma clara los resultados de cada validación.

En cuanto a relación método – objetivos:

Objetivos	Actividades	Herramientas	Producto esperado
<b>1. Análisis de requerimientos</b>	<ul style="list-style-type: none"> <li>- Revisión de las plantillas Word utilizadas por el área de RF.</li> <li>- Entrevistas con el supervisor, tutor y compañero desarrollador.</li> <li>- Identificación y mapeo de campos obligatorios (tablas, gráficas, secciones).</li> <li>- Especificación de requerimientos.</li> </ul>	Microsoft Word, entrevistas estructuradas, documentación técnica interna, diagramas de flujo.	Especificación y priorización de requisitos funcionales y no funcionales (RF y RNF).
<b>2. Diseño del modelo de arquitectura</b>	<ul style="list-style-type: none"> <li>- Historias de usuario (DCU).</li> <li>- Elaboración del flujo de conversión (Word → XML → HTML).</li> <li>- Modelado de la estructura de datos y definición de reglas de validación.</li> <li>- Identificación de plugins y funciones necesarias en Godot.</li> <li>- Modelo de despliegue.</li> </ul>	Godot Engine (diseño de estructura), Pandoc (para conversión de formatos), XML/HTML viewers, diagramas UML.	Documento del modelo de arquitectura, modelo de datos, reglas de validación definidas y diseño de interfaz/módulos.
<b>3. Implementación</b>	- Implementación del software en Godot e integración con	Godot Engine (GDScript),	Prototipo funcional del

Objetivos	Actividades	Herramientas	Producto esperado
<b>y validación del producto</b>	Pandoc y plugins de lectura XML. - Ejecución de pruebas con los tres usuarios seleccionados. - Comparación de métricas: tiempo de validación manual vs. automatizada y porcentaje de detección de faltantes.	Pandoc, datasets de reportes técnicos reales, hojas de cálculo para análisis de resultados.	validador de reportes, métricas de desempeño, informe de pruebas y validación preliminar.

Tabla 1: Actividades de los objetivos

En conjunto, este enfoque metodológico permitirá demostrar la viabilidad técnica y funcional del sistema propuesto, asegurando que la automatización planteada se adapte a las necesidades reales del área de Radiofrecuencia y contribuya a mejorar el control de calidad de los reportes técnicos.

## 1.5 CRONOGRAMA

Fase / Actividad principal	Responsables	Fechas
1. Identificación de la problemática	Catalina Alba Fortich	13 – 21 septiembre 2025
	Michael Luna Gómez	
1.1 Revisión del proceso actual y detección de oportunidades de mejora	Catalina Alba Fortich	13 – 21 septiembre 2025
2. Estudio y selección de la propuesta	Catalina Alba Fortich	22 septiembre – 5 octubre 2025
	Jaidier De La Rosa Rondón	

2.1 Evaluación de alternativas y selección del tema final	Catalina Alba Fortich	22 septiembre – 5 octubre 2025
3. Análisis de requerimientos técnicos	Catalina Alba Fortich	6 – 19 octubre 2025
	Jaider De La Rosa Rondón	
	David Alfonso Vallejo Muñoz	
3.1 Revisión de plantillas Word y mapeo de campos obligatorios	Catalina Alba Fortich	6 – 12 octubre 2025
3.2 Definición de requerimientos funcionales y no funcionales	Catalina Alba Fortich	13 – 16 octubre 2025
	Gerald Breek Fuenmayor Rivadeneira	
3.3 Definición del flujo de procesamiento (.docx → XML → HTML)	Catalina Alba Fortich	17 – 19 octubre 2025
	Miguel Eduardo Alba Fortich	
4. Diseño del sistema	Catalina Alba Fortich	20 octubre – 20 noviembre 2025
	Miguel Eduardo Aba Fortich	
4.1 Diseño del modelo de arquitectura	Catalina Alba Fortich	20 – 27 octubre 2025
	Miguel Eduardo Alba Fortich	
4.2 Modelo de datos y criterios de validación	Catalina Alba Fortich	28 octubre – 2 noviembre 2025
4.3 Diseño de la interfaz básica en Godot	Catalina Alba Fortich	3 – 10 noviembre 2025
	Miguel Eduardo Alba Fortich	
4.4 Validación del diseño con los tutores técnicos	Catalina Alba Fortich	10 – 20 noviembre 2025
	Michael Luna Gómez	

	Jaider De La Rosa Rondón	
5. Implementación y pruebas	Catalina Alba Fortich	21 noviembre – 1 diciembre 2025
	Michael Luna Gómez	
	Jaider De La Rosa Rondón	
	5 sujetos de prueba adicionales	
5.1 Desarrollo del producto en Godot	Catalina Alba Fortich	21 – 28 noviembre 2025
5.2 Integración con Pandoc y plugins XML	Catalina Alba Fortich	21 – 23 noviembre 2025
	Miguel Eduardo Alba Fortich	
5.3 Pruebas funcionales con reportes reales	Catalina Alba Fortich	23 noviembre – 1 diciembre 2025
	Michael Luna Gómez	
	David Alfonso Vallejo Muñoz	
	Jaider De La Rosa Rondón	
6. Documentación final	Catalina Alba Fortich	11 noviembre 2025 – 9 enero 2026
6.1 Redacción del informe técnico y académico	Catalina Alba Fortich	11 noviembre 2025 – 9 enero 2026
7. Entrega y cierre del proyecto	Catalina Alba Fortich	9 enero – 16 enero 2026
	Gerald Breek Fuenmayor Rivadeneira	
7.1 Revisión final	Catalina Alba Fortich	9 – 14 enero 2026
	Gerald Breek Fuenmayor Rivadeneira	



## 2.1 MARCO TEÓRICO

El marco teórico reúne los fundamentos técnicos necesarios para comprender el contexto tecnológico en el cual se desarrollan los reportes técnicos validados por el software propuesto. Estos elementos permiten contextualizar cómo la infraestructura y la operación de redes móviles influyen en la estructura y contenido de los documentos generados en las pruebas de campo.

### 2.1.1 REDES MÓVILES 3G (UMTS)

La tecnología **UMTS** (Universal Mobile Telecommunications System) representa la tercera generación (**3G**) de comunicaciones móviles, estandarizada por el 3GPP (3rd Generation Partnership Project) en sus releases iniciales [16].

UMTS opera típicamente en bandas licenciadas como 850 MHz (Banda 5), 1900 MHz (Banda 2) y 2100 MHz (Banda 1), aunque existen implementaciones en otras frecuencias según la región geográfica y las asignaciones espectrales de cada país. El acceso múltiple se realiza mediante WCDMA (Wideband Code Division Multiple Access), técnica que permite a múltiples usuarios compartir el mismo espectro de frecuencia mediante códigos ortogonales de esparcimiento, con un ancho de banda de portadora de 5 MHz.

Los reportes asociados a UMTS suelen incluir:

- Intensidad de señal (**RSCP**)
- Calidad de señal (**Ec/Io**)
- Bandas y portadoras
- Identificadores de celdas (**PSC**), que son códigos de 0 a 511 que identifica unívocamente una celda dentro de un área de planificación radio. El PSC permite al equipo de usuario diferenciar entre señales provenientes de distintas celdas que operan en la misma frecuencia, siendo fundamental para los procesos de selección y reelección celular.

Estos parámetros permiten evaluar la calidad de cobertura y desempeño durante drive tests.

## 2.1.2 REDES MÓVILES 4G (LTE)

La tecnología LTE (Long-Term Evolution) define la cuarta generación (4G) de redes móviles, representando un cambio paradigmático en la arquitectura de telecomunicaciones al establecer una red completamente basada en conmutación de paquetes IP (IP Packet Switching), eliminando el dominio de conmutación de circuitos presente en generaciones anteriores. Estandarizada inicialmente en el 3GPP Release 8 y evolucionada en releases posteriores [17], LTE fue diseñada con el objetivo de proporcionar mayores velocidades de datos, menor latencia y mayor eficiencia espectral en comparación con las tecnologías 3G.

Los reportes LTE incluyen parámetros fundamentales como:

- Potencia y calidad de señal (**RSRP, SINR**)
- Frecuencias y anchos de banda (**EARFCN**), que es un identificador numérico que especifica la frecuencia de portadora LTE, con un esquema de numeración que permite identificar tanto la banda como la frecuencia específica dentro de ella.
- Identificadores como **PCI**, identificador de 0 a 503 que permite al equipo de usuario distinguir entre diferentes celdas. La planificación adecuada de PCIs es fundamental para evitar colisiones y confusiones que puedan degradar el rendimiento de la red.
- Medición de throughput y eventos de handover.

El análisis de estos parámetros es clave para evaluar el rendimiento del radioenlace.

## 2.1.3 REDES MÓVILES 5G (NR)

La tecnología **5G NR** (New Radio) introduce capacidades avanzadas como:

- Uso de bandas sub-6 GHz y ondas milimétricas (mmWave)
- Soporte para MIMO masivo
- Numerologías flexibles
- Anchos de banda superiores (hasta 100 MHz en sub-6 y 400 MHz en mmWave) [18]

Aunque los reportes técnicos del área de estudio aún no profundizan exhaustivamente en pruebas de campo 5G debido a la fase temprana de despliegue en la región, los conceptos teóricos de esta tecnología son fundamentales para comprender la evolución de los requisitos de validación documental en el mediano y largo plazo.

#### **2.1.4 PRUEBAS DE CAMPO (DRIVE TESTS)**

Las pruebas de campo, comúnmente denominadas drive tests en la terminología de la industria de telecomunicaciones, constituyen una metodología fundamental para la evaluación empírica del rendimiento de redes móviles en condiciones reales de operación [19]. A diferencia de las simulaciones teóricas o mediciones estacionarias, los drives tests permiten capturar la experiencia real del usuario en movilidad a través de áreas geográficas específicas, proporcionando datos críticos para la planificación, optimización, validación y resolución de problemas en redes inalámbricas.

- Cobertura
- Calidad de señal
- Velocidades de datos
- Eventos (handover, reelección, llamadas fallidas)

Los reportes resultantes incluyen tablas y gráficas que documentan los resultados obtenidos, siendo estos justamente los elementos que tu software debe detectar y validar.

#### **2.1.5 ESTRUCTURA DOCUMENTAL TÉCNICA EN RADIOFRECUENCIA**

Los documentos técnicos en el sector de telecomunicaciones, particularmente aquellos que documentan pruebas de campo, auditorías de red, o proyectos de implementación, siguen convenciones estructurales específicas que han evolucionado como mejores prácticas de la industria. Esta estandarización, aunque no siempre rige bajo normas formales escritas, responde a necesidades prácticas de comunicación técnica efectiva, trazabilidad, reproducibilidad y cumplimiento regulatorio.

Los reportes de pruebas de campo en el caso de LTE suelen incluir:

- Portada
- Resumen de KPIs inicial
- Tabla de Contenidos
- Basic Information
  - Test Plan Information
  - Drive Test Route
- LTE PS Test High Band
- LTE VoLTE Intermediate Call Test High Band
- LTE PS Test Low Band
- LTE VoLTE Intermediate Call Test Low Band
- Optimization & Analysis Summary

Cada bloque 5–8 contiene 6 submódulos esenciales:

- Statistical Result
- PCC Rank
- RLC Throughput DL
- PCC Avg CQI
- RLC Throughput UL
- Serving PCI / EARFCN / RSRP / SINR
- Abnormal Events

El software del proyecto tiene como fin verificar que estos componentes estén presentes antes de la validación técnica detallada.

## **2.2 MARCO CONCEPTUAL**

El marco conceptual agrupa y define los términos, conceptos y tecnologías clave que sustentan la comprensión integral del proyecto de validación documental automatizada.

Estos conceptos abarcan tanto el dominio de las telecomunicaciones como el procesamiento computacional de documentos, conformando el lenguaje técnico común necesario para el desarrollo, implementación y operación del sistema propuesto.

### 2.2.1 ARCHIVO .DOCX

El formato .docx (Office Open XML Document) representa el estándar contemporáneo para documentos de procesamiento de texto, introducido por Microsoft en Office 2007 como sucesor del formato binario .doc utilizado en versiones anteriores de Microsoft Word. Este formato fue estandarizado posteriormente por ECMA International como ECMA-376 y por ISO/IEC como ISO/IEC 29500, convirtiéndose en un estándar internacional abierto ampliamente adoptado más allá del ecosistema Microsoft [20].

#### **Arquitectura interna del formato .DOCX**

Un archivo .docx es, en esencia, un archivo comprimido en formato ZIP que contiene una estructura jerárquica de carpetas y archivos XML interrelacionados. Esta arquitectura puede explorarse descomprimiendo el archivo .docx con cualquier herramienta de compresión estándar, revelando su estructura interna:

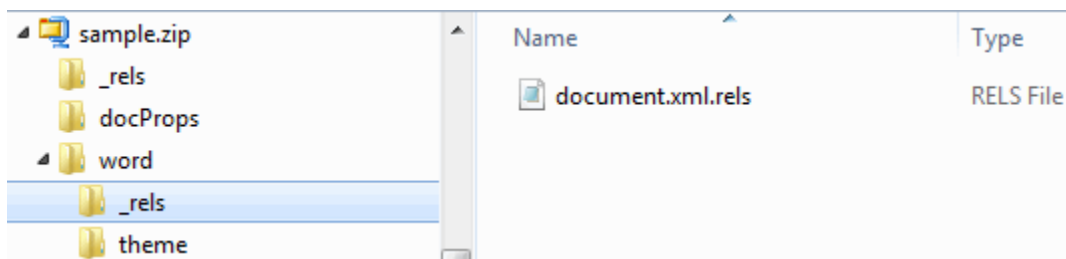


Figura 2: Ejemplo visual de organización de datos de un documento Word

Donde los componentes fundamentales:

**[Content\_Types].xml:** Archivo raíz que define los tipos MIME de todos los componentes del documento, permitiendo al software procesador identificar cómo interpretar cada parte. Incluye declaraciones como `<Default Extension="xml" ContentType="application/xml"/>` y mappings específicos para relaciones, imágenes, y componentes especializados.

**\_rels/ (Relationships):** Carpeta que contiene archivos de relaciones (.rels) que definen cómo los diferentes componentes del documento se vinculan entre sí. El archivo principal.rels establece la relación con word/document.xml como el documento principal. El archivo word/\_rels/document.xml.rels define relaciones entre el documento y recursos externos como imágenes, hipervínculos, o estilos.

**word/document.xml:** El archivo central que contiene el contenido estructural del documento: párrafos, tablas, imágenes (como referencias), listas, y toda la estructura semántica del texto. Este archivo utiliza un vocabulario XML específico de WordprocessingML con elementos como:

- `<w:p>` (*paragraph*): Define un párrafo
- `<w:r>` (*run*): Define una secuencia de texto con formato homogéneo
- `<w:t>` (*text*): Contenido textual literal
- `<w:tbl>` (*table*): Define una tabla
- `<w:tr>` (*table row*): Fila de tabla
- `<w:tc>` (*table cell*): Celda de tabla

### **Ventajas del formato .DOCX**

- **Interoperabilidad:** Al ser un estándar abierto, múltiples aplicaciones pueden leer y escribir archivos .docx: LibreOffice, Google Docs, Apple Pages, y numerosas bibliotecas de programación (python-docx, Apache POI, Mammoth, etc.).
- **Compresión:** El uso de ZIP reduce significativamente el tamaño de archivos, especialmente aquellos con múltiples imágenes de alta resolución.
- **Separación de contenido y presentación:** La arquitectura XML permite separar el contenido semántico (qué dice el texto) de su presentación visual (cómo se ve), facilitando transformaciones, búsquedas, y procesamiento automatizado.
- **Extensibilidad:** El formato permite incluir metadatos personalizados, componentes especializados (ecuaciones matemáticas con MathML, diagramas con DrawingML), y extensiones propietarias sin romper la compatibilidad base.

### **Relevancia para el proyecto**

El proyecto de validación documental debe interpretar la estructura interna del archivo .docx para extraer y analizar sus componentes. Específicamente, el sistema necesita:

- Parsear document.xml para identificar párrafos, tablas, y su estructura jerárquica
- Interpretar estilos para reconocer secciones estructurales (títulos, subtítulos)
- Extraer texto de tablas para validar contenidos específicos esperados.

Esta comprensión profunda del formato .docx es fundamental para diseñar algoritmos de validación robustos que no dependan de la renderización visual del documento sino de su estructura semántica subyacente, permitiendo validaciones consistentes independientemente del software utilizado para crear el documento.

### **2.2.2 XML (EXTENSIBLE MARKUP LANGUAGE)**

XML (eXtensible Markup Language) es un lenguaje de marcado extensible diseñado para almacenar y transportar datos de manera estructurada, jerárquica y legible tanto para humanos como para máquinas [21]. Desarrollado por el W3C (World Wide Web Consortium) como simplificación de SGML (Standard Generalized Markup Language), XML se ha convertido en uno de los pilares fundamentales del intercambio de información en la era digital.

