

Dounia Lakdimi Ahdri

**Desenvolupament d'un Sistema Intel·ligent per a la Generació automatitzada
d'Informes**

TREBALL DE FI DE GRAU

dirigit per Maria Ferré Bergadà

Grau d'Enginyeria Informàtica



UNIVERSITAT ROVIRA I VIRGILI

Tarragona

2025

Resum.

El present projecte, desenvolupat a l'empresa IDIADA, té com a objectiu principal dissenyar i implementar un sistema intel·ligent capaç de generar informes de manera automatitzada per als assajos de Compatibilitat Electromagnètica (EMC). Mitjançant la integració de tecnologies d'intel·ligència artificial i aprenentatge automàtic, el sistema supera les limitacions de l'eina existent i incorpora capacitats analítiques avançades.

La metodologia adoptada ha inclòs l'anàlisi de les necessitats específiques de l'equip EMC, el desenvolupament d'algoritmes d'IA per al processament de dades CSV, i la implementació de funcionalitats com la indexació automàtica, la generació d'enllaços dinàmics i l'anàlisi intel·ligent dels resultats. El sistema resultant no només automatitza la generació d'informes, sinó que també proporciona interpretacions preliminars dels assajos, detecta patrons i anomalies, i estableix comparatives amb proves anteriors.

Els resultats obtinguts demostren una millora significativa en l'eficiència del procés de documentació, reduint el temps dedicat a tasques repetitives i millorant la qualitat dels informes generats.

Resumen.

Este proyecto, desarrollado en la empresa IDIADA, tiene como objetivo principal diseñar e implementar un sistema inteligente capaz de generar informes de manera automatizada para los ensayos de Compatibilidad Electromagnética (EMC). Mediante la integración de tecnologías de inteligencia artificial y aprendizaje automático, el sistema supera las limitaciones de la herramienta existente e incorpora capacidades analíticas avanzadas.

La metodología adoptada ha incluido el análisis de las necesidades específicas del equipo EMC, el desarrollo de algoritmos de IA para el procesamiento de datos CSV, y la implementación de funcionalidades como la indexación automática, la generación de enlaces dinámicos y el análisis inteligente de los resultados. El sistema resultante no solo automatiza la generación de informes, sino que también proporciona interpretaciones preliminares de los ensayos, detecta patrones y anomalías, y establece comparativas con pruebas anteriores.

Los resultados obtenidos demuestran una mejora significativa en la eficiencia del proceso de documentación, reduciendo el tiempo dedicado a tareas repetitivas y mejorando la calidad de los informes generados.

Abstract.

This project, developed at IDIADA company, aims to design and implement an intelligent system capable of automatically generating reports for Electromagnetic Compatibility (EMC) testing. Through the integration of artificial intelligence and machine learning technologies, the system overcomes existing tool limitations and incorporates advanced analytical capabilities.

The adopted methodology included analyzing specific EMC team requirements, developing AI algorithms for CSV data processing, and implementing functionalities such as automatic indexing, dynamic link generation, and intelligent results analysis. The resulting system not only automates report generation but also provides preliminary test interpretations, detects patterns and anomalies, and establishes comparisons with previous tests.

The obtained results demonstrate significant improvement in documentation process efficiency, reducing time spent on repetitive tasks and enhancing generated report quality.

Índex

1	INTRODUCCIÓ	7
1.1	ORIGEN DEL PROJECTE.....	7
1.2	PROBLEMÀTICA IDENTIFICADA.....	7
1.3	OBJECTIUS DEL PROJECTE.....	7
1.3.1	<i>Objectius Específics:</i>	7
1.3.2	<i>Impacte Estratègic:</i>	8
2	DESCRIPCIÓ GENERAL DEL PROJECTE	9
2.1	ENTORN EMPRESARIAL.....	9
2.2	DARWIN.....	9
2.2.1	<i>Què és DARWIN?</i>	9
2.2.2	<i>Funcionalitats principals de DARWIN</i>	9
2.2.3	<i>Arquitectura i metodologia de treball</i>	10
2.2.4	<i>Beneficis i avantatges de DARWIN</i>	10
2.2.5	<i>Estandardització i automatització</i>	10
2.2.6	<i>Relació amb altres aplicacions</i>	11
2.2.7	<i>Interfície d'usuari i fluxos de treball</i>	11
2.3	NECESSITATS IDENTIFICADES.....	11
2.3.1	<i>Control de la indexació automàtica</i>	11
2.3.2	<i>Automatització d'enllaços a la test matrix</i>	11
2.3.3	<i>Validació i agrupació semàntica de textos mitjançant LLM</i>	12
2.3.4	<i>Substitució per una eina que calculi els pics d'una gràfica mitjançant un LLM</i>	12
2.3.5	<i>Anàlisi detallada i comparativa dels resultats dels assajos</i>	12
2.4	TIPOLOGIA D'ASSAJOS, PUNTS DE TEST I MODES D'OPERACIÓ.....	13
2.5	PREVISIONS D'ÚS.....	14
3	REQUISITS	15
3.1	REQUISITS FUNCIONALS.....	15
3.2	REQUISITS NO FUNCIONALS.....	15
4	ANÀLISI DELS REQUISITS FUNCIONALS	16
4.1	SISTEMA D'INDEXACIÓ AUTOMÀTICA.....	16
4.2	GENERACIÓ DE ENLLAÇOS A UNA <i>TEST MATRIX</i>	17
4.3	GENERACIÓ DE UN INFORME.....	18
5	DISSENY	20
5.1	LLM.....	20
5.1.1	<i>Definició i Característiques</i>	20
5.1.2	<i>Arquitectura Transformer</i>	20
5.1.3	<i>Procés d'Entrenament</i>	20
5.1.4	<i>Capacitats i Aplicacions</i>	21
5.1.5	<i>Exemples Destacats</i>	21
5.1.6	<i>Avantatges</i>	21
5.1.7	<i>Limitacions i Reptes</i>	22
5.1.8	<i>Tendències Futures</i>	22
5.2	DISSENY DE L'AGENT.....	23
5.2.1	<i>Llibreria Agentic-Core</i>	23
5.2.2	<i>Components Fonamentals de l'Agent</i>	23
5.2.3	<i>Gestió i Formatatge de Respostes</i>	23
5.2.4	<i>Configuració i Execució Modular</i>	24
5.3	ARQUITECTURA BASADA EN AWS.....	24