#### **Características fundamentales de XML**

**Autodescriptivo:** XML utiliza etiquetas (tags) que describen el significado del contenido que encapsulan, no solo su presentación. Por ejemplo:

```
<estudiante>
  <nombre>Juan Pérez</nombre>
  <edad>22</edad>
  <carrera>Ingeniería de Telecomunicaciones</carrera>
</estudiante>
```

**Jerárquico:** Los elementos XML se organizan en estructuras de árbol con un elemento raíz único y relaciones padre-hijo claramente definidas, permitiendo representar relaciones complejas de manera natural.

**Estrictamente estructurado:** XML requiere sintaxis precisa:

- Todas las etiquetas deben cerrarse explícitamente (<tag></tag> o <tag/>)
- Los nombres de etiquetas son sensibles a mayúsculas/minúsculas
- Los valores de atributos deben ir entre comillas
- Los elementos deben anidarse correctamente sin solapamientos

**Extensible:** No existe un conjunto predefinido de etiquetas; cada aplicación puede definir su propio vocabulario XML según sus necesidades específicas. Esto permite adaptar XML a dominios tan diversos como finanzas (XBRL), documentos de oficina (OOXML), telecomunicaciones (XML de configuración de equipos), ciencias (MathML para matemáticas), y muchos otros.

## Componentes sintácticos de XML

**Declaración XML:** Opcional pero recomendada, especifica la versión de XML y la codificación de caracteres:

```
<?xml version="1.0" encoding="UTF-8"?>
```

**Elementos:** Unidad fundamental de información, definida por etiquetas de apertura y cierre:

```
<elemento>contenido</elemento>  
<elementoVacio/>
```

**Atributos:** Pares nombre-valor asociados a elementos que proporcionan metadatos adicionales:

```
<celda id="Cell_001" tecnologia="LTE" banda="AWS">
```

**Texto:** Contenido literal dentro de elementos.

**Comentarios:** Anotaciones no procesadas:

```
<!-- Este es un comentario explicativo -->
```

**CDATA:** Secciones donde los caracteres especiales de XML no se interpretan, útil para incluir código o datos con caracteres reservados:

```
<![CDATA[Texto con <, >, & sin escape]]>
```

**Espacios de nombres (Namespaces):** Permiten evitar conflictos cuando se combinan vocabularios XML de diferentes orígenes:

```
<w:document xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main">
```

## Parsing y procesamiento de XML

El procesamiento de XML puede realizarse mediante dos paradigmas principales:

**DOM (Document Object Model):** Carga todo el documento XML en memoria como estructura de árbol, permitiendo navegación bidireccional y modificaciones. Ventajoso para documentos pequeños-medianos donde se requiere acceso aleatorio a múltiples partes. Desventajoso para documentos muy grandes por consumo de memoria.

**SAX (Simple API for XML):** Procesamiento secuencial basado en eventos (evento de inicio de elemento, evento de texto, evento de fin de elemento). Eficiente en memoria pues no carga todo el documento, pero limitado a procesamiento lineal unidireccional.

**Bibliotecas modernas:** Lenguajes contemporáneos incluyen bibliotecas robustas:

- Python: xml.etree.ElementTree, lxml
- JavaScript: DOMParser, xml2js
- Java: JAXP, DOM4J
- C#: System.Xml, LINQ to XML

## Aplicación en el proyecto

En el contexto del proyecto de validación documental, XML es el lenguaje fundamental que estructura los archivos .docx internamente. El sistema debe:

**Parsear document.xml** para extraer la estructura del documento:

```
<w:document>
  <w:body>
    <w:p><!-- párrafo --></w:p>
    <w:tbl><!-- tabla --></w:tbl>
  </w:body>
</w:document>
```

**Navegar el árbol XML** para identificar elementos específicos:

- Contar tablas mediante búsqueda de elementos <w:tbl>
- Identificar imágenes mediante elementos <w:drawing> o <w:pict>
- Reconocer títulos mediante estilos <w:pStyle w:val="Heading1">

**Extraer texto** de elementos específicos para validación de contenido:

```
<w:tc>
  <w:p>
    <w:r>
      <w:t>RSRP (dBm)</w:t>
    </w:r>
  </w:p>
</w:tc>
```

**Interpretar relaciones** mediante archivos. Rels para localizar imágenes referenciadas.

La capacidad de manipular XML eficientemente es por tanto una competencia técnica central para implementar el sistema de validación, requiriendo comprensión tanto de la sintaxis XML como de los vocabularios específicos de WordprocessingML utilizados en documentos .docx.

### 2.2.3 PANDOC

Pandoc es una herramienta de conversión de documentos universal ampliamente reconocida en la comunidad académica y técnica, desarrollada por John MacFarlane en el lenguaje de programación Haskell. Frecuentemente denominada el "cuchillo suizo" de los conversores de documentos, Pandoc soporta transformaciones entre más de 40 formatos diferentes, incluyendo Markdown, HTML, LaTeX, PDF, DOCX, EPUB, MediaWiki, y muchos otros [22].

#### Características principales

- **Conversión bidireccional:** Pandoc no solo convierte desde un formato origen hacia múltiples destinos, sino que puede también leer múltiples formatos y convertirlos entre sí, actuando como traductor universal de formatos de documentos.
- **Preservación de estructura:** Durante las conversiones, Pandoc intenta preservar la estructura semántica del documento (encabezados, listas, tablas, énfasis, enlaces) independientemente de cómo se represente visualmente en cada formato específico.
- **Extensibilidad mediante filtros:** Pandoc permite la creación de filtros personalizados (en Lua, Python, o Haskell) que pueden modificar el AST (Abstract Syntax Tree) del documento durante la conversión, permitiendo transformaciones personalizadas complejas.

### Funcionamiento interno

El proceso de conversión de Pandoc opera en tres fases:

- **Parsing (Análisis):** El documento origen se parsea en una representación intermedia abstracta (AST) que captura la estructura semántica del documento independientemente de su formato específico.
- **Transformación (opcional):** Se pueden aplicar filtros que modifican el AST, agregando, eliminando, o transformando elementos.
- **Writing (Escritura):** El AST se serializa al formato de destino especificado, aplicando las reglas de sintaxis y formato específicas de ese formato.

Esta arquitectura de tres fases permite que Pandoc soporte N formatos de entrada y M formatos de salida con solo N parsers y M writers, en lugar de requerir N×M conversores específicos.

### Uso en el proyecto

En el contexto del proyecto de validación documental, Pandoc se utiliza específicamente para transformar documentos .docx a HTML:

```
pandoc documento.docx to documento.html
```

### Ventajas de esta conversión:

- **Simplificación del análisis:** HTML es más simple de procesar que la estructura XML compleja de .docx. Herramientas estándar de parsing HTML (BeautifulSoup en Python, Cheerio en Node.js, DOMParser en navegadores) están ampliamente disponibles y optimizadas.
- **Preservación de estructura:** Pandoc convierte tablas .docx en elementos <table> HTML, imágenes en <img>, títulos en <h1>, <h2>, etc., manteniendo la jerarquía semántica que el sistema de validación necesita analizar.
- **Facilita detección de componentes:** En HTML, identificar elementos es directo:
  - Contar tablas: `document.querySelectorAll('table').length`
  - Identificar imágenes: `document.querySelectorAll('img').length`
  - Extraer títulos: `document.querySelectorAll('h1, h2, h3')`

#### **Limitaciones y consideraciones:**

- **Pérdida de formato visual:** Colores específicos, fuentes exactas, o posicionamiento preciso pueden no preservarse completamente, pero para propósitos de validación estructural esto no es relevante ya que el sistema valida contenido y estructura, no apariencia.
- **Imágenes embebidas:** Pandoc puede extraer imágenes a una carpeta separada o codificarlas en base64 dentro del HTML, requiriendo configuración adecuada según las necesidades del sistema.
- **Tablas complejas:** Tablas con celdas fusionadas, anidamiento, o formato muy complejo pueden convertirse con pérdida parcial de estructura, requiriendo validación robusta que tolere variaciones.

A pesar de estas limitaciones, Pandoc representa una herramienta extremadamente valiosa para el proyecto al simplificar dramáticamente el procesamiento de documentos .docx, permitiendo enfocarse en la lógica de validación en lugar de la complejidad de parsing de formatos propietarios.

#### **2.2.4 VALIDACIÓN DOCUMENTAL**

La validación documental, en el contexto de documentos técnicos de telecomunicaciones, constituye el proceso sistemático mediante el cual se verifica que un documento cumpla con un conjunto predefinido de requisitos estructurales, de contenido, y de calidad antes de considerarlo apto para revisión técnica detallada o aprobación final. Este enfoque coincide con los principios de validación estructural descritos en [25].

Es fundamental distinguir el alcance de la validación documental automatizada propuesta en este proyecto versus la revisión técnica profunda:

El sistema **SÍ** valida:

- Presencia de tablas, gráficas, secciones requeridas
- Estructura general del documento
- Completitud formal del reporte

El sistema **NO** valida (requiere revisión humana experta):

- Si los valores de RSRP son aceptables técnicamente
- Si las recomendaciones propuestas son apropiadas
- Si la metodología utilizada es óptima para los objetivos
- Si las conclusiones están correctamente fundamentadas

El sistema actúa como un filtro de primera línea que asegura que solo reportes estructuralmente completos lleguen a revisión técnica detallada, mejorando el flujo de trabajo general del área de validación de reportes.

## **2.2.5 CRITERIOS DE VALIDACIÓN**

Los criterios de validación constituyen el conjunto de reglas específicas, medibles y verificables que definen los requisitos que un documento debe cumplir para considerarse válido. Estos criterios traducen requisitos de negocio y técnicos en condiciones comprobables algorítmicamente.

### **Características de criterios de validación efectivos**

**Específicos:** Deben ser inequívocos y precisos, evitando ambigüedad.

- Incorrecto: "El documento debe tener suficientes tablas"
- Correcto: "El documento debe contener al menos 3 tablas"

**Medibles:** Deben poder evaluarse objetivamente mediante procesamiento automatizado.

- "Número de elementos <table> en HTML  $\geq 3$ "
- "Existencia de imagen con alt-text conteniendo 'RSRP'"

**Relevantes:** Deben relacionarse directamente con la calidad o completitud del documento.

### **Tipos de criterios implementados en el proyecto**

**Criterios de conteo:**

- "Debe contener al menos tener datos en las tablas"
- "Debe incluir mínimo 5 gráficas/imágenes"

El diseño del sistema debe facilitar la adición, modificación, o eliminación de criterios sin requerir rediseño arquitectónico significativo.

### **2.2.6 DRIVE TEST REPORT**

El Drive Test Report (Reporte de Pruebas de Campo) es el producto documental resultante de la ejecución de pruebas de medición in situ en redes de telecomunicaciones móviles. Este documento técnico especializado constituye la evidencia formal de las mediciones realizadas, el análisis de resultados, y las recomendaciones derivadas del análisis de rendimiento de la red evaluada. Incluye gráficos de:

- RSRP / RSCP
- SINR / Ec/Io
- Throughput

El Drive Test Report es precisamente el tipo de documento que el sistema de validación automatizada debe procesar. El sistema debe reconocer la estructura estándar de estos reportes y verificar que incluyan los componentes mínimos requeridos para considerarse completos y listos para revisión técnica detallada.

## 2.2.7 INTERFAZ GRÁFICA (GODOT ENGINE)

Godot Engine es un motor de videojuegos de código abierto (open source) multiplataforma que, aunque diseñado originalmente para desarrollo de videojuegos 2D y 3D, es ampliamente utilizado también para desarrollo de aplicaciones de escritorio con interfaces gráficas debido a su arquitectura flexible, sistema de escenas/nodos intuitivos, y capacidades robustas de UI [23].

### Características principales de Godot

**Open source y gratuito:** Licenciado bajo MIT, Godot es completamente gratuito incluso para uso comercial, sin royalties, restricciones, o costos ocultos. El código fuente está disponible públicamente, permitiendo auditoría, modificaciones, y contribuciones comunitarias.

**Multiplataforma:** Godot permite desarrollar una vez y exportar a múltiples plataformas:

- Escritorio: Windows, macOS, Linux
- Móvil: Android, iOS
- Web: HTML5/WebAssembly
- Consolas: (mediante licencias de fabricante)

**Sistema de escenas y nodos:** Arquitectura basada en árboles de nodos donde cada elemento (botón, etiqueta, panel) es un nodo con propiedades y comportamientos específicos. Las escenas son composiciones reutilizables de nodos que pueden instanciarse múltiples veces.

**GDScript:** Lenguaje de scripting propio similar a Python, diseñado específicamente para Godot, que ofrece sintaxis simple, tipado opcional, e integración perfecta con el motor. Alternativamente soporta C#, C++, y otros lenguajes mediante GDNative.

**Editor visual integrado:** IDE completo con editor de escenas visual, debugger, profiler, gestor de recursos, y preview en tiempo real que acelera dramáticamente el desarrollo.

**UI robusta:** Nodos especializados para interfaces gráficas:

- Contenedores (VBoxContainer, HBoxContainer, GridContainer)
- Controles (Button, LineEdit, TextEdit, FileDialog)
- Display (Label, RichTextLabel, TextureRect)
- Layout automático responsive

### **Uso en el proyecto de validación documental**

Godot se utiliza como plataforma para construir la aplicación de escritorio del validador documental, proporcionando:

#### **Interfaz de usuario intuitiva:**

- Botón/área para seleccionar archivo .docx a validar (mediante FileDialog)
- Área de visualización de resultados de validación
- Indicadores visuales de estado (validación en progreso, completada, errores)
- Lista de criterios cumplidos/incumplidos

**Gestión de archivos:** Nodos como FileDialog simplifican la selección de archivos por el usuario, manejando automáticamente diferencias entre sistemas operativos.

**Invocación de lógica de validación:** Desde GDScript se pueden ejecutar scripts de Python (mediante OS.execute() o HTTPRequest a servicios locales) que implementan la lógica de parsing y validación del .docx.

**Extensibilidad futura:** La arquitectura modular de Godot facilita agregar funcionalidades adicionales:

- Configuración de criterios mediante UI

- Historial de validaciones
- Comparación de múltiples reportes
- Generación de reportes de validación en PDF

### **Alternativas consideradas**

Otras opciones para desarrollo de la interfaz gráfica incluían:

- **Electron:** Framework para aplicaciones web empaquetadas como apps de escritorio. Más pesado (requiere empaquetar Chromium) pero familiar para desarrolladores web.
- **Qt:** Framework C++/Python robusto para aplicaciones de escritorio. Más complejo, pero extremadamente poderoso.
- **Tkinter (Python):** Biblioteca estándar de Python para GUI. Simple, pero con capacidades visuales limitadas.
- **WPF/WinForms:** Específicos de Windows, limitando portabilidad.

Godot fue seleccionado por su balance entre facilidad de desarrollo, capacidades visuales atractivas, multiplataforma, y ser open source sin costos de licencia.

### **2.2.8 AUTOMATIZACIÓN**

La automatización de procesos documentales se ha consolidado como una estrategia efectiva para mejorar la eficiencia operativa y garantizar la consistencia en tareas de verificación formal, según los principios de aseguramiento de calidad establecidos en normas internacionales como ISO 9000:2015 [24], como:

- Abrir documentos
- Buscar tablas o gráficas
- Verificar secciones

### **Beneficios de la automatización**

- **Velocidad:** Un proceso manual de validación preliminar puede tomar 15-30 minutos por documento; automatizado: segundos.
- **Consistencia:** Cada documento se evalúa exactamente de la misma manera, sin variabilidad humana por fatiga, apuro, o interpretación subjetiva.
- **Escalabilidad:** El sistema puede procesar decenas o cientos de documentos sin incremento proporcional de recursos humanos.
- **Disponibilidad:** El sistema puede operar 24/7 sin restricciones de horario laboral.
- **Liberación de recursos humanos:** Ingenieros especializados pueden enfocarse en revisión técnica profunda (análisis de KPIs, evaluación de recomendaciones) en lugar de verificación formal estructural.
- **Reducción de errores:** La automatización elimina errores humanos típicos como:
  - Olvidar verificar algún criterio
  - Confundir documentos o versiones
  - Registrar incorrectamente resultados
  - Aplicar criterios inconsistentemente entre revisores

### **Alcance de automatización en el proyecto**

El proyecto automatiza específicamente la fase de validación preliminar de reportes técnicos:

Automatizado:

- Lectura y parsing del documento .docx
- Extracción de estructura (tablas, imágenes, secciones)
- Conteo de elementos
- Búsqueda de términos clave
- Verificación contra criterios predefinidos
- Generación de reporte de validación

NO automatizado (requiere revisión humana experta):

- Evaluación de si los valores de KPIs son técnicamente aceptables

- Valoración de la calidad de las recomendaciones
- Verificación de coherencia técnica del análisis
- Validación de la apropiación de la metodología
- Evaluación de la calidad de las conclusiones
- Aprobación final del reporte

### **Métricas de éxito de automatización**

Para evaluar el valor de la automatización implementada, se pueden establecer métricas como:

- **Tiempo de validación:**
  - Antes: X minutos por documento (manual)
  - Después: Y segundos por documento (automatizado)
  - Reducción: Z%
- **A lo largo:**
  - Documentos procesables por día/semana
  - Incremento en capacidad de procesamiento
- **Calidad:**
  - Reducción en documentos incompletos que avanzan a revisión técnica
  - Reducción en iteraciones por problemas estructurales
  - Incremento en consistencia de criterios aplicados
- **Satisfacción de usuarios:**
  - Feedback de revisores técnicos
  - Tiempo de ciclo total de revisión
- **Limitaciones y consideraciones**
  - *No reemplaza revisión técnica:* La automatización aborda validación estructural/formal, no análisis técnico profundo que requiere expertise humano.
  - *Requiere mantenimiento:* Criterios deben actualizarse, casos especiales manejarse, bugs corregirse. El sistema no es "configura y olvida".

- *Curva de aprendizaje:* Usuarios deben capacitarse en uso del sistema, interpretación de resultados, y procedimientos de escalamiento cuando encuentren problemas.
- *Dependencia tecnológica:* Se introduce dependencia en el software de validación; si falla, el proceso se interrumpe. Deben existir procedimientos de contingencia.

A pesar de estas consideraciones, los beneficios de automatización en validación documental de reportes técnicos superan ampliamente las limitaciones, justificando plenamente el desarrollo e implementación del sistema propuesto en este proyecto.



Figura 3: Relación entre Marco Teórico y Marco Conceptual

## 2.3 ESTADO DEL ARTE

El análisis automatizado de documentos técnicos ha cobrado gran relevancia en los últimos años, impulsado por el crecimiento de herramientas capaces de procesar archivos estructurados y extraer información relevante sin intervención manual. En el caso de los

reportes técnicos del área de Radiofrecuencia, esta tendencia se vuelve especialmente importante, ya que dichos documentos suelen generarse en formato Word y contienen múltiples tablas, gráficas y parámetros que requieren verificación constante.

Uno de los conceptos fundamentales que sustenta este proyecto es la estructura interna de los archivos .docx, los cuales son, en realidad, contenedores comprimidos (ZIP) que almacenan distintos archivos XML, entre ellos document.xml y styles.xml. La posibilidad de extraer y analizar (parsear) estos XML permite localizar secciones, tablas e imágenes incrustadas sin necesidad de recurrir a métodos basados en reconocimiento óptico de caracteres (OCR) [1]. Este principio constituye la base técnica para el software propuesto, el cual accederá directamente a la estructura interna de los reportes para verificar su completitud.

A este proceso se suma la utilización de Pandoc, una herramienta de conversión de código abierto que permite transformar archivos Word (.docx) en HTML u otros formatos intermedios, lo que facilita la lectura estructurada del contenido, incluyendo cabeceras, tablas e imágenes [2][3]. En este proyecto, Pandoc se aplicará como puente entre la lectura XML y la detección de los elementos requeridos, integrándose dentro del flujo de validación del software desarrollado en Godot.

Desde el punto de vista técnico en telecomunicaciones, conceptos como MOCN y MORAN son esenciales, ya que se refieren a la compartición de infraestructura y a la integración de redes LTE y UMTS entre distintos operadores. En estos procesos, los reportes de campo deben incluir parámetros específicos, como bandas de frecuencia, potencias o configuraciones de red, que son verificados en las etapas de aseguramiento de calidad (QA) [4]. De ahí la importancia de un sistema que garantice que dichos reportes estén completos antes de su análisis técnico más profundo.

A nivel internacional, destacan diversos estudios sobre la comprensión automática de documentos (Document Understanding), entre ellos los modelos LayoutLMv2 y LayoutLMv3, desarrollados por Microsoft en 2021 y 2022 respectivamente [5][6]. Estos modelos de inteligencia artificial utilizan información textual, visual y de formato para comprender documentos complejos, demostrando que los enfoques layout-aware mejoran la identificación de secciones, tablas y gráficos. Si bien su implementación requiere capacidades avanzadas de IA y entrenamiento masivo de datos, su filosofía metodológica

respalda el principio del software propuesto: reconocer la estructura documental y detectar automáticamente los elementos esenciales.

Otros trabajos relevantes, como READoc [7], consolidan la evaluación de pipelines reales para la extracción estructurada de información (Document Structured Extraction, DSE), resaltando las limitaciones de las herramientas actuales frente a documentos extensos y heterogéneos. Estos estudios sustentan la elección de un enfoque por etapas en el proyecto, basado en la conversión, el análisis y la aplicación de reglas específicas.

En el sector de las telecomunicaciones, documentos técnicos como el MORAN Testing for NaaS Environments White Paper (2021) [8] y el Ericsson Mobility Report (2025) [9] destacan la tendencia de la industria hacia la automatización de procesos de prueba, validación y análisis de red. Estas publicaciones enfatizan la incorporación de automatización e inteligencia artificial en la gestión de datos técnicos, lo que justifica la necesidad de desarrollar software que optimice la validación de reportes en proyectos de despliegue y compartición de infraestructura.

A nivel nacional, Colombia ha avanzado significativamente en la automatización de procesos documentales, especialmente mediante RPA (Automatización Robótica de Procesos). La validación del prototipo con usuarios directamente involucrados en el proceso operativo coincide con prácticas reportadas en proyectos de automatización similares, donde la retroalimentación temprana permite ajustar funcionalidades y asegurar la aplicabilidad real de la solución [10], demostrando cómo la automatización de reglas reduce errores humanos y mejora la eficiencia. De forma similar, proyectos universitarios recientes sobre automatización de reportes empresariales [11] y el diseño de macros para procesamiento de datos en la Universidad Cooperativa de Colombia [12] evidencian el potencial de los scripts y programas específicos para mejorar tareas repetitivas. Estos antecedentes aportan metodologías y enfoques que son directamente aplicables en este proyecto, particularmente en la lectura estructurada de documentos y la generación de validaciones automáticas.

En el plano normativo, el Ministerio TIC de Colombia y el Archivo General de la Nación han publicado lineamientos entre 2021 y 2022 [13] que promueven la automatización de la gestión documental y la trazabilidad de la información. Dichas iniciativas refuerzan la relevancia de proyectos que impulsen la digitalización y la estandarización de flujos de validación dentro de entornos empresariales.

Finalmente, estudios nacionales más recientes sobre el impacto de la Automatización Robótica de Procesos (RPA) como se reporta en [14] demuestran que este tipo de soluciones son eficaces para tareas repetitivas en distintos sectores productivos, validando la viabilidad y el impacto de la automatización en procesos que dependen fuertemente de documentos técnicos.

Estos antecedentes muestran que, aunque existen muchas herramientas para el análisis estructural de documentos y la automatización de tareas, no se ha encontrado una aplicación enfocada en la validación de reportes técnicos de pruebas de campo en telecomunicaciones. Por tanto, el software propuesto representa una aplicación novedosa y práctica que combina tecnologías abiertas como Godot y Pandoc para procesar documentos Word (.docx), identificar elementos faltantes y mejorar la precisión y consistencia en la revisión de reportes técnicos dentro del área de Radiofrecuencia de Huawei-Colombia.

A diferencia de los enfoques basados en modelos avanzados de Document Understanding, como LayoutLMv2 y LayoutLMv3, que integran análisis visual y semántico mediante técnicas de inteligencia artificial, la solución propuesta en este trabajo adopta un enfoque ligero, determinista y orientado a reglas, adecuado para su implementación inmediata en entornos empresariales sin requerimientos de entrenamiento de modelos ni grandes volúmenes de datos.

Mientras que muchas de las herramientas reportadas en la literatura se centran en la extracción semántica o clasificación de documentos, ScanDoc se enfoca específicamente en la validación estructural de reportes técnicos de pruebas de campo, un problema operativo concreto que no ha sido abordado directamente por soluciones existentes. Asimismo, a diferencia de herramientas genéricas de análisis documental, el software desarrollado se adapta a las plantillas y convenciones internas del área de Radiofrecuencia de Huawei-Colombia, lo que incrementa su aplicabilidad práctica.

En este sentido, la principal contribución del proyecto no radica en la incorporación de algoritmos complejos de inteligencia artificial, sino en la integración efectiva de herramientas abiertas (Pandoc y Godot) dentro de un flujo automatizado que resuelve una

necesidad real del entorno laboral, logrando mejoras significativas en eficiencia, estandarización y control de calidad documental.

### 3 ANÁLISIS

Este capítulo consolida los requerimientos del cliente y los requisitos del software que orientan el diseño e implementación del sistema de validación de reportes técnicos. La información se definió a partir del diagnóstico del proceso manual, la revisión de plantillas de reportes en formato .docx utilizados por el área de Radiofrecuencia y la retroalimentación de los actores involucrados. Para efectos del presente documento, se entiende por **requerimientos** las necesidades/expectativas del cliente, y por **requisitos** las características y condiciones del software que dan respuesta a dichos requerimientos.

#### 3.1 REQUERIMIENTOS DEL CLIENTE

- I) Automatizar la validación preliminar de reportes técnicos de pruebas de campo (LTE y UMTS) que actualmente se revisan de forma manual.
- II) Reducir el tiempo total de validación preliminar y habilitar el procesamiento masivo de reportes en escenarios de alta carga (p. ej., cientos de documentos en un día).
- III) Detectar tempranamente reportes incompletos mediante la identificación de ausencia de secciones, tablas, gráficas o parámetros esenciales, para solicitar correcciones inmediatas.
- IV) Estandarizar la validación preliminar aplicando criterios uniformes, minimizando la variabilidad derivada del revisor (fatiga, interpretación o experiencia).
- V) Trabajar con reportes en formato Microsoft Word (.docx), de acuerdo con el formato de entrega utilizado por el área.
- VI) Preservar la integridad del reporte fuente: el sistema no debe modificar el documento original, y debe generar la salida en un archivo exportado independiente con comentarios u observaciones.
- VII) Entregar retroalimentación clara y accionable sobre los faltantes detectados, facilitando la corrección por parte de los responsables.

## 3.2 REQUISITOS DEL SOFTWARE

Los requisitos del software definen de manera clara y estructurada las funcionalidades, características y restricciones que debe cumplir un sistema, sirviendo como base para su diseño y desarrollo. Incluyen tanto requisitos funcionales, que describen las acciones que el sistema debe realizar, como no funcionales, enfocados en aspectos de calidad como rendimiento, seguridad y usabilidad. Su correcta definición permite asegurar que el software satisfaga las necesidades de los usuarios y cumpla con los objetivos establecidos.

### 3.2.1 REQUISITOS FUNCIONALES (RF)

ID	Descripción	Prioridad
RF01	El sistema debe permitir cargar uno o varios archivos en formato .docx para su validación.	Alta
RF02	El sistema debe validar las secciones obligatorias de las plantillas de reportes de pruebas de campo (LTE y UMTS), analizando su estructura para identificar tablas e imágenes requeridas.	Alta
RF03	El sistema debe realizar la conversión del documento a un formato intermedio que facilite el análisis estructural (p. ej., HTML a partir de .docx mediante Pandoc).	Alta
RF04	El sistema debe aplicar criterios de validación de forma automática y uniforme para todos los documentos procesados.	Media
RF05	El sistema debe generar un resultado por documento que indique el cumplimiento o incumplimiento, junto con el detalle de elementos faltantes.	Alta
RF06	El sistema debe generar un archivo exportado independiente que incluya comentarios u observaciones de validación, sin modificar el archivo original.	Alta
RF07	El sistema debe presentar los resultados de validación de forma clara en la interfaz, incluyendo indicadores de estado por sitio/documento cuando aplique.	Media

RF08	El sistema debe permitir exportar el resultado de validación de manera individual, por sitio o de forma masiva (según el flujo descrito en el capítulo de desarrollo).	Media
------	--	-------

*Tabla 3: Requisitos Funcionales*

### 3.2.2 REQUISITOS NO FUNCIONALES (RNF)

ID	Descripción	Prioridad
RNF01	El sistema debe ejecutar la validación preliminar en tiempos significativamente menores al proceso manual (orden de segundos por documento, según condiciones del equipo y tamaño del archivo).	Alta
RNF02	El sistema debe garantizar consistencia en la aplicación de criterios: un mismo documento debe producir el mismo resultado bajo las mismas reglas.	Alta
RNF03	El sistema no debe alterar ni sobrescribir los archivos originales cargados por el usuario.	Alta
RNF04	El sistema debe soportar el procesamiento de volúmenes altos de documentos (p. ej., más de 200 archivos) sin requerir intervención adicional durante la ejecución.	Media
RNF05	La solución debe ejecutarse como aplicación de escritorio en un entorno de trabajo típico (Windows), acorde con el uso de herramientas del área.	Media
RNF06	La arquitectura debe facilitar la mantenibilidad: permitir agregar o ajustar criterios de validación sin rediseño completo del sistema.	Baja

*Tabla 4: Requisitos No Funcionales*

### 3.3 RESTRICCIONES Y SUPUESTOS

- El sistema se limita a la validación preliminar estructural del reporte (completitud y presencia de componentes), y no reemplaza la revisión técnica experta de resultados de RF.

- El alcance contempla reportes de tecnologías LTE y UMTS para los operadores indicados en el capítulo de marco general.
- Los criterios de validación dependen de las plantillas y convenciones internas; cambios en las plantillas pueden requerir ajuste de reglas.
- La conversión y análisis dependen de la correcta lectura del archivo .docx y del proceso de conversión a formato intermedio; documentos corruptos o con estructuras atípicas pueden generar resultados no concluyentes.
- La incorporación de inteligencia artificial para análisis profundo se considera una posibilidad futura, pero no hace parte del alcance inicial.

### **3.4 CRITERIOS GENERALES DE ACEPTACIÓN**

- I) El sistema debe permitir cargar y procesar uno o varios archivos .docx sin modificar el archivo fuente.
- II) Para cada documento procesado, el sistema debe indicar: (a) estado de cumplimiento, y (b) lista de elementos faltantes cuando aplique.
- III) El sistema debe generar un archivo exportado independiente con comentarios/observaciones de validación coherentes con el resultado mostrado en la interfaz.
- IV) El sistema debe poder ejecutar validaciones masivas con resultados consistentes, manteniendo trazabilidad por documento.

## **4 DESARROLLO DE LA SOLUCIÓN**

En este capítulo se describe de manera detallada la implementación técnica y funcional de la propuesta planteada, explicando cómo se materializa el diseño definido previamente. En este apartado se abordan la arquitectura del sistema, las herramientas y tecnologías utilizadas, el flujo de procesamiento de los documentos Word y los mecanismos que permiten la carga, gestión y análisis simultáneo de múltiples archivos, evidenciando cómo la solución satisface los requerimientos identificados y cumple los objetivos del proyecto.

La selección de Godot Engine como plataforma de desarrollo respondió a la necesidad de contar con una herramienta multiplataforma, de código abierto y con capacidades robustas para la construcción de interfaces gráficas, sin incurrir en costos de licencia. Su arquitectura basada en nodos permitió estructurar el sistema de forma modular y facilitar futuras ampliaciones. No obstante, su uso implicó una curva de aprendizaje inicial, al tratarse de un motor tradicionalmente orientado al desarrollo de videojuegos.

Por su parte, Pandoc fue seleccionado por su capacidad de conversión confiable entre formatos documentales y por su amplia adopción en contextos académicos y técnicos. Su principal ventaja radica en la preservación de la estructura semántica del documento durante la conversión; sin embargo, pueden presentarse limitaciones en tablas o gráficos altamente complejos, lo cual fue mitigado mediante reglas de validación tolerantes.

#### **4.1 MÉTODO DE DESARROLLO**

Para el desarrollo de ScanDoc se seleccionó como modelo de desarrollo el Proceso Unificado Rational (RUP, Rational Unified Process), debido a que es una metodología iterativa e incremental que permite construir el producto en versiones sucesivas, incorporando retroalimentación continua y controlando los riesgos desde etapas tempranas. Esta característica resulta adecuada para el presente proyecto, dado que la solución integra varios componentes (interfaz gráfica, conversión de documentos, análisis estructural y exportación de resultados) que debían validarse progresivamente con reportes reales y con los actores del proceso (supervisor y usuarios de prueba).