6	IMPLEMENTACIÓ	26
6.1	TECNOLOGIES USADES	26
6.1.1	<i>Amazon Textract</i>	26
6.1.2	<i>Amazon Bedrock</i>	26
6.1.3	<i>Aspose.Words</i>	26
6.2	DESENVOLUPAMENT DEL SISTEMA D'INDEXACIÓ AUTOMÀTICA	27
6.2.1	<i>Extracció de dades</i>	27
6.2.2	<i>Detecció de títols</i>	30
6.2.3	<i>Correcció de numeració al document</i>	30
6.3	GENERACIÓ DE ENLLAÇOS A UNA <i>TEST MATRIX</i>	31
6.3.1	<i>Extracció d'informació del job</i>	31
6.3.2	<i>Manipulació del document Word amb Aspose.Words</i>	32
6.3.3	<i>Generació programàtica dels enllaços</i>	32
6.3.4	<i>Inserció i validació dels enllaços</i>	32
6.4	IMPLEMENTACIÓ DE LA DETECCIÓ DE PICS MITJANÇANT UN AGENT	32
6.4.1	<i>Detecció tradicional vs detecció amb LLM</i>	33
6.4.2	<i>Definició del prompt per a l'Agent</i>	33
6.4.3	<i>Execució del model i generació dels resultats</i>	34
6.4.4	<i>Validació i control de qualitat</i>	34
6.5	IDENTIFICACIÓ DE CONFIGURACIONS MITJANÇANT UN AGENT	35
6.6	GENERACIÓ DE GRÀFIQUES PERSONALITZADES AMB PROGRAMACIÓ TRADICIONAL	35
6.6.1	<i>Requisits específics de les gràfiques</i>	36
6.6.2	<i>Implementació i integració amb l'Agent</i>	36
6.6.3	<i>Consideracions finals</i>	37
6.7	GENERACIÓ I OPLIMENT AUTOMÀTIC DEL DOCUMENT FINAL	37
6.7.1	<i>Procés general de generació</i>	37
6.8	PROCESSAMENT COMPLET DE LES DADES D'UN ASSAIG	38
6.8.1	<i>Entrada de dades</i>	38
6.8.2	<i>Generació i concatenació d'informes</i>	38
6.8.3	<i>Gestió automàtica de la indexació</i>	38
6.9	ANÀLISI D'UN ASSAIG	39
6.9.1	<i>Entrada del sistema</i>	39
6.9.2	<i>Objectius principals de l'anàlisi</i>	39
6.10	GENERACIÓ DE DADES SINTÈTIQUES	39
6.10.1	<i>Característiques Tècniques</i>	40
6.11	IMPLEMENTACIÓ DEL <i>SUBSTREAM</i> AL FRONT-END DE DOCS	40
6.11.1	<i>Estratègia d'Implementació Adoptada</i>	40
6.11.2	<i>Arquitectura del Stream a DOCS</i>	40
7	AVALUACIÓ	42
7.1	DESENVOLUPAMENT DEL SISTEMA D'INDEXACIÓ AUTOMÀTICA	42
7.1.1	<i>Entrada</i>	42
	<i>Document mal indexat</i>	42
7.1.2	<i>Sortida</i>	42
7.1.3	<i>Debug</i>	43
7.2	GENERACIÓ DE ENLLAÇOS A UNA <i>TEST MATRIX</i>	44
7.2.1	<i>Entrada</i>	45
7.2.2	<i>Sortida</i>	45
7.3	IMPLEMENTACIÓ DE LA DETECCIÓ DE PICS MITJANÇANT UN AGENT	46
7.3.1	<i>Sortida</i>	46
7.4	ANÀLISI D'UN ASSAIG	46
7.5	RESULTATS FINALS	51
7.5.1	<i>Entrada</i>	51
7.5.2	<i>Validació funcional</i>	55

7.5.3	<i>Validació de la sortida generada</i>	56
7.5.4	<i>Comparació amb el sistema original</i>	56
7.5.5	<i>Temps d'execució i limitacions</i>	57
7.5.6	<i>Millores futures</i>	57
8	CONCLUSIONS:	61
9	REFERÈNCIES	62
10	ANNEXES:	63
10.1	POSADA EN MARXA	63
10.1.1	<i>Configuració de l'Emmagatzematge (AWS S3)</i>	63
10.1.2	<i>Configuració de Llicències</i>	63
10.1.3	<i>Instal·lació del Framework</i>	63
10.1.4	<i>Configuració del Sistema</i>	63
10.1.5	<i>Execució i Verificació</i>	63
10.2	REQUERIMENTS	64
10.2.1	<i>Requeriments de programari</i>	64
10.3	REQUERIMENTS DE LLICÈNCIES	64
10.3.1	<i>Llicències Comercials</i>	64
10.4	REQUERIMENTS DE INFRAESTRUCTURA	64
10.4.1	<i>Serveis en el Núvol</i>	64
10.5	CODIS	65
10.5.1	<i>Llistem csv's en un bucket de s3</i>	65
10.5.2	<i>Llistem documents en un bucket de s3</i>	65
10.5.3	<i>Obtenim credencials amb diferents opcions</i>	66
10.5.4	<i>Configurem client bedrock i sessió S3</i>	67
10.5.5	<i>Codi de un agent amb una Foramatter Tool:</i>	67