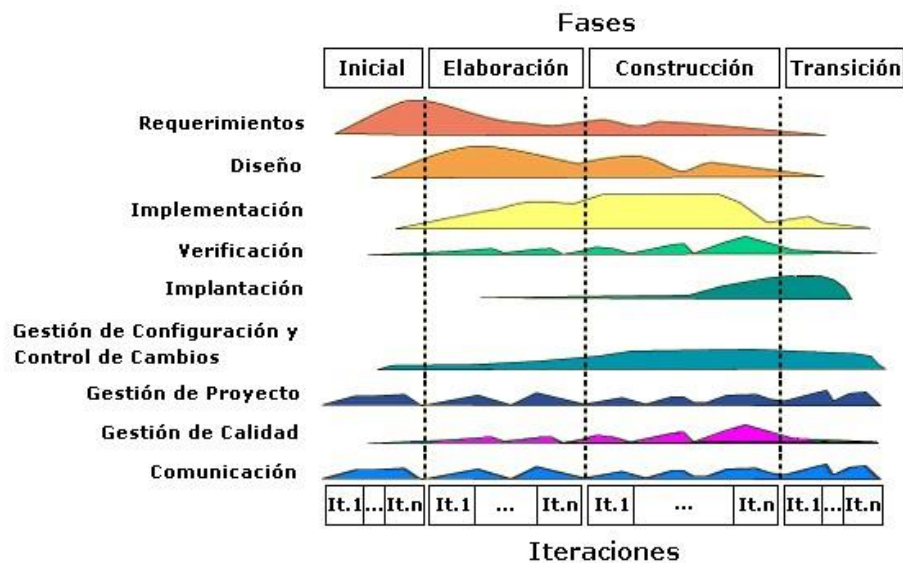


Figura 4: Ciclo de vida RUP

Tomado de:

<https://www.fing.edu.uy/inco/cursos/ingsoft/pis/memoria/dvd02/experiencia2017/MUM/index.htm>

Adicionalmente, RUP favorece la definición temprana de la arquitectura y la trazabilidad entre requerimientos, diseño e implementación, aspectos clave para asegurar que las reglas de validación y el flujo de procesamiento (lectura de .docx, conversión a HTML, verificación de criterios y generación del archivo exportado) respondieran a las necesidades operativas identificadas. En este sentido, el enfoque iterativo permitió:

- I) Identificar y priorizar funcionalidades esenciales.
- II) Implementar prototipos funcionales por módulos.
- III) Ejecutar ciclos repetidos de prueba–ajuste hasta alcanzar un comportamiento estable y coherente con los criterios de aceptación del proyecto.

#### 4.1.1 ETAPAS DE LA METODOLOGÍA RUP EN EL PROYECTO

RUP se estructura en cuatro fases (Inicio, Elaboración, Construcción y Transición). Para este proyecto, dichas fases se adaptaron al contexto de desarrollo de un prototipo funcional de validación documental, manteniendo el principio de iterar y refinar de forma gradual. A continuación, se describen las actividades principales realizadas en cada etapa.

**Inicio (Inception):** Se identificó la problemática del proceso manual de validación de reportes, se definieron objetivos y alcance, y se recolectaron insumos iniciales (plantillas de reportes LTE/UMTS, criterios mínimos de completitud y expectativas del supervisor). Como resultado, se consolidaron los requerimientos del cliente y los criterios generales de aceptación que guían el desarrollo.

**Elaboración (Elaboration):** Se diseñó la arquitectura general del sistema y el flujo de procesamiento (entrada de documentos, extracción/transformación a HTML, aplicación de reglas y presentación/exportación de resultados). En esta fase también se definió el modelo de datos mínimo para representar documentos, secciones y resultados, y se especificaron los criterios de validación que serían implementados.

**Construcción (Construction):** Se implementaron los módulos del sistema en iteraciones, integrando la interfaz en Godot con la conversión mediante Pandoc y las rutinas de análisis estructural. Cada incremento incorporó ajustes a la lógica de validación, mejoras de usabilidad y manejo de casos de error, con base en pruebas repetidas sobre documentos reales.

**Transición (Transition):** Se ejecutaron pruebas con sujetos de prueba y con el supervisor (incluyendo el escenario de carga masiva), se compararon métricas frente a la validación manual, y se realizaron correcciones finales orientadas a estabilidad y claridad de resultados. Finalmente, se consolidó la documentación y los resultados para la entrega del producto y su evaluación.

## 4.2 DISEÑO DE LA SOLUCIÓN

El diseño de la solución propuesta se orienta al desarrollo de un software automatizado capaz de validar la completitud estructural de reportes técnicos generados a partir de pruebas de campo en redes LTE y UMTS. Este diseño tiene como finalidad establecer una arquitectura clara y modular que permita procesar documentos en formato Word (.docx), analizar su estructura interna y verificar la presencia de los elementos técnicos obligatorios definidos por el área de Radiofrecuencia de Huawei-Colombia.

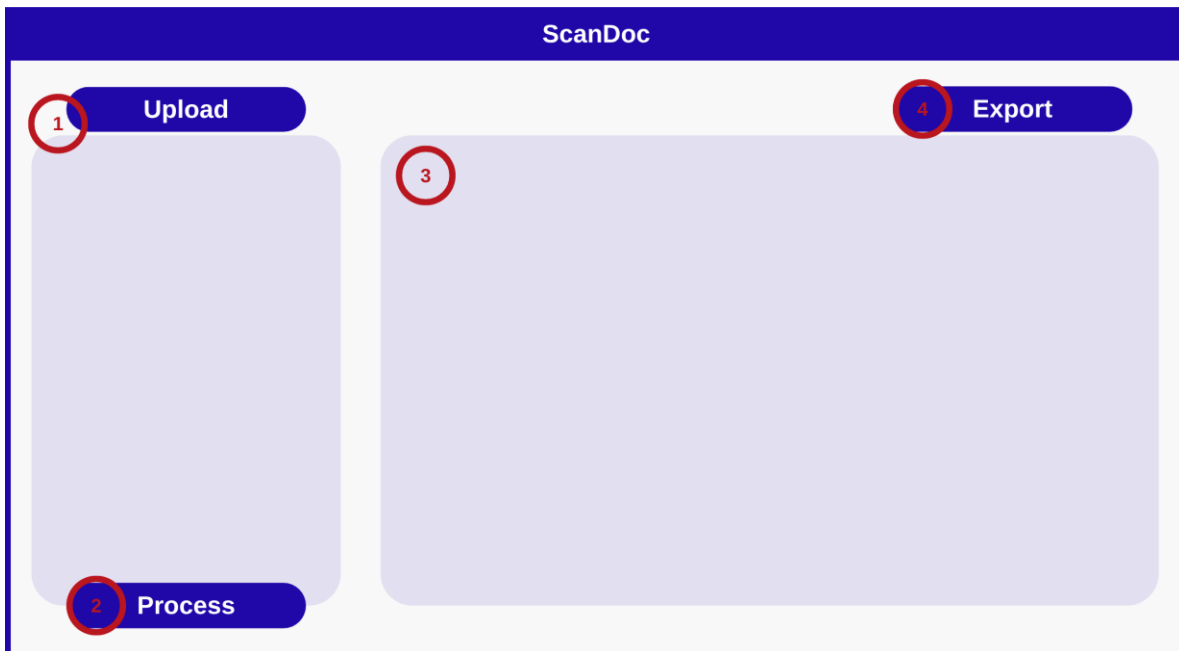


Figura 5: Prototipo del software ScanDoc 1

La solución se concibe bajo un enfoque modular, en el cual cada componente cumple una función específica dentro del flujo de validación. En primer lugar, el sistema debe ser capaz de recibir uno o varios reportes técnicos en formato .docx, los cuales contienen información crítica como tablas de KPIs, gráficas de potencia, mapas de cobertura y secciones de análisis. Posteriormente, estos documentos son procesados internamente mediante la extracción de su estructura XML y su conversión a un formato intermedio que facilite el análisis automático.

#### 4.2.1 ARQUITECTURA GENERAL DEL SISTEMA

La arquitectura general del sistema se diseñó bajo un enfoque modular, con el fin de facilitar el procesamiento estructurado de reportes técnicos y permitir una implementación clara y mantenible. El sistema se compone de cuatro módulos principales: el módulo de entrada de documentos, el módulo de procesamiento, el módulo de presentación de resultados y el módulo de exportación.

El módulo de entrada se encarga de recibir uno o varios archivos en formato Word (.docx) seleccionados por el usuario a través de la interfaz gráfica. Estos archivos corresponden a

reportes técnicos generados durante las pruebas de campo del área de Radiofrecuencia. El sistema **NO modifica el documento original**; preserva el archivo fuente y, a partir del análisis, **genera un archivo exportado aparte** que incluye comentarios de validación, observaciones y datos relevantes, con el fin de dejar trazabilidad directa dentro del reporte entregable.

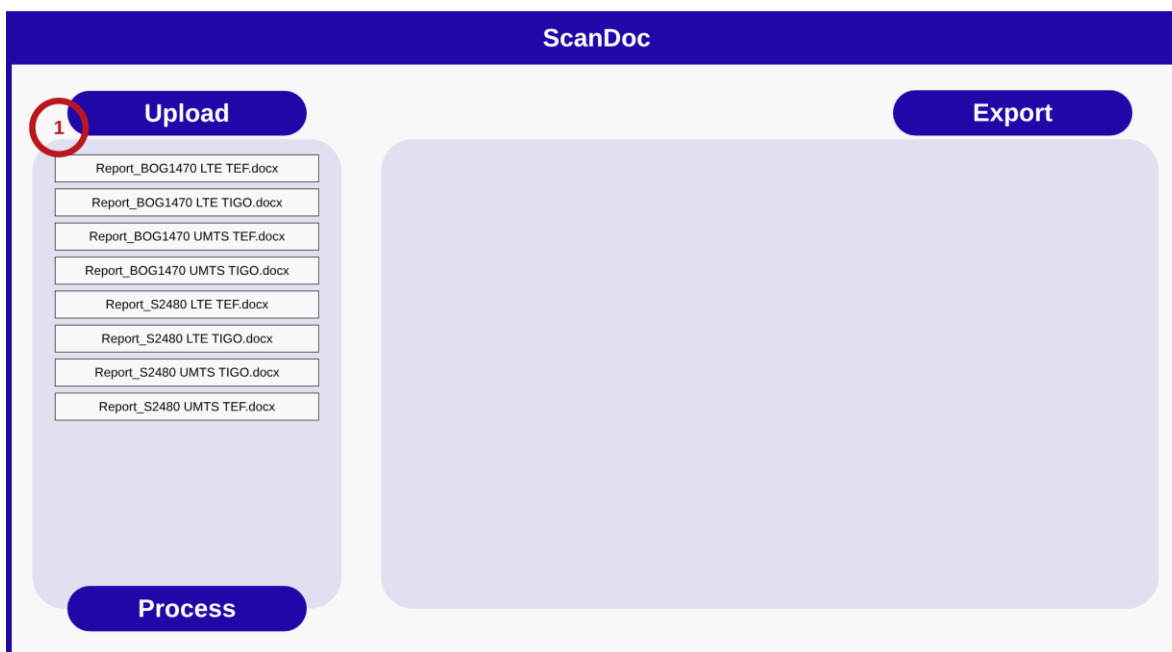


Figura 6: Prototipo del software ScanDoc 2

El módulo de procesamiento constituye el núcleo del sistema. En esta etapa se realiza la extracción de la estructura interna del archivo .docx, aprovechando que este formato se basa en archivos XML comprimidos. Para facilitar el análisis estructural, se integra la herramienta Pandoc, la cual convierte el contenido del documento a un formato HTML que preserva la jerarquía de títulos, tablas e imágenes.

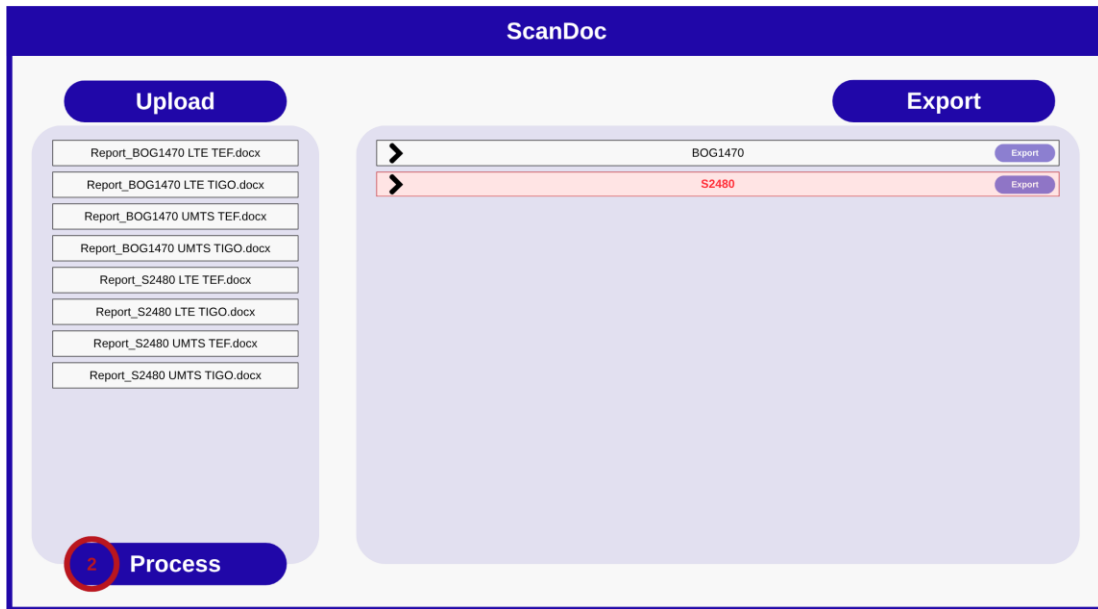


Figura 7: Prototipo del software ScanDoc 3

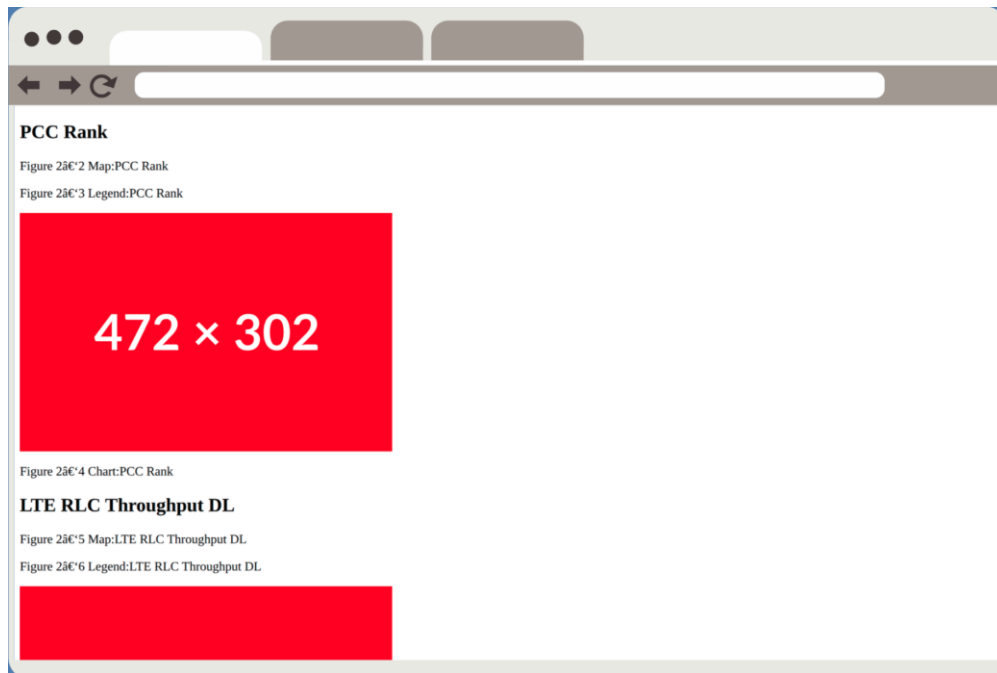


Figura 8: Vista preliminar del documento HTML (sin gráfcos ni elementos finales)

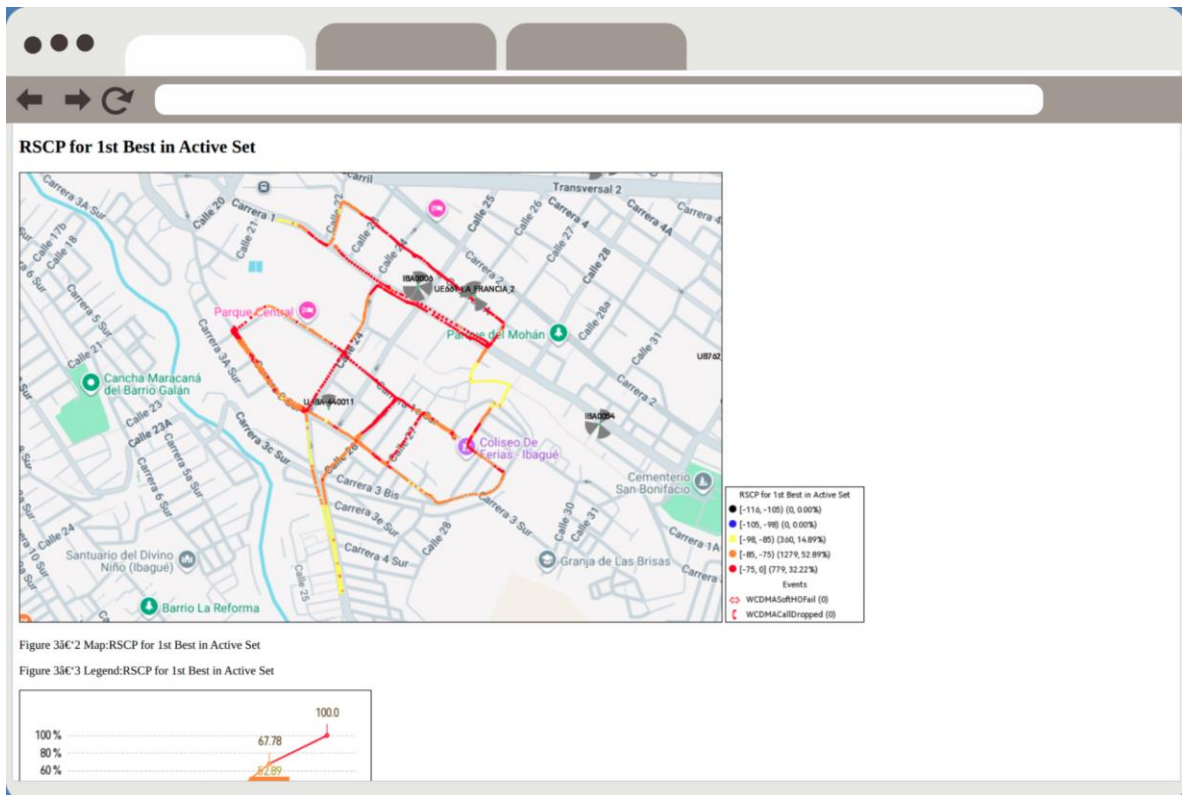


Figura 9: Vista preliminar del documento HTML

Posteriormente, el módulo de presentación aplica un conjunto de reglas predefinidas que permiten verificar la presencia de elementos obligatorios dentro del reporte, tales como secciones, tablas de KPIs y gráficas técnicas, mostrando los resultados de la validación a través de la interfaz desarrollada en Godot, indicando de forma clara si el reporte cumple o no con los criterios establecidos y señalando los elementos faltantes cuando aplique. Estas reglas se ejecutan de forma automática y uniforme para todos los documentos procesados.

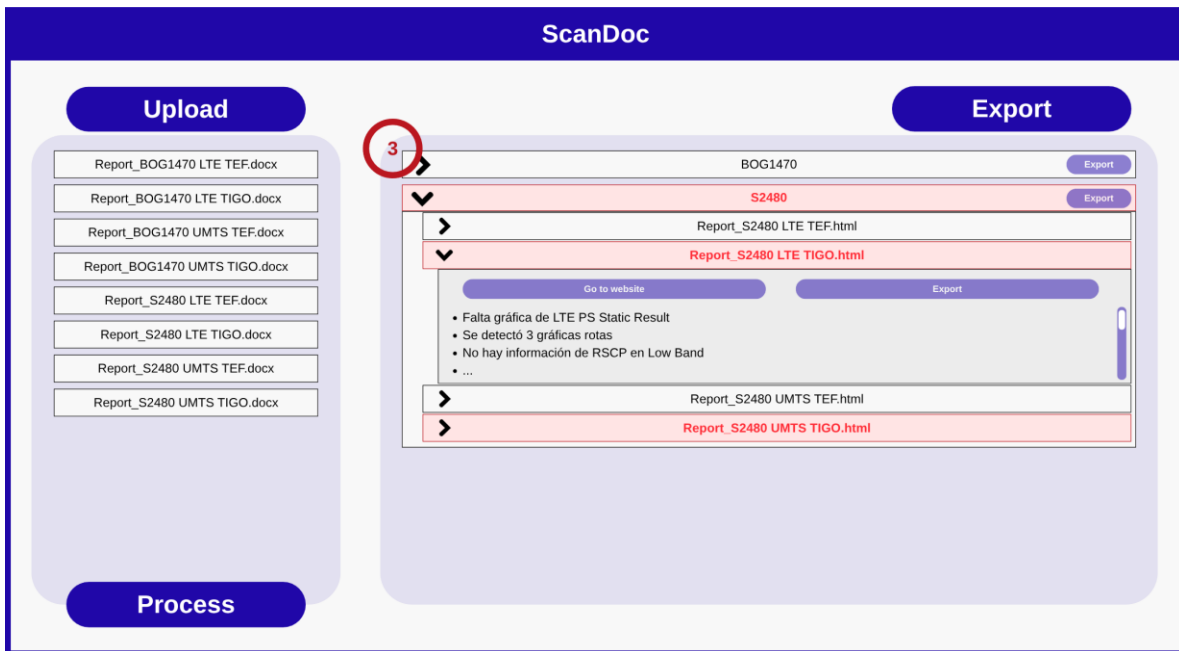


Figura 10: Prototipo del software ScanDoc 4

Finalmente, el módulo de exportación se encarga de generar el documento final a partir del reporte original, incorporando de forma automática los elementos definidos por el sistema (por ejemplo, comentarios, notas de verificación y/o secciones resumen). Este proceso se realiza sin alterar ni sobrescribir el archivo fuente, garantizando la integridad del reporte original y produciendo una versión exportada independiente lista para revisión y entrega. Adicionalmente, el módulo permite realizar la exportación de manera individual, por sitio o de forma masiva, de acuerdo con el flujo de trabajo establecido para el procesamiento de múltiples documentos.



El flujo de procesamiento de documentos inicia con la carga del archivo Word (.docx) por parte del usuario. Una vez seleccionado el archivo, el sistema accede a su estructura interna, la cual se encuentra compuesta por múltiples archivos XML que describen el contenido, el formato y la organización del documento.

En una primera etapa, el sistema identifica los archivos XML relevantes, principalmente aquellos que contienen el cuerpo del documento y sus estilos. A continuación, se utiliza Pandoc como herramienta de conversión para transformar el contenido del documento a formato HTML, conservando la estructura lógica de encabezados, tablas e imágenes.

El HTML generado actúa como una representación intermedia que facilita el análisis automático, ya que permite identificar etiquetas y nodos asociados a secciones específicas del reporte. Sobre esta estructura, el sistema ejecuta procesos de lectura y análisis que buscan detectar la presencia de los elementos definidos como obligatorios dentro del reporte técnico.

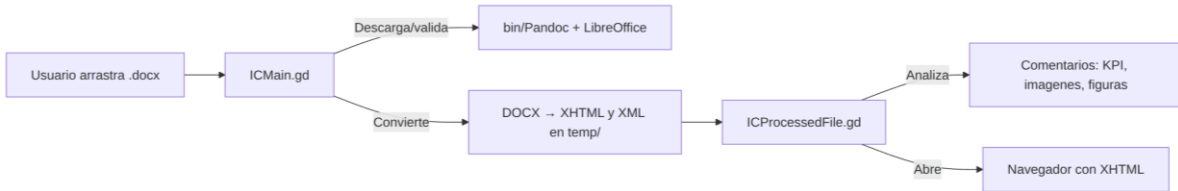


Figura 13: Flujo general del software

Este flujo garantiza un procesamiento consistente y controlado de los documentos, evitando dependencias de métodos basados en reconocimiento visual u OCR, y asegurando que el análisis se realice directamente sobre la estructura interna del archivo. Como resultado, el sistema mantiene intacto el documento original y produce un documento final exportado en el que se agregan comentarios de validación y datos relevantes del análisis para dejar trazabilidad de los hallazgos.

### 4.2.3 MODELO DE DATOS

El modelo de datos del sistema se diseñó para representar de manera estructurada la información mínima necesaria para realizar la validación de los reportes técnicos. Este

modelo no almacena el contenido completo de los documentos, sino que gestiona metadatos y resultados derivados del proceso de validación.

El modelo contempla entidades como Documento, Sección, Elemento validable y Resultado de validación. La entidad Documento representa el archivo procesado e incluye atributos básicos como nombre del archivo y fecha de procesamiento. La entidad Sección corresponde a los bloques principales del reporte, mientras que los Elementos validables representan componentes específicos como tablas, gráficas o mapas.

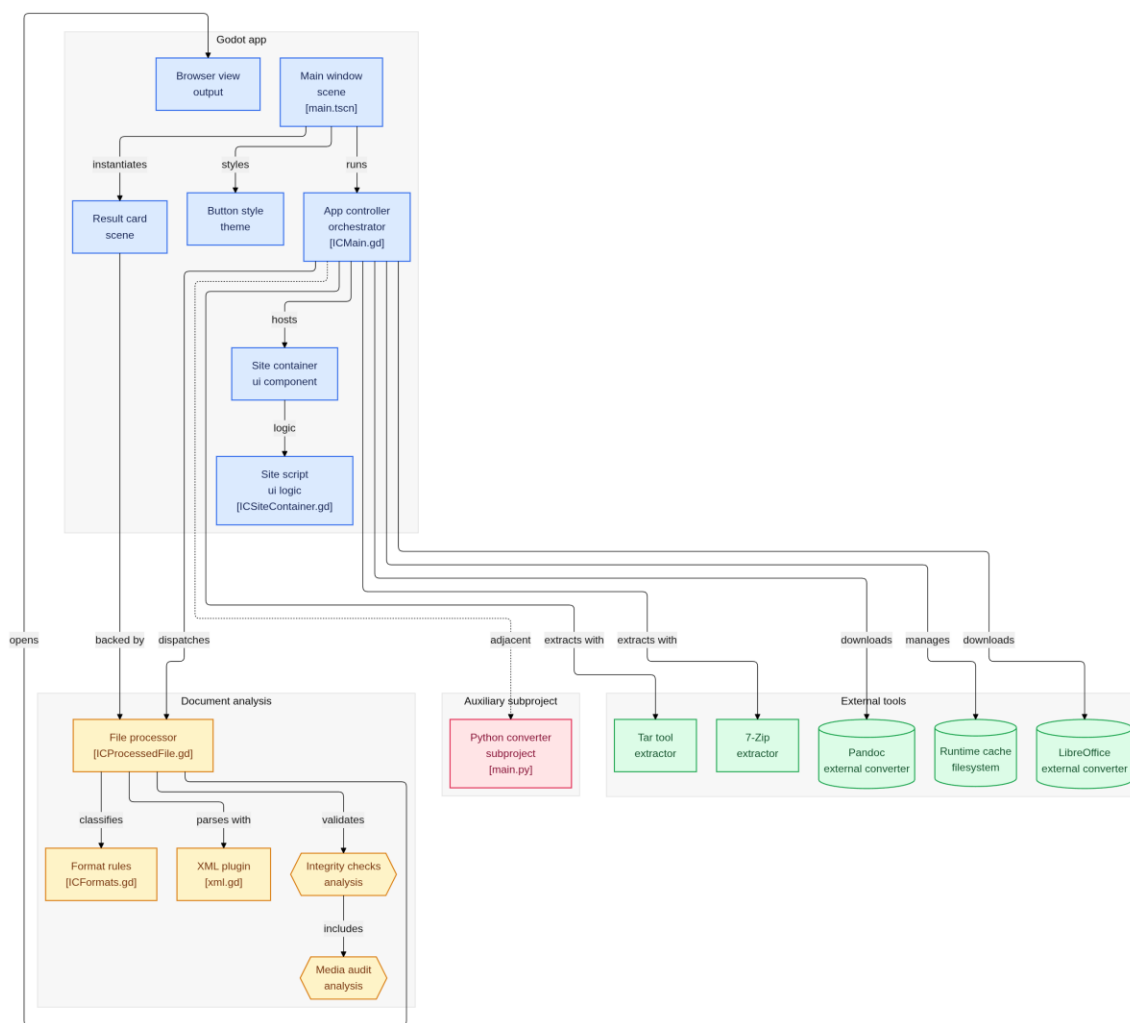


Figura 14: Diagrama de datos relación entidades

Finalmente, la entidad Resultado de validación asocia cada documento con el estado de cumplimiento de los criterios definidos, permitiendo identificar de forma clara qué elementos están presentes y cuáles faltan. Este modelo de datos sirve como base para la visualización de resultados dentro de la interfaz gráfica.

#### 4.2.4 CRITERIOS DE VALIDACIÓN

Los criterios de validación definen las reglas mediante las cuales el sistema evalúa la completitud estructural de los reportes técnicos. Estos criterios se establecieron con base en las plantillas utilizadas por el área de Radiofrecuencia y en los requerimientos mínimos exigidos para la aceptación de pruebas de campo en redes LTE y UMTS.

```
if started:
# count img_titles an images
img_count += line.count("<img")
table_count += line.to_lower().count("<table")

if line.contains("Serving PCI"):
if not line.contains("Chart:"):
figure_count += line.count(" Figure ")
else:
figure_count += line.count(" Figure ")
```

Entre los criterios definidos se encuentran la verificación de la existencia de secciones obligatorias, la presencia de tablas de KPIs, la inclusión de gráficas técnicas asociadas a parámetros de red y la correcta organización general del documento. Cada criterio se evalúa de manera independiente, permitiendo identificar de forma precisa las omisiones dentro del reporte.

```
if row_title == CLUSTER_SINR_AVERAGE:
if row_value.is_empty():
commentaries.append("No information was made or found of SINR in %s." % last_band)
continue

var value: float = row_value.to_float()
var label: String = ""

if value < 0:
label = "low levels"
elif value >= 0 and value < 10:
label = "normal levels"
else:
label = "good levels"
```

```
if remaining_low_cluster_avg > 0:  
    remaining_low_cluster_avg -= 1  
    cluster_values_found += 1  
  
if remaining_high_cluster_avg > 0:  
    remaining_high_cluster_avg -= 1  
    cluster_values_found += 1  
  
commentaries.append("In %s the SINR average has %s" % [last_band, label])
```

## 5 IMPLEMENTACIÓN DEL SISTEMA

La implementación del sistema se realizó con base en el diseño de la arquitectura y el modelo de datos definidos en la sección anterior, siguiendo un enfoque iterativo que permitió validar cada componente antes de su integración total. El desarrollo se centró en la creación de un producto funcional capaz de procesar reportes técnicos en formato Word (.docx) y ejecutar validaciones estructurales de manera automatizada.



Figura 15: Ícono de ScanDoc

El entorno de desarrollo seleccionado fue Godot Engine, utilizando su lenguaje nativo GDScript, debido a su flexibilidad para la construcción de interfaces gráficas, su capacidad de integración con herramientas externas y su facilidad para estructurar aplicaciones modulares. Godot actúa como núcleo del sistema, gestionando la interacción con el usuario, el flujo de ejecución y la visualización de resultados.

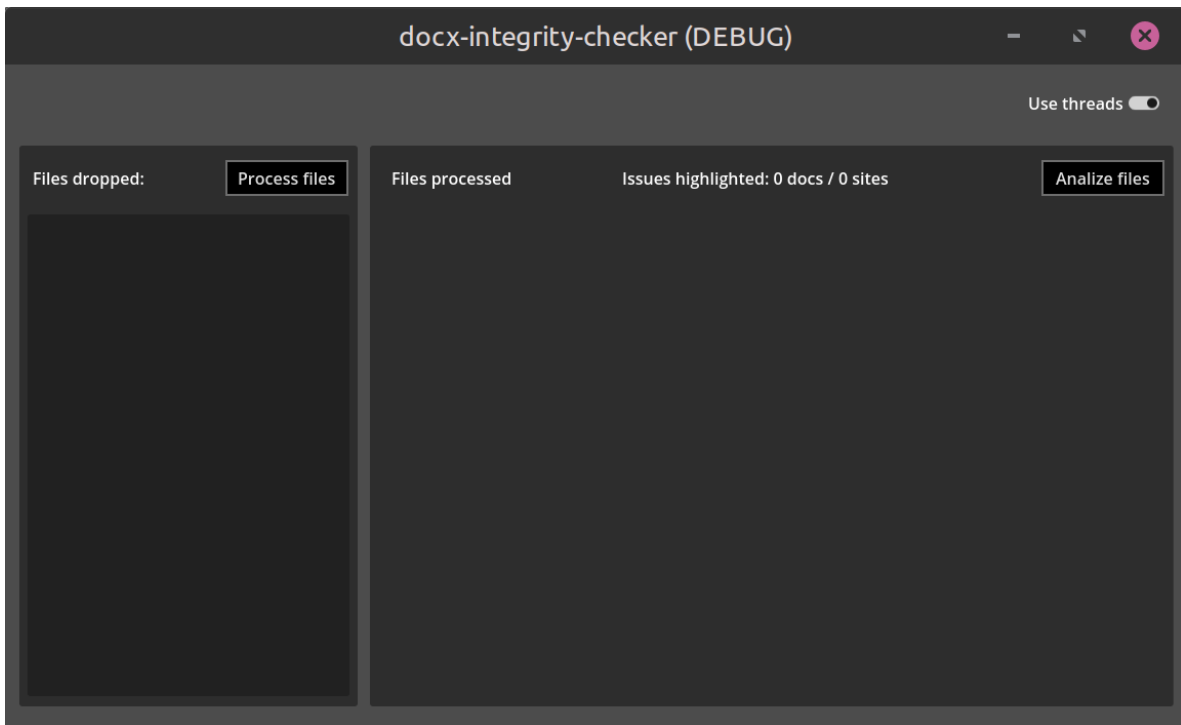


Figura 16: Interfaz de Scandoc inicial

Durante la implementación, se desarrolló un módulo de carga de documentos, el cual permite al usuario seleccionar uno o varios archivos Word desde el sistema de archivos. Una vez cargados, el sistema inicia el flujo de procesamiento definido en la arquitectura. Estos archivos se procesan internamente para ejecutar la validación estructural y, al finalizar, el sistema habilita la exportación de un documento final (archivo aparte) con comentarios y observaciones.