Índex de figures

FIGURA 1. FLUX SISTEMA D'INDEXACIÓ	16
FIGURA 2. FLUX GENERACIÓ D'ENLLAÇOS	17
FIGURA 3. FLUX GENERACIÓ D'INFORMES	18
FIGURA 4. EXTRACCIÓ DE TÍTOLS	29
FIGURA 5. <i>SUBSTREAM</i> DETECTOR DE PICS	41
FIGURA 6. EXEMPLE D'ENTRADA AMB MALA INDEXACIÓ.....	42
FIGURA 7. EXEMPLE DE TÍTOL MAL INDEXAT	42
FIGURA 8. ÍNDEX CORREGIT.....	43
FIGURA 9. TÍTOL CORREGIT.....	43
FIGURA 10. <i>DEBUG</i> EXTREURE TÍTOLS	44
FIGURA 11. <i>DEBUG</i> DETECTOR DE TÍTOLS	44
FIGURA 12. TAULA DE PLANTILLA	45
FIGURA 13. <i>TEST MATRIX</i>	45
FIGURA 14 TÍTOL SECCIÓ 1.1.1	46
FIGURA 15. DETECTOR DE PICS	46
FIGURA 16. DEFINICIÓ BANDES.....	48
FIGURA 17. COMPARATIVA	49
FIGURA 18. RESUM	50
FIGURA 19. EXEMPLE DADES D'ASSAIG AN.....	51
FIGURA 20. EXEMPLE DADES D'ASSAIG CP.....	52
FIGURA 21. EXEMPLE DADES D'ASSAIG RE	52
FIGURA 22. ESTRUCTURA DE LES DADES	53
FIGURA 23. PLANTILLA ENTRADA PÀGINA 1 I 2	54
FIGURA 24. PLANTILLA D'ENTRADA PAGINA 3 I 4	54
FIGURA 25. PLANTILLA D'ENTRADA PÀGINA 5 I 6	55
FIGURA 26. BANDES B1-B4 I NFC	58
FIGURA 27. BANDES B5 I FM.....	58
FIGURA 27. BANDES B5 I FM.....	59
FIGURA 28. BANDES 4M BOS AVERAGE I PEAK.....	59
FIGURA 28. BANDES 4M BOS AVERAGE I PEAK.....	59
FIGURA 28. BANDES 4M BOS AVERAGE I PEAK.....	59

Glossari

Assaig: Assaig és un procediment tècnic controlat que s'aplica a un dispositiu o sistema electrònic per avaluar el seu comportament davant emissions o interferències electromagnètiques, segons normes establertes.

Test Point: Test Point d'accés físic dissenyat dins un circuit electrònic (com una placa de circuit o un connector) que permet mesurar, injectar o monitoritzar senyals elèctrics durant proves, diagnosi o validació, com les proves de compatibilitat electromagnètica (EMC).

Banda: Banda és un interval de freqüències electromagnètiques delimitat per una freqüència mínima i una màxima. En assaigs EMC, les bandes s'utilitzen per classificar i analitzar tant les emissions com la susceptibilitat dels dispositius en diferents zones de l'espectre.

S3: Amazon S3 és un servei d'emmagatzematge al núvol que proporciona emmagatzematge escalable, segur i durador d'objectes (fitxers), accessible a través d'Internet.

AWS: AWS és la plataforma de serveis al núvol d'Amazon. Ofereix una àmplia gamma de serveis informàtics a través d'Internet, com ara potència de càlcul, emmagatzematge, bases de dades, xarxa, intel·ligència artificial, i molt més.

Aspose: Aspose.Words és una llibreria de programari que permet crear, editar, convertir, imprimir, protegir i analitzar documents Word (.doc, .docx, .rtf, etc.)

Bucket: Un bucket és el contenidor bàsic de dades dins del servei Amazon S3 (Simple Storage Service). És on es desen els fitxers (anomenats objectes) al núvol.

Report: Un report és un document escrit que recull de manera organitzada i estructurada informació, dades, observacions o resultats d'una anàlisi, estudi, assaig o projecte, amb l'objectiu de comunicar conclusions i recomanacions a una audiència específica.

Plot: El plot és la representació visual de dades o funcions en un sistema de coordenades, utilitzant punts, línies, barres o altres formes per mostrar la relació entre variables i facilitar la seva interpretació.

Agents: Són entitats intel·ligents que actuen de manera autònoma i són capaços de gestionar fluxos de treball, prendre decisions i coordinar diferents passos del procés. L'agent utilitza el model LLM per comprendre i generar llenguatge natural, interpretar instruccions i adaptar la seva actuació segons el context i els objectius establerts.

Tools (Eines): Són components o mòduls especialitzats que l'agent pot invocar per dur a terme tasques concretes, com ara consultar bases de dades, executar càlculs, generar gràfiques, accedir a APIs externes, formatar sortides o gestionar fitxers. Aquestes eines permeten delegar funcions específiques i augmentar l'eficiència global del sistema.

KPIs: Mètriques específiques que mesuren l'eficàcia, qualitat i eficiència dels processos de compatibilitat electromagnètica en el desenvolupament, *testing* i certificació de productes electrònics.

Token: Unitat bàsica de processament que utilitzen els models de llenguatge per analitzar i generar text. És la manera com els LLMs "veuen" i processen el llenguatge humà.

Prompt: Instrucció o entrada de text que es proporciona a un model d'IA per obtenir una resposta específica.

Bucket: Contenidor virtual utilitzat per organitzar i emmagatzemar dades, fitxers o recursos de forma agrupada.

Batch: Conjunt de dades, tasques o processos que s'executen junts de forma agrupada i automàtica.

Debug: Procés d'identificar, analitzar i corregir errors en codi, programari o sistemes.

1 Introducció

Durant la meua estada de pràctiques a l'empresa **IDIADA**, vaig formar part de l'equip **DOCS**, especialitzat en l'automatització de tasques mitjançant **Intel·ligència Artificial**. Aquest equip funciona com a centre receptor de peticions procedents de diversos departaments de l'empresa, desenvolupant solucions d'automatització personalitzades per a múltiples casos d'ús.