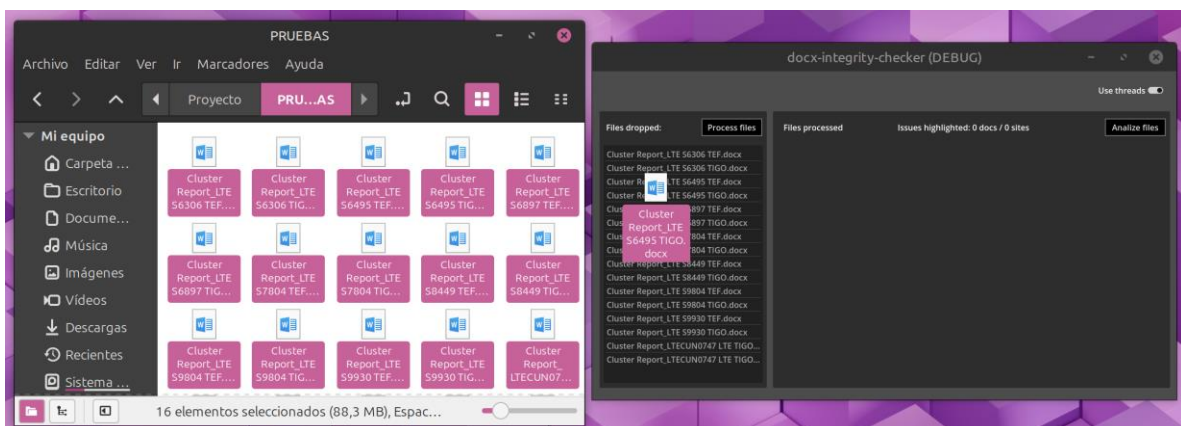


Figura 17: Arrastrando documentos a ScanDoc

Para la conversión de documentos, se integró la herramienta Pandoc, la cual es invocada desde Godot mediante scripts que controlan la ejecución del proceso de conversión. Pandoc transforma los archivos .docx en documentos HTML, preservando la estructura jerárquica de encabezados, tablas e imágenes. Esta conversión constituye una etapa clave, ya que el formato HTML facilita el análisis estructural posterior.

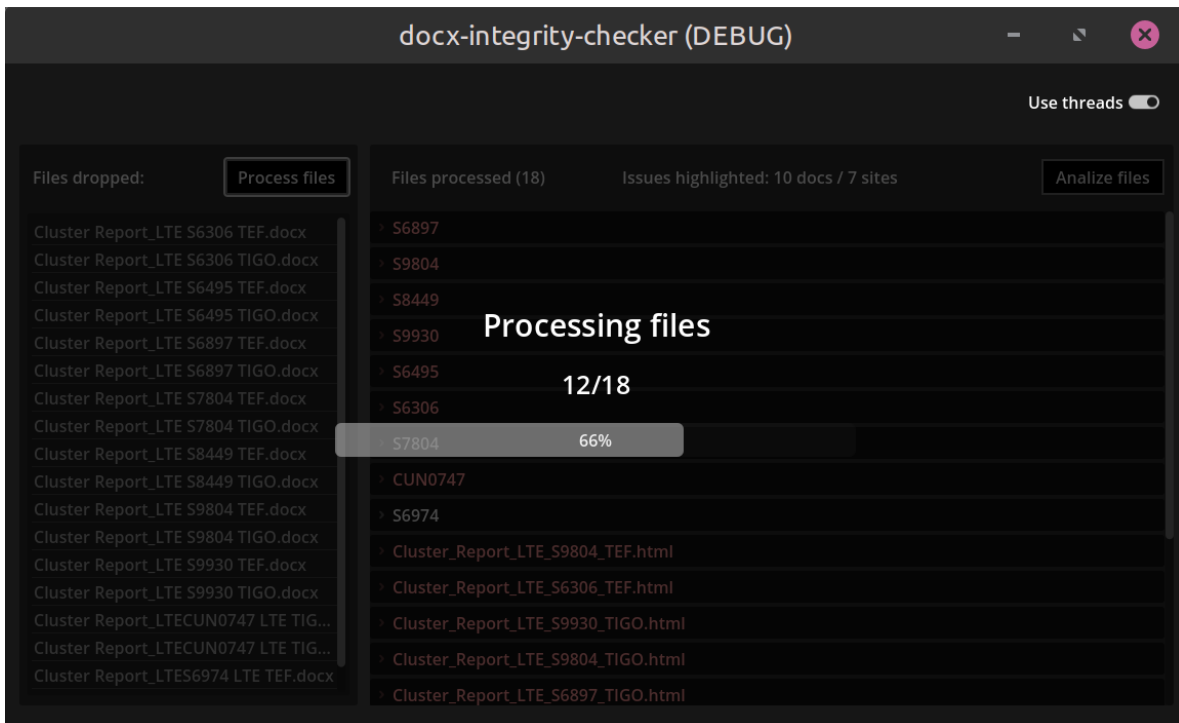


Figura 18: Interfaz de carga procesando N documentos

```
ScanDoc
❄️converting Cluster Report_LTE S6495 TEF.docx to xml
❄️converting Cluster Report_LTE S6897 TEF.docx to xml
❄️converting Cluster Report_LTES6974 LTE TEF.docx to xml
❄️converting Cluster Report_LTECUN0747 LTE TIG0 good.docx to xml
❄️converting Cluster Report_LTE S7804 TIG0.docx to xml
❄️converting Cluster Report_LTE S7804 TEF.docx to xml
❄️converting Cluster Report_LTE S8449 TEF.docx to xml
❄️converting Cluster Report_LTECUN0747 LTE TIG0.docx to xml
WARNING: Media count mismatch in section " LTE VoLTE Intermediate Call KPI Summary", figures
: 0, imgs: 0, tables: 1, file: Cluster_Report_LTE_S6897_TIG0.html
at: push_warning (core/variant/variant_utility.cpp:1034)
GDScript backtrace (most recent call first):
[0] summarize_media_counts_by_category (res://scripts/ICProcessedFile.gd:776)
[1] validate_integrity (res://scripts/ICProcessedFile.gd:77)
[2] process_file (res://scripts/ICMain.gd:308)
WARNING: Media count mismatch in section " LTE VoLTE Intermediate Call KPI Summary", figures
: 0, imgs: 0, tables: 1, file: Cluster_Report_LTE_S6897_TIG0.html
at: push_warning (core/variant/variant_utility.cpp:1034)
GDScript backtrace (most recent call first):
[0] summarize_media_counts_by_category (res://scripts/ICProcessedFile.gd:776)
[1] validate_integrity (res://scripts/ICProcessedFile.gd:77)
[2] process_file (res://scripts/ICMain.gd:308)
🐞 [S6897] Cluster Report_LTE S6897 TIG0.docx converted to html & xml
WARNING: Media count mismatch in section " LTE VoLTE Intermediate Call KPI Summary", figures
```

*Figura 19: Vista de desarrollador durante el procesamiento de documentos*

Posteriormente, se implementaron funciones de análisis estructural encargadas de recorrer el HTML generado y detectar la presencia de los elementos definidos como obligatorios. Estas funciones identifican secciones específicas del documento, así como tablas y referencias gráficas, utilizando patrones y reglas previamente establecidas. El resultado de este análisis se almacena temporalmente en estructuras de datos internas, de acuerdo con el modelo definido en la etapa de diseño.

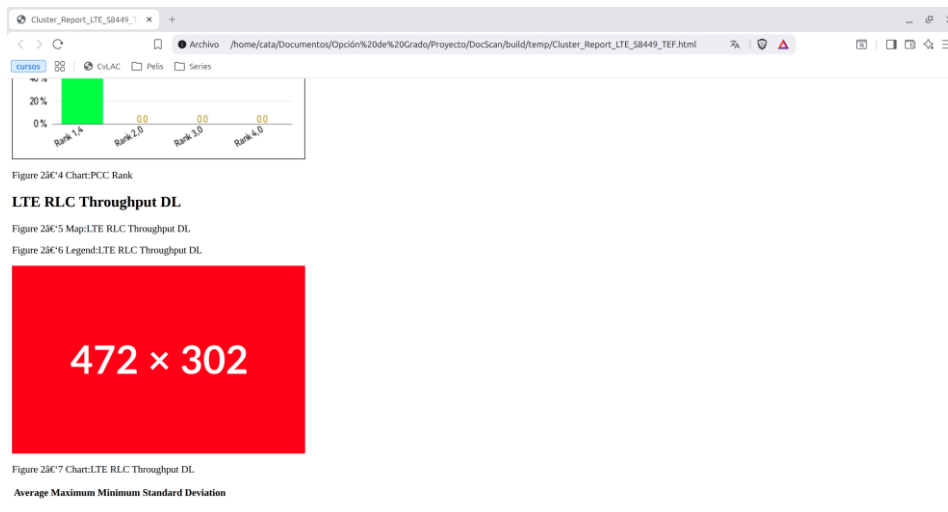


Figura 20: Visión preliminar en HTML de un reporte

Finalmente, se desarrolló una interfaz gráfica de resultados, desde la cual el usuario puede visualizar de forma clara el estado de validación de cada documento procesado. La interfaz presenta indicadores simples de cumplimiento y listas de elementos faltantes, permitiendo una revisión rápida sin necesidad de abrir manualmente el reporte técnico.

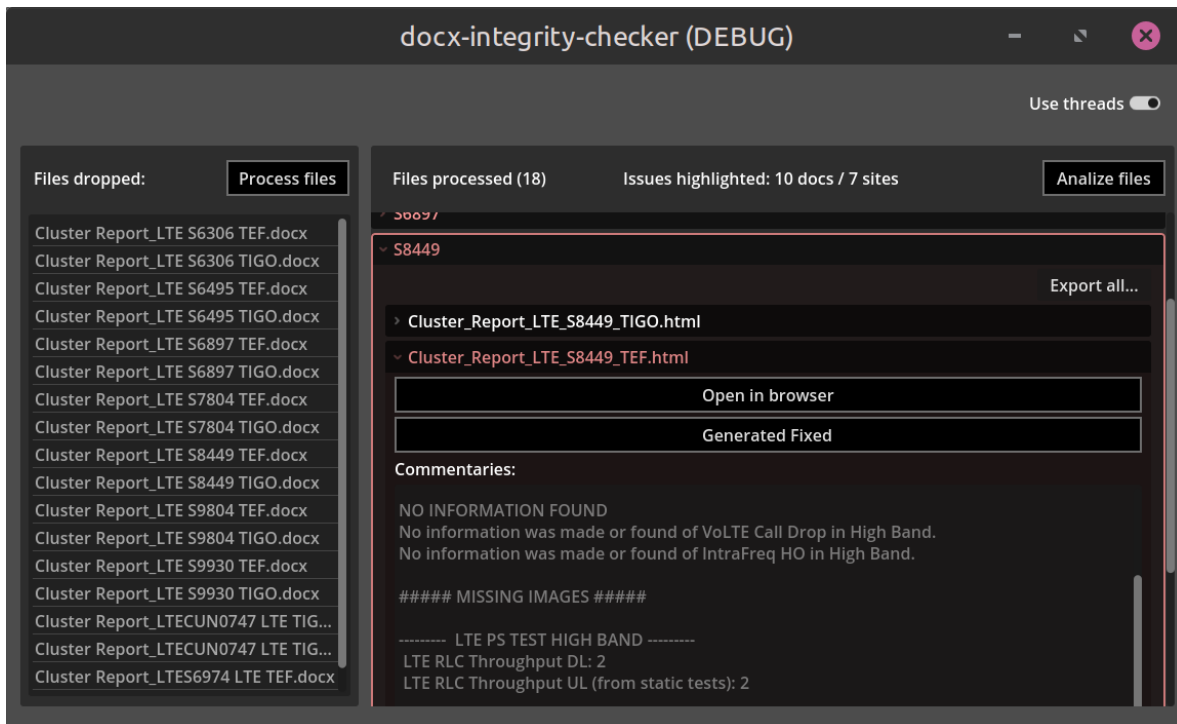


Figura 21: Visualización de estado de reportes

La implementación fue validada mediante pruebas funcionales realizadas con reportes reales del área de Radiofrecuencia, contando con la retroalimentación del supervisor de área, el tutor de trabajo y un compañero desarrollador. Estas pruebas permitieron verificar el correcto funcionamiento del sistema y su adecuación a las necesidades operativas reales, cumpliendo con los objetivos planteados para esta etapa del proyecto.

## **5.1 PRUEBAS Y VALIDACIÓN**

Se realizó una fase de pruebas con documentos reales generados por el área de Radiofrecuencia para verificar el funcionamiento del software y evaluar su aporte al proceso de validación de reportes técnicos; esta permitió comprobar que el sistema cumple con los requerimientos definidos, procesa archivos Word (.docx), identifica la estructura interna de los reportes y detecta la ausencia de elementos obligatorios, además de evidenciar resultados coherentes con la revisión manual. Asimismo, se evaluaron aspectos de usabilidad y claridad en la presentación de resultados, y la validación se llevó a cabo junto con personal del área, cuya retroalimentación confirmó la viabilidad de la solución como apoyo a la revisión preliminar.

### **5.1.1 ESCENARIOS DE PRUEBA**

Para la ejecución de las pruebas, se definieron diferentes escenarios que representan situaciones reales del proceso de validación de reportes técnicos. Estos escenarios incluyeron documentos completos, que cumplen con todos los elementos estructurales requeridos, así como reportes incompletos en los que se omitieron intencionalmente ciertas secciones, tablas o gráficas.

Con la siguiente figura se toma ejemplo de 24 documentos, eso equivale a 4 análisis de 6 sitios, posterior se arrastran al programa en el apartado de *Files Dropped* y se da clic en el botón de "Process files".