1.1 Origen del Projecte

El meu Treball de Fi de Grau va sorgir quan se'm va proposar abordar un àmbit relativament inexplorat dins l'empresa: l'automatització avançada en la generació d'informes mitjançant IA. Existia una petició concreta d'un equip que realitza *assajos de Compatibilitat Electromagnètica (EMC)* al laboratori.

Aquest equip actualment extreu els resultats dels seus tests en format CSV i genera informes amb gràfiques utilitzant una eina preexistent. No obstant això, aquesta solució presenta **limitacions significatives** que afecten l'eficiència del procés i la qualitat dels informes resultants. Davant d'aquesta situació, l'equip DOCS va identificar una oportunitat per aplicar tècniques d'Intel·ligència Artificial que poguessin superar aquestes restriccions.

1.2 Problemàtica Identificada

L'eina actualment implementada, tot i complir amb la funció bàsica de generació d'informes, presenta diverses restriccions quant a flexibilitat, personalització i capacitat d'anàlisi. Aquestes limitacions no només alenteixen el procés de documentació, sinó que també impedeixen aprofitar tot el potencial analític de les dades recollides durant els assajos.

En aquest context, l'aplicació d'Intel·ligència Artificial representa una oportunitat per **transformar significativament** aquest procés, superant les barreres existents i afegint capacitats analítiques avançades que actualment són inviàbles amb les eines convencionals.

1.3 Objectius del Projecte

1.3.1 Objectius Específics:

1. Substitució tecnològica:

- Reemplaçar l'eina actual de generació d'informes per una solució basada en IA
- Eliminar les limitacions tècniques identificades en el sistema actual
- Desenvolupar un sistema **escalable** que superi les restriccions de la programació fixa existent

2. Capacitats analítiques avançades:

- Integrar funcionalitats d'anàlisi avançada
- Extreure informació més profunda dels assajos EMC
- Proporcionar valor afegit respecte a la solució existent

3. Optimització del flux de treball:

- Reduir el temps dedicat a tasques repetitives de documentació
- Permetre als enginyers centrar-se en l'anàlisi i interpretació de resultats

- Millorar l'eficiència operativa de l'equip d'assajos EMC

1.3.2 Impacte Estratègic:

- Demostrar el potencial de la IA en l'automatització de processos documentals complexos
- Establir un precedent per a futures aplicacions similars en altres departaments d'IDIADA
- Crear una solució escalable i adaptable que serveixi de base per a futures expansions tecnològiques

2 Descripció general del projecte

2.1 Entorn Empresarial

IDIADA és una empresa líder en serveis d'enginyeria per a la indústria automotriu, especialitzada en proves, homologacions i desenvolupament de vehicles. L'empresa compta amb diverses divisions i equips tècnics que realitzen assajos i proves sota estàndards internacionals, generant una gran quantitat de dades que necessiten ser processades i documentades de manera eficient.

Els assajos de **Compatibilitat Electromagnètica (EMC)** són fonamentals en la indústria automotriu moderna, ja que garanteixen que els sistemes electrònics del vehicle funcionin correctament sense interferències mútues ni amb l'entorn electromagnètic. Amb l'increment dels vehicles elèctrics i la integració massiva de tecnologies digitals, connectivitat i sistemes d'assistència avançada (ADAS), aquests assajos s'han tornat crítics per assegurar la seguretat, fiabilitat i compliment normatiu dels vehicles abans de la seva comercialització.

En aquest context, l'equip DOCS actua com a facilitador de la transformació digital, desenvolupant solucions d'automatització basades en IA per optimitzar processos interns. La meua proposta s'emmarca dins d'aquesta estratègia, centrant-se específicament en l'equip de DARWIN que fa la generació de documents dels assajos de Compatibilitat Electromagnètica (EMC).

2.2 DARWIN

2.2.1 Què és DARWIN?

DARWIN és una aplicació web basada en el núvol dissenyada per a l'anàlisi de dades i generació d'informes per a procediments d'assaig en el sector de l'automoció. Es posiciona com l'eina de referència i transversal a IDIADA per a totes les activitats que requereixen anàlisi de dades i generació d'informes, reemplaçant solucions i aplicacions d'escriptori ja existents en cada àrea.

2.2.2 Funcionalitats principals de DARWIN

DARWIN s'estructura en diversos mòduls que funcionen de manera integrada:

1. **Mòdul d'ingestió de dades:** Actua com un "*listener*" al servidor que detecta nous fitxers d'assaig, els puja al núvol i els converteix a un format universal JSON estandarditzat per a DARWIN.
2. **Mòdul de preprocessat:** Un cop els fitxers d'assaig estan convertits al format universal i localitzats al núvol, s'apliquen estandarditzacions de noms de senyals, correccions necessàries, filtres i offset, per poder analitzar-los segons les especificacions de les normatives. El resultat és un fitxer JSON preprocessat.
3. **Mòdul de processat/anàlisi:** A partir dels fitxers JSON preprocessats, s'analitzen les dades, es generen els gràfics i es calculen els resultats i KPIs necessaris definits en els arxius de configuració. Els resultats es guarden al núvol en format JSON i PNG, i poden ser utilitzats pel mòdul de report de resultats.

4. **Mòdul de report i generació d'informes:** Amb els resultats calculats en el mòdul de processat/anàlisi i la informació procedent de TELMA ¹sobre el vehicle, instrumentació, etc., també disponible al núvol, es generen els entregables del projecte. Es disposa d'un catàleg de plantilles d'informe prèviament definits, on l'usuari pot escollir el tipus de format de sortida per a cada informe.

2.2.3 Arquitectura i metodologia de treball

DARWIN està basat en una arquitectura de microserveis que suporta un llistat de funcionalitats agrupades en els mòduls mencionats. Cada mòdul pot executar-se automàticament amb arxius de configuració.

La metodologia de treball es divideix en fases ben definides:

1. Ingestió de dades
2. Preprocessat
3. Anàlisi/Processat
4. Generació d'informes

Aquest model permet la traçabilitat completa del procés i garanteix la consistència dels resultats.