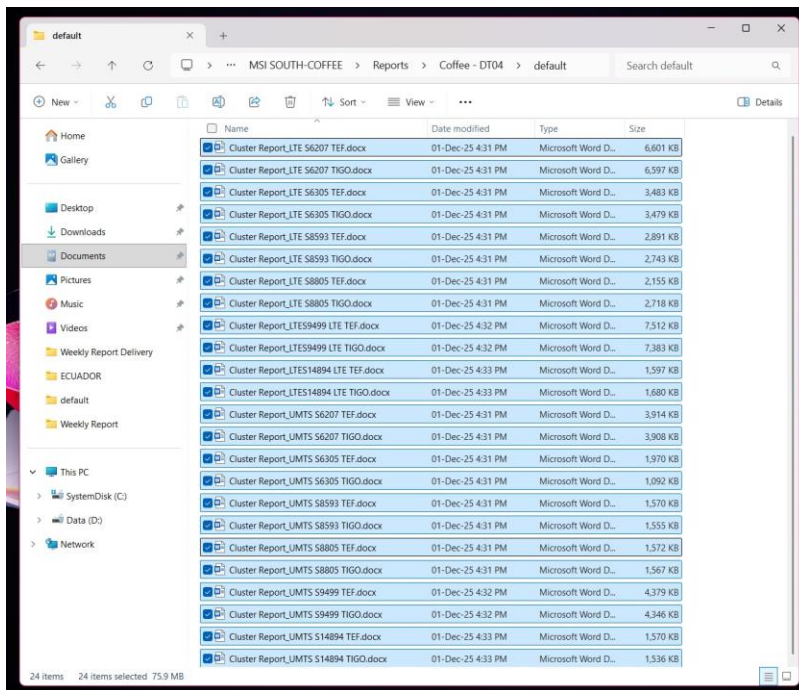


Figura 22: Selección de documentos a probar

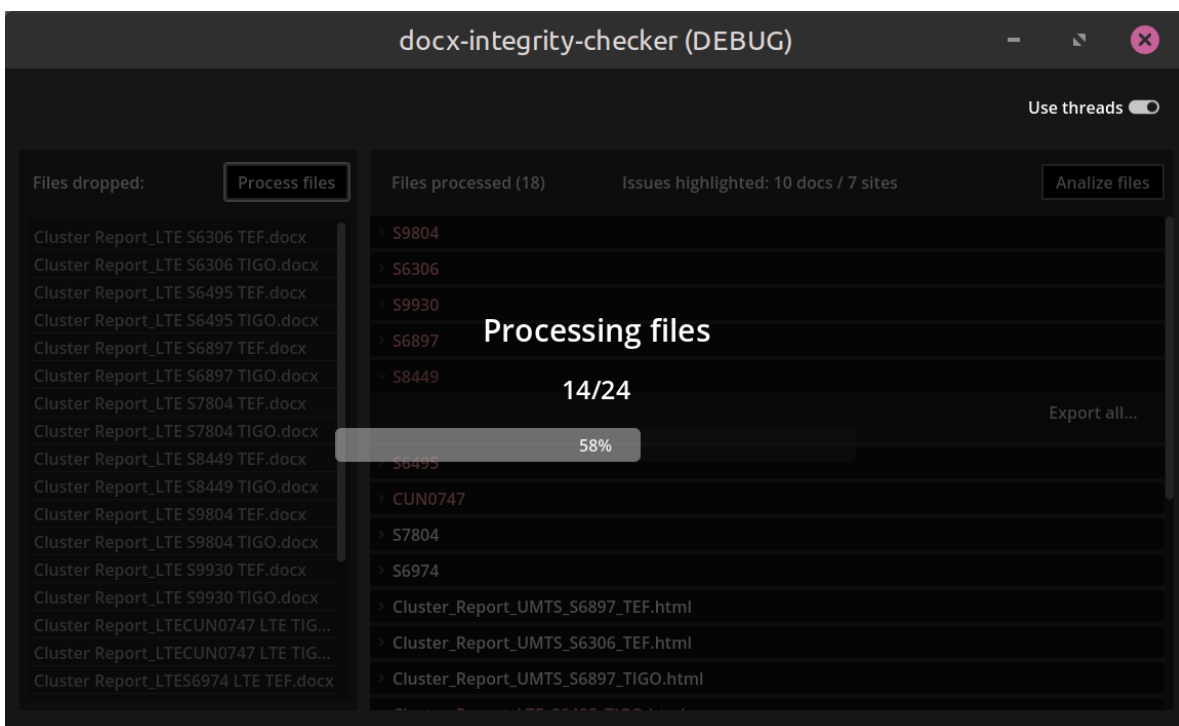


Figura 24: Interfaz de carga procesando documentos

Una vez procesado los documentos, como se puede observar en la siguiente imagen figura 24, visualmente se puede ver que los documentos se agruparon por sitios y los sitios que estén en rojo son en los que uno o varios de sus análisis salieron con errores (falta de gráficas o datos), la parte de superior se puede dar un total de cuantos sitios y documentos están mal.

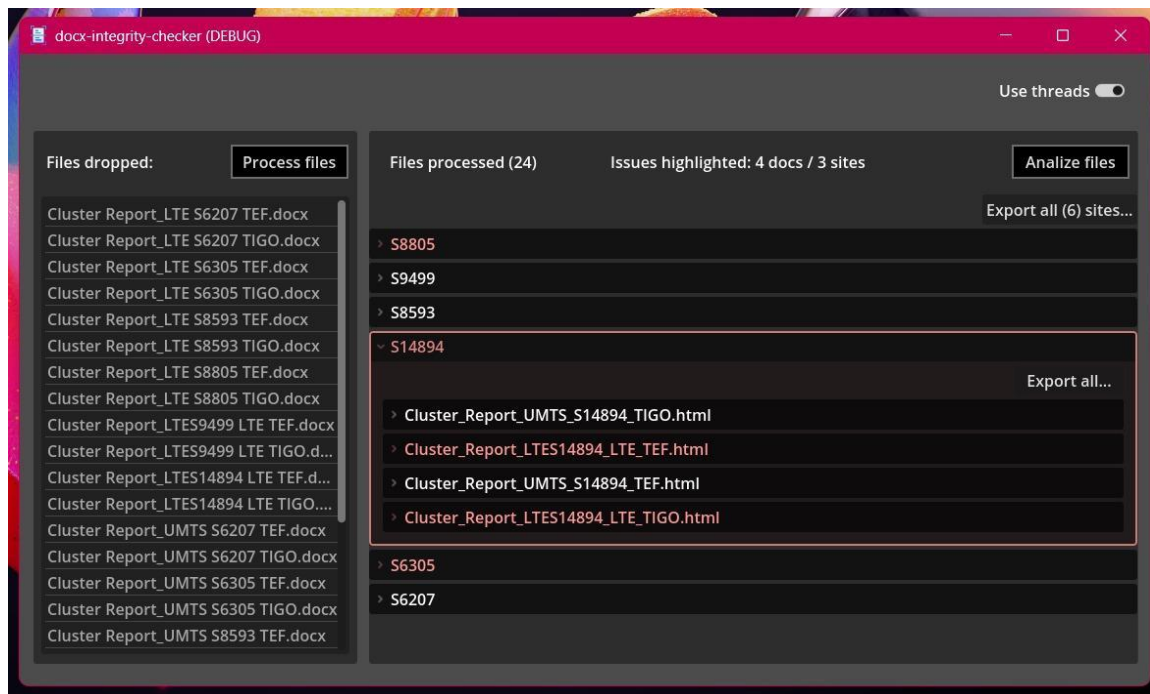
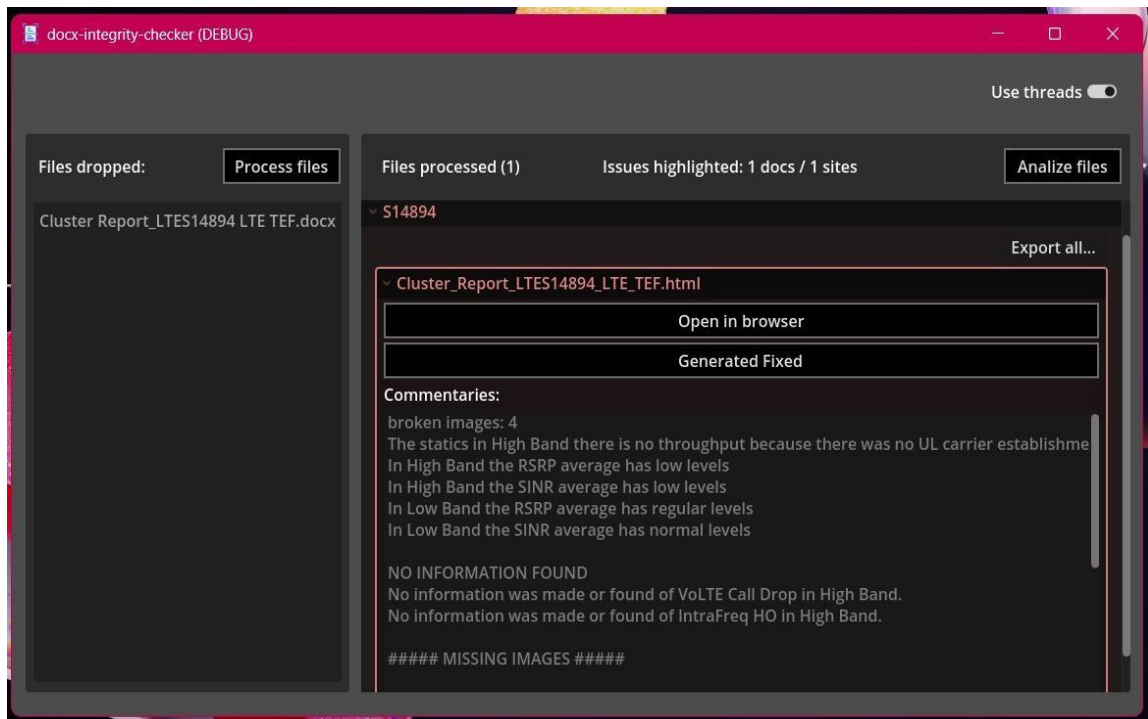


Figura 25: Resultado de procesamiento de documentos

A su vez se puede desplegar cada documento en la que se hizo comentarios de qué hace falta en él, como se muestra en la siguiente imagen.



*Figura 26: Más detalles sobre documento incompleto*

Cada documento se puede visualizar por HTML para antes de descargar la versión final del documento (**archivo exportado aparte, con comentarios añadidos**), dando clic en el botón de “Open in browser”, así pues, se abrirá una página para verificar que esté completo o rectificar de que si faltan gráficas o datos. También en cada análisis se deja comentarios como resumen del análisis que se hizo, entonces el programa añade por defecto los comentarios de acuerdo con las reglas que se implementó como evaluación, como se observa en la siguiente imagen.

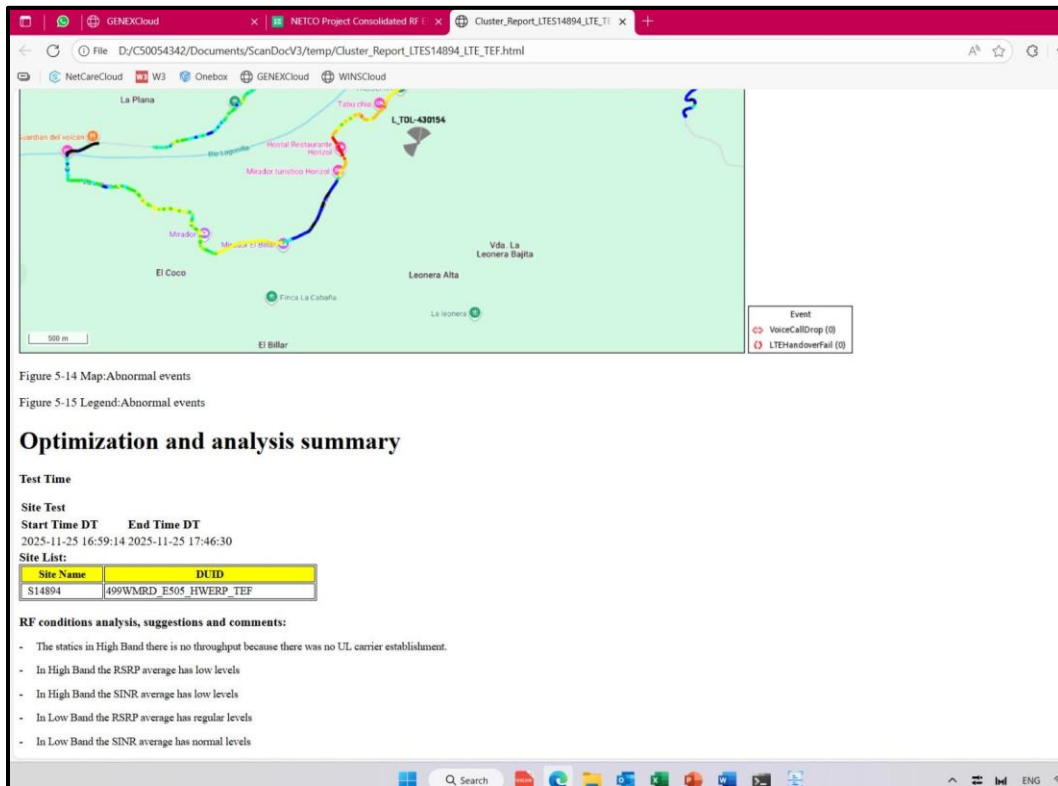


Figura 27: Visualización documento word en HTML

Una vez se haya hecho la previsualización, con la figura 26, hay 3 formas de exportar el documento final. La primera es exportar el documento final individualmente, la segunda es exportar todos los documentos de un sitio en específico y por último exportar todos los documentos de todos los sitios que hayan sido procesados que se deben agrupar en 6 carpetas (6 sitios como se decía al principio), como se muestra en la siguiente imagen.

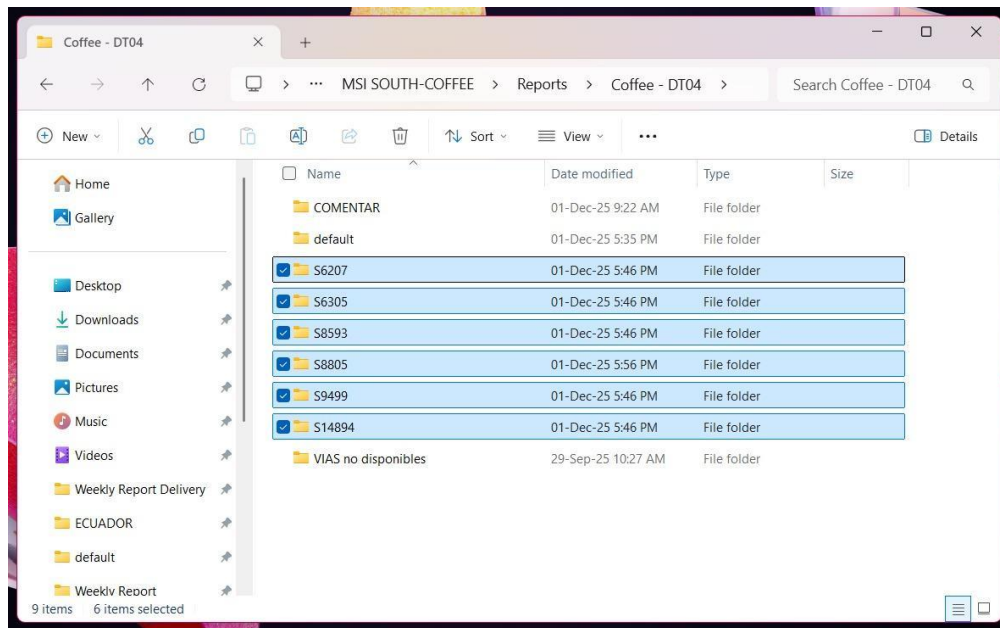


Figura 28: Exportación de documentos con comentarios añadidos

Y al abrir el documento final se obtienen los comentarios vistos con el HTML, y la persona que hace el análisis pueda complementar más de ello.

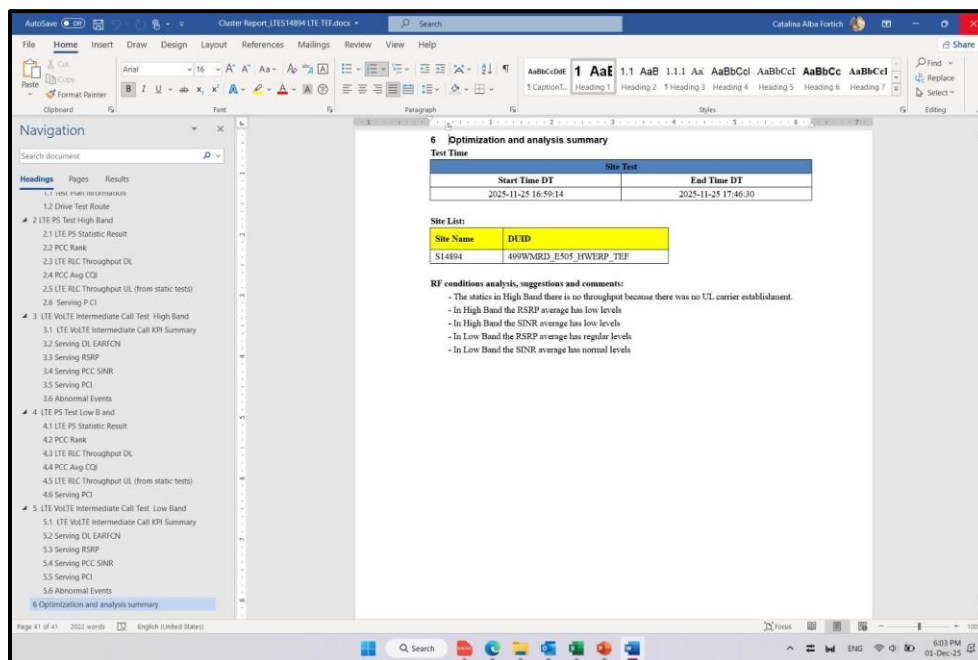


Figura 29: Visualización de comentarios añadidos por el programa

Los usuarios participantes en las pruebas —subjefe de área, tutor de trabajo y compañero desarrollador— interactuaron con el sistema para validar la facilidad de uso de la interfaz y la claridad de los resultados presentados. Los escenarios de prueba permitieron verificar que el sistema identifica correctamente los elementos faltantes y presenta los resultados de forma comprensible para el usuario final.

### 5.1.2 RESULTADOS OBTENIDOS

Los resultados obtenidos en las pruebas evidencian que el software desarrollado permite reducir significativamente el tiempo requerido para la validación preliminar de reportes técnicos, en comparación con el proceso manual tradicional. La automatización del análisis estructural facilitó la detección temprana de reportes incompletos, evitando la revisión detallada de documentos que no cumplen con los criterios mínimos establecidos.