2.2.4 Beneficis i avantatges de DARWIN

Els principals avantatges que ofereix DARWIN són:

- **Procés automatitzat de generació d'informes**, minimitzant la interacció de l'usuari.
- **Estalvi de costos i temps d'anàlisi** en el projecte.
- **Major eficiència** en el processament de dades.
- **Consistència i qualitat** en les dades i resultats obtinguts.
- Creació d'un **catàleg de dades explotable** per a la presa de decisions d'enginyeria en el desenvolupament del vehicle.
- **Accés global a les dades** des de qualsevol ubicació, per part de les diferents unitats de negoci.

2.2.5 Estandardització i automatització

DARWIN permet l'estandardització dels mètodes i funcionalitats d'anàlisi i l'automatització de la generació d'informes a través de:

- La definició de diccionaris i arxius de configuració de referència per a cada una de les fases i estàndard/normativa de les activitats d'enginyeria en automoció.
- La divisió en fases del procés en diferents passos que permeten la traçabilitat, aplicant arxius de configuració de referència per a l'anàlisi i generació de resultats.

¹ L'aplicació es basa en principis de governança de dades de manera que tota la informació recollida durant els processos d'enginyeria sigui traçable i accessible per a l'anàlisi, la generació d'informes o l'explotació posterior.

- Una llibreria de funcions transversal i parametritzada, que permet fàcilment la configuració, generació i càlcul de gràfics i KPIs de manera dinàmica i adaptable per l'usuari.

2.2.6 Relació amb altres aplicacions

DARWIN s'executa després de la definició d'un projecte i assaigs en TELMA. S'alimenta de la informació de les mostres, assaigs, tasques, etc. definides al projecte de TELMA pels enginyers i tècnics d'automoció.

Després de l'execució de l'anàlisi d'un projecte, DARWIN genera un catàleg de dades que pot ser explotat amb altres eines de comparació, com la DNA. La DNA permet utilitzar les dades analitzades de diferents projectes, posicionant el vehicle, comportament i característiques amb altres vehicles prèviament analitzats, facilitant a l'enginyer la presa de decisions i conclusions sobre els components i característiques en el desenvolupament.

2.2.7 Interfície d'usuari i fluxos de treball

La interfície d'usuari de DARWIN està organitzada en diversos mòduls:

- **Projectes:** Creació i visualització de dades d'un projecte
- **Operacions:** Definició i execució d'anàlisis
- **Catàlegs:** Gestió de configuracions estandarditzades per cada fase del procés
- **Llibreria de funcions:** Accés a les funcionalitats transversals

El flux de treball típic comença amb la creació d'un projecte, la seva vinculació amb TELMA, la definició de les anàlisis a executar, i finalment l'execució automàtica del procés complet fins a l'obtenció dels informes.

2.3 Necessitats Identificades

Es van identificar dues problemàtiques concretes a partir de l'anàlisi de documents finals generats automàticament des de DARWIN. Aquestes dificultats, tot i ser puntuals, generaven ineficiències o errors que resultaven complicats de solucionar durant el procés de generació del document. Per aquest motiu, es va optar per treballar sobre els documents un cop completats, amb l'objectiu de corregir aquestes mancances de manera automatitzada.

Les dues problemàtiques principals a resoldre van ser:

2.3.1 Control de la indexació automàtica

En utilitzar una plantilla comuna per generar documents a partir d'assaigs, es va detectar que la numeració dels apartats no es generava de manera dinàmica. En tots els documents, els índexs es repetien amb el mateix format (1.1.1 i 1.1.1.1), sense tenir en compte la jerarquia real dels continguts. Aquesta manca de variació dificultava la lectura i comprensió del document final, i no permetia distingir correctament entre seccions diferents.

2.3.2 Automatització d'enllaços a la test matrix

Una altra carència detectada es trobava a la secció anomenada *test matrix*. En aquest apartat, s'hi havia d'incloure manualment l'enllaç a la gràfica corresponent a cada prova. Aquest procés, a més de ser laboriós, augmentava el risc d'errors humans i inconsistències.

Per això, se'm va demanar automatitzar aquesta tasca, generant automàticament els enllaços a les gràfiques correctes dins del document final.

A part de les problemàtiques es van identificar millores al sistema ja existent.

2.3.3 Validació i agrupació semàntica de textos mitjançant LLM

En el sistema actual, la gestió i classificació de textos amb nomenclatures diverses presenta dificultats degut a la inconsistència i variabilitat dels termes emprats. Això genera problemes a l'hora d'agrupar informació relacionada i afecta la qualitat de l'anàlisi. Per tant, és necessari implementar un mecanisme capaç de reconèixer i agrupar automàticament fragments textuais amb significat similar, més enllà de simples coincidències literals. La incorporació d'un model de llenguatge gran LLM per a la validació i agrupació semàntica dels textos permetria superar aquestes limitacions, millorant la coherència, precisió i eficiència en el tractament de la informació.

Aquesta tasca amb la programació tradicional és bastant limitant ja que s'haurien de fer bastantes comprovacions, i es limita al cas concret que estiguis tractant.

2.3.4 Substitució per una eina que calculi els pics d'una gràfica mitjançant un LLM²

Una de les millores identificades al sistema existent és la substitució de l'algorisme fix de detecció de pics per un model de llenguatge LLM capaç d'identificar-los de manera semàntica mitjançant un *prompt*. Aquest enfocament permet definir els criteris de detecció en llenguatge natural, fent el sistema molt més flexible i fàcil d'entendre. A més, simplifica la modificació del comportament davant d'errors o canvis, ja que només cal ajustar el text del *prompt*, evitant canvis en el codi. Això suposa una millora important en accessibilitat, i adaptabilitat a diferents contextos.