Sujeto	Manual (1 doc)	Programa (1 doc)	Manual (4 docs)	Programa (4 docs)	Manual (40 docs = 10 sitios)	Programa (40 docs)
1	5 min	3 s	20 min	6 s	5 h 00 min	1 min 00 s
2	6 min	3 s	18 min	7 s	4 h 30 min	55 s
3	4 min	2 s	22 min	6 s	5 h 20 min	1 min 10 s
4	7 min	4 s	25 min	8 s	5 h 45 min	1 min 05 s
5	5 min	3 s	16 min	5 s	4 h 50 min	58 s
6	6 min	3 s	28 min	7 s	6 h 00 min	1 min 15 s
7	4 min	2 s	19 min	6 s	5 h 10 min	1 min 02 s

*Tabla 5: Resultados comparativos con los sujetos de prueba y ScanDoc*

La Tabla 5 presenta una comparación directa entre el tiempo requerido por los sujetos de prueba para realizar la validación manual y el tiempo de ejecución del software ScanDoc para los mismos volúmenes de documentos. En todos los casos evaluados (1 documento, 4 documentos y 40 documentos equivalentes a 10 sitios), se evidencia que la validación manual incrementa de forma proporcional al número de archivos, alcanzando rangos de minutos por documento y varias horas para lotes grandes.

En contraste, ScanDoc mantiene tiempos de procesamiento del orden de segundos por documento y alrededor de un minuto para 40 documentos, lo que confirma la eficiencia y escalabilidad del sistema.

Esta diferencia se explica porque el software automatiza la verificación estructural (presencia de secciones, tablas y gráficas), permitiendo identificar faltantes de manera inmediata y reduciendo significativamente la carga operativa del equipo durante escenarios de alta demanda.

Caso especial: validación con supervisor

<b>Caso</b>	<b>Carga</b>	<b>Equivale</b>	<b>Tiempo del programa</b>	<b>Revisión manual (equipo)</b>
Supervisor (prueba especial)	208 documentos	52 sitios	5–6 min	4 personas en paralelo, <b>5 h cada uno</b> (simultáneo)

*Tabla 6: Prueba de 208 documentos bajo la lupa del supervisor*

La Tabla 6 corresponde a un caso especial de validación ejecutado por el supervisor, en el que se evaluó el desempeño del sistema ante una carga masiva (208 documentos, equivalentes a 52 sitios), representativa de escenarios reales de alta demanda. En este contexto, el programa logra completar el procesamiento en aproximadamente 5–6 minutos, consolidando de forma automática el estado de los reportes y permitiendo priorizar rápidamente los casos con inconsistencias.

Por el contrario, aun distribuyendo la revisión manual entre 4 personas en paralelo, el esfuerzo requerido se mantiene elevado (alrededor de 5 horas por revisor), lo que evidencia que el cuello de botella operativo persiste incluso con más recurso humano.

Este resultado resalta la utilidad de ScanDoc como herramienta de apoyo para supervisión y control de calidad, al reducir de manera drástica el tiempo de respuesta y permitir que el equipo enfoque su trabajo en la revisión técnica y la toma de decisiones, en lugar de tareas repetitivas de verificación estructural.

## **6 CONCLUSIONES**

El desarrollo del presente proyecto permitió abordar de manera efectiva la problemática identificada en el área de Radiofrecuencia de Huawei-Colombia, relacionada con el alto consumo de tiempo y esfuerzo requerido para la validación preliminar de reportes técnicos generados tras las pruebas de campo en redes LTE y UMTS. A partir del análisis del proceso actual, se evidenció la necesidad de contar con una solución automatizada que apoyara la revisión estructural de dichos documentos.

Como resultado del proyecto, se diseñó e implementó un software capaz de procesar reportes técnicos en formato Word (.docx) y validar de forma automática la presencia de elementos estructurales obligatorios, tales como secciones, tablas y gráficas. La solución desarrollada cumple con los requerimientos técnicos y operativos definidos, demostrando que es posible automatizar esta etapa del proceso sin alterar el documento original, mediante la generación de un documento final exportado que incluye comentarios, observaciones y datos relevantes del reporte, sin depender de métodos basados en reconocimiento visual u OCR.

No obstante, el sistema presenta algunas limitaciones. En primer lugar, la validación se restringe al aspecto estructural y formal del documento, por lo que no evalúa la calidad técnica de los parámetros de radiofrecuencia ni la pertinencia del análisis incluido en los reportes, tareas que continúan requiriendo el criterio experto del ingeniero. Asimismo, el desempeño del sistema depende de la adherencia de los documentos a las plantillas establecidas y de la correcta conversión del archivo .docx a formato intermedio.

Durante el desarrollo del proyecto se enfrentaron dificultades técnicas, principalmente asociadas a la heterogeneidad de los documentos Word, el manejo de estructuras complejas en XML y la preservación de la jerarquía de secciones y elementos gráficos durante el proceso de conversión con Pandoc. Estas situaciones requirieron ajustes iterativos en la lógica de validación y el diseño de reglas tolerantes a variaciones.

En la fase de pruebas, el sistema evidenció una reducción significativa en el tiempo requerido para la validación preliminar de los reportes, en comparación con el proceso manual tradicional. Asimismo, la aplicación de criterios de validación uniformes permitió

obtener resultados consistentes, independientemente del usuario que ejecutara la revisión, lo cual contribuye a mejorar la eficiencia y confiabilidad del proceso.

El uso de herramientas como Godot Engine y Pandoc demostró ser adecuado para el desarrollo de soluciones internas orientadas a la automatización de procesos documentales, integrando de manera efectiva la interfaz gráfica, el procesamiento estructural y la visualización de resultados. La validación realizada con personal del área de Radiofrecuencia confirmó la utilidad práctica del software como apoyo al control de calidad de los reportes técnicos.

Finalmente, se concluye que el sistema desarrollado cumple con los objetivos planteados y constituye una herramienta de apoyo efectiva para la validación preliminar de reportes técnicos. Sin embargo, su evolución futura podría fortalecer aún más su aporte al área de Radiofrecuencia, ampliando sus capacidades y adaptándose a escenarios más complejos de análisis documental.

## **6.1 TRABAJOS FUTUROS**

A partir de los resultados obtenidos, se identifican diversas oportunidades de mejora y ampliación del sistema desarrollado. En primer lugar, se propone la incorporación de mecanismos de almacenamiento de resultados que permitan llevar un historial de validaciones, facilitando el seguimiento y la trazabilidad de los reportes procesados.

Adicionalmente, el sistema podría ampliarse para soportar otros formatos de archivo, como PDF o Excel, lo que incrementa su aplicabilidad en diferentes procesos del área. Otra línea de trabajo futuro corresponde a la incorporación de técnicas de inteligencia artificial orientadas al análisis del contenido técnico de los reportes, como la evaluación automática de parámetros de radiofrecuencia o la detección de anomalías en los resultados.

Asimismo, se sugiere la integración del software con otras herramientas internas de la empresa, de modo que los resultados de la validación puedan formar parte de un flujo de trabajo más amplio de aseguramiento de calidad. Finalmente, se contempla la mejora de la interfaz gráfica y la experiencia de usuario, incorporando visualizaciones más detalladas y

opciones de configuración de criterios de validación según las necesidades del proyecto o del operador.

## REFERENCIAS

- [1] Wikipedia, "Office Open XML," Wikipedia, la enciclopedia libre. [https://es.wikipedia.org/wiki/Office\\_Open\\_XML](https://es.wikipedia.org/wiki/Office_Open_XML) (accedido oct. 11, 2025).
- [2] noraj, "Respuesta a 'How can doc/docx files be converted to markdown or structured text?'" Stack Overflow, may. 2013. <https://stackoverflow.com/questions/16383237/how-can-doc-docx-files-be-converted-to-markdown-or-structured-text> (accedido oct. 13, 2025).
- [3] J. MacFarlane, "Pandoc - A universal document converter," Pandoc. <https://pandoc.org/> (accedido oct. 18, 2025).
- [4] salmaamari, "Huawei Products Technical Solutions and Marketing Save and Share Costing for Infrastructures of Multiple Operators Section 1," Huawei Enterprise Support Community. <https://forum.huawei.com/enterprise/intl/en/thread/764896374068830208> (accedido oct. 18, 2025).
- [5] Y. Xu, Y. Xu, T. Lv, L. Cui, F. Wei, G. Wang, Y. Lu, D. Florencio, C. Zhang, W. Che, M. Zhang, y L. Zhou, "LayoutLMv2: Multi-modal pre-training for visually-rich document understanding," en Proc. 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP), 2021, pp. 2579-2591. [En línea]. Disponible: <https://aclanthology.org/2021.acl-long.201.pdf>
- [6] Y. Huang, T. Lv, L. Cui, Y. Lu, y F. Wei, "LayoutLMv3: Pre-training for document AI with unified text and image masking," arXiv preprint arXiv:2204.08387, 2022. [En línea]. Disponible: <https://arxiv.org/pdf/2204.08387>
- [7] Z. Li, A. Abulaiti, Y. Lu, X. Chen, J. Zheng, H. Lin, X. Han, y L. Sun, "ReadDoc: A unified benchmark for realistic document structured extraction," arXiv preprint arXiv:2409.05137, 2024. [En línea]. Disponible: <https://arxiv.org/abs/2409.05137>
- [8] Telecom Infra Project, "Testing for NaaS environment white paper," TIP NaaS MORAN, White Paper v1.0. [En línea]. Disponible: [https://cdn.mediavalet.com/usva/telecominfraproject/xc2D7LdMJky5R5TDxENdPQ/0WM2q6D\\_70mDd7w6u1uhRw/Original/TIP\\_NaaS\\_MORAN\\_Testing\\_for\\_NaaS\\_Environment\\_White\\_Paper\\_v10\\_FINAL\\_GREEN\\_-\\_Public\\_Access.pdf](https://cdn.mediavalet.com/usva/telecominfraproject/xc2D7LdMJky5R5TDxENdPQ/0WM2q6D_70mDd7w6u1uhRw/Original/TIP_NaaS_MORAN_Testing_for_NaaS_Environment_White_Paper_v10_FINAL_GREEN_-_Public_Access.pdf)

- [9] Ericsson, "Ericsson mobility report," Ericsson, Tech. Rep., jun. 2025. [En línea]. Disponible: <https://www.ericsson.com/49e9b6/assets/local/reports-papers/mobility-report/documents/2025/ericsson-mobility-report-june-2025.pdf>
- [10] S. E. Casas, "Automatización Del Proceso 'Validación Facturación Outbound'", Edu.co, 2023. [En línea]. Disponible en: <https://bibliotecadigital.udea.edu.co/server/api/core/bitstreams/d18a000f-eb04-483b-8dad-655e0b638a50/content> [Consultado: 10-oct-2025].
- [11] K. Daniela, H. Rada, N. Catalina, y P. Ocampo, "AUTOMATIZACIÓN DEL REPORTE DE ENVÍOS MANUALES EN EL ÁREA DE ATENCIÓN AL CLIENTE DE UNA EMPRESA ENERGÉTICA", Edu.co. [En línea]. Disponible en: <https://repository.universidadean.edu.co/server/api/core/bitstreams/4174a3de-1add-4064-b4f4-43e6eb22aebb/content> [Consultado: 19-oct-2025].
- [12] J. D. F. Bonilla, "Diseño de macros para optimizar el tiempo de recolección de datos en los cruces de bases en las herramientas de aprendizaje de las cuentas propias de Emtelco S.A.S", Edu.co, 2023. [En línea]. Disponible en: <https://repository.ucc.edu.co/server/api/core/bitstreams/4a9f144c-1826-4fdc-b573-9c06a97335d1/content> [Consultado: 10-oct-2025].
- [13] E. P. O. A. de Planeación y Estudios Sectoriales, "INFORME DE GESTIÓN Año 2021", Gov.co. [En línea]. Disponible en: [https://www.mintic.gov.co/portal/715/articles-198584\\_recurso\\_1.pdf?utm\\_source](https://www.mintic.gov.co/portal/715/articles-198584_recurso_1.pdf?utm_source) [Consultado: 19-oct-2025].
- [14] J. F. Fuentes Bautista, "Análisis del impacto de la Automatización Robótica de Procesos (RPA) en el sector educativo en", Edu.co. [En línea]. Disponible en: [https://repository.unilibre.edu.co/bitstream/handle/10901/31803/ARTICULO%20FINAL%20INVESTIGACION.pdf?isAllowed=y&sequence=1&utm\\_source](https://repository.unilibre.edu.co/bitstream/handle/10901/31803/ARTICULO%20FINAL%20INVESTIGACION.pdf?isAllowed=y&sequence=1&utm_source) [Consultado: 19-oct-2025].
- [15] Cata-AF. (s. f.). *ScanDoc* [Repositorio de software]. GitHub. <https://github.com/Cata-AF/ScanDoc>
- [16] 3GPP, "Release 1999," 3GPP. <https://www.3gpp.org/specifications-technologies/releases/release-1999> (accedido oct. 2025).
- [17] 3GPP, "Release 8," 3GPP. <https://www.3gpp.org/specifications-technologies/releases/release-8> (accedido oct. 2025).

- [18] 3GPP, "Release 15," 3GPP. <https://www.3gpp.org/specifications-technologies/releases/release-15> (accedido oct. 2025).
- [19] Wikipedia, "Drive testing," Wikipedia, the free encyclopedia. [https://en.wikipedia.org/wiki/Drive\\_testing](https://en.wikipedia.org/wiki/Drive_testing) (accedido oct. 2025).
- [20] Ecma International, "ECMA-376 Office Open XML file formats," 2021. <https://ecma-international.org/publications-and-standards/standards/ecma-376/> (accedido oct. 2025).
- [21] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, y F. Yergeau, "Extensible Markup Language (XML) 1.0 (Fifth Edition)," W3C Recommendation, Nov. 2008. <https://www.w3.org/TR/REC-xml/> (accedido oct. 2025).
- [22] J. MacFarlane, "Pandoc - a universal document converter," Pandoc. <https://pandoc.org/> (accedido oct. 2025).
- [23] Godot Engine, "License," Godot Engine. <https://godotengine.org/license/> (accedido oct. 2025).
- [24] ISO, "Quality assurance," ISO. <https://www.iso.org/quality-management/quality-assurance> (accedido oct. 2025).
- [25] Quality Gurus, "Design Review, Verification and Validation (ISO 9000:2015 definitions)," Quality Gurus. <https://www.qualitygurus.com/design-review-verification-and-validation/> (accedido oct. 2025).
- [26] Upm.es. [En línea]. Disponible en: [https://oa.upm.es/44209/3/TFM\\_FRANCISCO\\_MAESTRE\\_TORREBLANCA.pdf](https://oa.upm.es/44209/3/TFM_FRANCISCO_MAESTRE_TORREBLANCA.pdf). [Consultado: 08-abr-2026].

## LISTA DE FIGURAS

Figura 1. Cronograma en diagrama de Gantt	11
Figura 2. Ejemplo visual de organización de datos de un documento Word	16
Figura 3. Relación entre Marco Teórico y Marco Conceptual	31
Figura 4. Ciclo de vida RUP	39
Figura 5. Prototipo del software ScanDoc 1	41
Figura 6. Prototipo del software ScanDoc 2	42
Figura 7. Prototipo del software ScanDoc 3	43
Figura 8. Vista preliminar del documento HTML (sin gráficos ni elementos finales)	43
Figura 9. Vista preliminar del documento HTML	44
Figura 10. Prototipo del software ScanDoc 4	45
Figura 11. Prototipo del software ScanDoc 5	46
Figura 12. Reporte antes y después del procesamiento y exportación	46
Figura 13. Flujo general del software	47
Figura 14. Diagrama de datos relación entidades	48
Figura 15. Ícono de ScanDoc	50
Figura 16. Interfaz de Scandoc inicial	51
Figura 17. Arrastrando documentos a ScanDoc	52
Figura 18. Interfaz de carga procesando N documentos	52
Figura 19. Vista de desarrollador durante el procesamiento de documentos	53
Figura 20. Visión preliminar en HTML de un reporte	54
Figura 21. Visualización de estado de reportes	54
Figura 22. Selección de documentos a probar	56
Figura 24. Interfaz de carga procesando documentos	56
Figura 25. Resultado de procesamiento de documentos	57
Figura 26. Más detalles sobre documento incompleto	58
Figura 27. Visualización documento word en HTML	59
Figura 28. Exportación de documentos con comentarios añadidos	60
Figura 29. Visualización de comentarios añadidos por el programa	60

## LISTA DE TABLAS

Tabla 1. Actividades de los Objetivos	7
Tabla 2. Cronograma de Actividades	8
Tabla 3. Requisitos Funcionales	36
Tabla 4. Requisitos No Funcionales	37
Tabla 5. Resultados comparativos con los sujetos de prueba y ScanDoc	61
Tabla 6. Prueba de 208 documentos bajo la lupa del supervisor	62