2.3.5 Anàlisi detallada i comparativa dels resultats dels assajos

Un cop generat l'informe complet de l'assaig, que cobreixi totes les necessitats especificades en el document, es preveu incorporar una funcionalitat addicional enfocada a l'**anàlisi dels assajos**. Aquesta funcionalitat permetrà aprofundir en l'exploració interna dels resultats, identificant els punts o fases on es produeixen errors o fallades en els assajos.

Mitjançant l'ús de gràfiques detallades i la comparació dels diferents resultats obtinguts, es podrà obtenir una visió més precisa i visual de l'evolució dels assajos i dels seus possibles punts dèbils. Aquesta anàlisi avançada facilitarà la detecció de patrons, la localització de causes i la presa de decisions més informades per a la millora del sistema i la fiabilitat dels assajos.

2.3.6 Generació de dades sintètiques

² Sistema d'intel·ligència artificial entrenat amb grans quantitats de text que pot comprendre i generar llenguatge natural de manera coherent i contextual.

2.3.5.1 Context de Confidencialitat

El principal desafiament identificat deriva de la **naturalesa confidencial** de les dades reals d'assajos EMC. Aquestes dades contenen informació propietària de l'empresa i dels clients que no pot ser divulgada fora de l'entorn corporatiu.

2.3.5.2 Restricció per Demostració Acadèmica

Les polítiques de seguretat empresarial i els acords de confidencialitat impedeixen l'ús de dades reals per la **demostració pública del TFG**. Això crea una limitació crítica per presentar i validar el funcionament del sistema en un context acadèmic.

2.3.5.3 Necessitat de Solució

Per realitzar una demostració completa del flux desenvolupat és imprescindible generar dades sintètiques que:

- Permetin l'**execució completa del pipeline** sense comprometre informació sensible
- Mantinguin les **característiques estructurals** de les dades reals
- Preservin el **realisme necessari** per una demostració efectiva del sistema

La generació de dades sintètiques es converteix així en un requisit essencial per conciliar la protecció de la confidencialitat amb la necessitat de validació acadèmica del projecte desenvolupat.

2.4 Tipologia d'assajos, punts de test i modes d'operació

Per tal de garantir un anàlisi correcte i rigorós del comportament dels sistemes electrònics avaluats, és essencial entendre quins són els **tipus d'assaig** que s'estan duent a terme i sobre **quins punts de test (*test points*)** s'apliquen. Aquesta comprensió proporciona el context necessari per interpretar els resultats de forma adequada i assegurar la validesa de les conclusions obtingudes.

En aquest projecte s'han tingut en compte tres tipus principals d'assajos, cadascun amb una funció específica dins del marc de validació electromagnètica del dispositiu:

- **RE – Radiated Emissions:** consisteix en la mesura de les emissions electromagnètiques irradiades pel sistema. Aquestes emissions poden interferir amb altres equips electrònics propers, per la qual cosa cal comprovar que es mantenen dins dels límits normatius establerts.
- **AN – Antenna (Radiated Immunity):** aquest assaig consisteix a exposar el dispositiu a camps electromagnètics generats per una antena per comprovar la seva **immunitat irradiada**, és a dir, com es comporta davant d'interferències externes.
- **CP – Conducted Perturbations:** s'avalua la susceptibilitat del dispositiu a perturbacions conduïdes a través dels cables d'alimentació i senyal. Es pot tractar tant de la injecció com de la mesura de soroll en aquestes línies.

Per cadascun d'aquests assajos, es realitzen proves sobre un conjunt de punts d'accés elèctrics del sistema, anomenats **test points**. Aquests test points representen punts clau dins la xarxa elèctrica del dispositiu (alimentació, alta tensió, senyals de control, etc.), i són on s'apliquen les condicions de prova o es realitzen les mesures corresponents.

Cada combinació de *test point* i assaig s'evalua sota diversos **modes operatius (*operation modes*)**, que representen diferents estats o condicions de funcionament del

sistema. Això permet estudiar el comportament electromagnètic del sistema de manera completa i exhaustiva.

Aquí tenim la taula dels diferents *Test Points* usats:

Test Point	Significat
E_S_KL30S	Kl.30 commutada –Alimentació de bateria commutada (via relé o clau)
E_S_WUP	Wake-Up signal – Senyal per activar el mòdul des de xarxes (CAN, LIN, etc.)
B+	Born positiu de bateria (Kl.30)
B-	Born negatiu o massa de bateria (GND)
HV+	Alta tensió positiva – línia general de HV
HV+ LT2	Alta tensió positiva del bloc de tracció LT2
HV-	Alta tensió negativa (referència de massa de HV)
HV- LT2	Alta tensió negativa del bloc LT2
HV+ LT5	Alta tensió positiva del bloc de tracció LT5
HV- LT5	Alta tensió negativa del bloc LT5

Taula 1. *Test Points*

2.5 Previsions d'Ús

L'eina està dissenyada principalment per ser integrada en el sistema de l'equip DARWIN, facilitant el processament automatitzat dels resultats dels assajos. Les millores implementades estan orientades a optimitzar el flux de treball dels analistes que gestionen aquests resultats, proporcionant funcionalitats avançades d'anàlisi i visualització.

En concret, l'objectiu és que, mitjançant l'anàlisi final dels assajos, es pugui realitzar una **detecció ràpida i precisa de les zones o punts crítics on es produeixen fallades**, sense necessitat d'examinar individualment cada gràfica o dada. Aquesta capacitat d'identificació àgil redueix considerablement el temps d'anàlisi, incrementa l'eficiència en la presa de decisions i millora la fiabilitat dels diagnòstics.

3 Requisits

3.1 Requisits funcionals

- **Lectura de dades:**

El sistema ha de poder llegir dades tècniques en format JSON, provinents del processat de bandes i modes.

- **Detecció de pics:**

El sistema ha de permetre detectar automàticament els punts que compleixen els criteris definits per ser pic, i ordenar-los per marges negatius.

- **Generació de gràfiques:**

El sistema ha de generar gràfiques a partir de les dades numèriques, amb format i estètica predefinitos.

- **Identificació de configuracions equivalents:**

El sistema ha d'identificar configuracions tècnicament equivalents encara que els noms no coincideixin exactament, mitjançant un model de llenguatge.

- **Generació automàtica de documents:**

El sistema ha d'omplir automàticament una plantilla Word amb les dades, gràfiques i resultats corresponents a cada mode.

- **Inserció d'enllaços a la *test matrix*:**

El sistema ha d'inserir automàticament els enllaços a les gràfiques dins la *test matrix* del document.

3.2 Requisits no funcionals

- **Consistència visual:**

Tots els documents generats han de mantenir un format coherent, evitant elements com la notació exponencial i respectant els estils de la plantilla.

- **Compatibilitat:**

Els documents generats han de ser compatibles amb Microsoft Word (versió 2016 o superior).

- **Precisió semàntica en IA:**

En l'ús de LLMs, el sistema ha de validar manualment els resultats per evitar inferències incorrectes, especialment en identificació de configuracions.

- **Seguretat dels recursos:**

Les gràfiques han d'estar correctament enllaçades des del sistema remot (ex. S3) i no contenir dades sensibles.

- **Robustesa:**

El sistema ha de detectar i gestionar errors en el format d'entrada (noms inconsistents, JSON mal format, gràfiques absents).

4 Anàlisi dels requisits funcionals

4.1 Sistema d'indexació automàtica

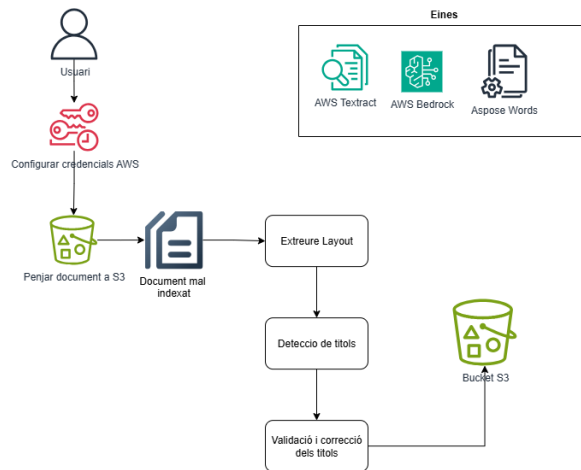


Figura 1. Flux Sistema d'indexació

1. **L'usuari** inicia el procés del sistema.
2. Es configuren les **credencials AWS** necessàries per accedir als serveis.
3. S'utilitza la línia d'**eines** que inclou:
 - *AWS TextTract* (extracció de text)
 - *AWS Bedrock* (IA generativa)
 - *Aspose Words*
4. L'usuari **penja el document** al sistema d'emmagatzematge **S3**.
5. Es processa el **document mal indexat** que necessita millores.
6. Es realitza l'**extracció de títols amb Layout** per obtenir els títols del document.
7. Es procedeix a la **detecció de títols** dins del document.
8. Es connecta amb el **bucket S3** per a l'emmagatzematge persistent.
9. Finalment, es porta a terme la **validació i correcció dels títols** detectats.

4.2 Generació de enllaços a una *Test Matrix*

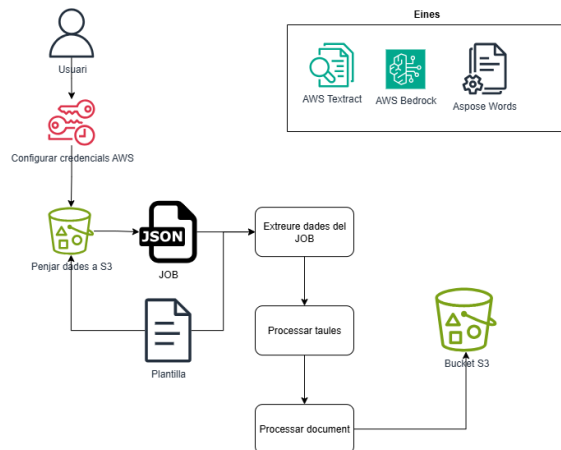


Figura 2. Flux Generació d'enllaços

1. **L'usuari** inicia el procés del sistema.
2. Es **configuren les credencials AWS** necessàries per accedir als serveis.
3. S'utilitza la línia d'eines que inclou:
 - *AWS Textract* (extracció de text)
 - *AWS Bedrock* (IA generativa)
 - *Aspose Words*
4. Es **pengen els arxius a S3** per emmagatzemar els documents a processar.
5. Es genera i processa la informació en format **JSON** amb les dades estructurades.
6. S'utilitza una **plantilla** específica per al tractament dels documents.
7. S'**extreuen les dades del JOB** per obtenir la informació rellevant de la tasca.
8. Es **processen les taules** contingudes dins dels documents per extreure dades tabulars.
9. Es **processa el document** complet per obtenir tota la informació textual i estructural.
10. El resultat final es desena al **bucket S3** per a l'emmagatzematge persistent.

4.3 Generació de un informe

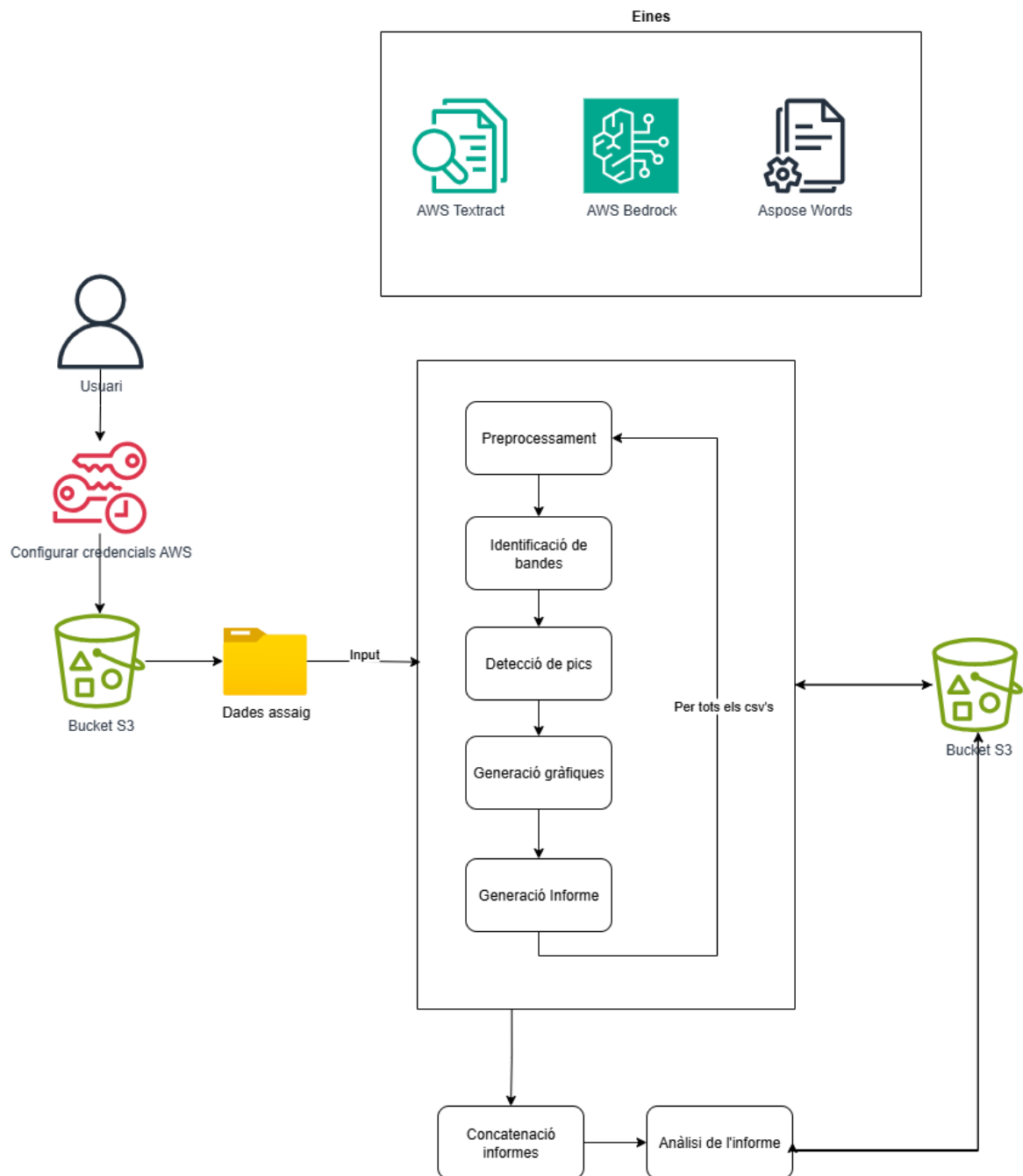


Figura 3. Flux Generació d'informes

1. **L'usuari** inicia el procés del sistema.
2. Es **configuren les credencials AWS** necessàries per accedir als serveis.
3. S'utilitza la línia d'eines que inclou:
 - *AWS TextTract* (extracció de text)
 - *AWS Bedrock* (IA generativa)
 - *Aspose Words*

4. Es **penja el document a S3** per emmagatzemar-lo al servei d'emmagatzematge.
5. Es processa el **document mal indexat** que necessita millores.
6. Es realitza el **preprocessament** inicial del document per preparar-lo.
7. Es porta a terme la **identificació de bandes** per segmentar les diferents seccions.
8. Es fa la **detecció de pics** per trobar valors atípics o punts rellevants.
9. Es processa la **generació de gràfiques** per visualitzar les dades analitzades.
10. Es du a terme la **generació d'informe** amb els resultats obtinguts.
11. Es desa tot al **bucket S3** per a l'emmagatzematge persistent.
12. Es fa la **concatenació d'informes** per unificar tots els resultats.
13. Finalment, es realitza l'**anàlisi de l'informe** per obtenir les conclusions finals.

5 Disseny

5.1 LLM

5.1.1 Definició i Característiques

Els **Grans Models Lingüístics** (*Large Language Models* - LLM) són sistemes d'intel·ligència artificial especialitzats en el processament i generació de llenguatge natural. Es caracteritzen per incorporar centenars de milions a bilions de paràmetres entrenats amb corpus textuals massius. Aquestes eines presenten una escala massiva que varia entre 10^8 i 10^{12} paràmetres entrenables, utilitzen entrenament autoregressiu per predir la següent paraula en una seqüència, desenvolupen capacitats emergents que sorgeixen de l'escala sense programació explícita, i mostren una versatilitat que permet la seva aplicació a múltiples tasques sense necessitat de re-entrenament específic.

5.1.2 Arquitectura Transformer

La base tecnològica dels LLMs moderns és l'arquitectura **Transformer**, desenvolupada per Vaswani et al (2017). Aquesta arquitectura revolucionària supera les limitacions de les xarxes neuronals recurrents mitjançant tres components clau: el mecanisme d'atenció que permet identificar relacions entre elements distants en una seqüència, el processament paral·lel que ofereix la capacitat de processar múltiples *tokens* simultàniament, i l'estructura codificador-descodificador que transforma l'entrada en representacions útils.

Els avantatges principals d'aquesta arquitectura inclouen una eficiència computacional superior a les RNNs³, una capacitat de context a llarg termini que permet mantenir coherència en textos extensos, i una escalabilitat excel·lent amb l'augment de recursos computacionals.

5.1.3 Procés d'Entrenament

L'entrenament dels LLMs es realitza en dues fases diferenciades. La fase de **pre-entrenament** té com a objectiu aprendre representacions generals del llenguatge utilitzant text no estructurat d'Internet com CommonCrawl, Wikipedia i llibres digitals. Aquesta fase emprà metodologies d'aprenentatge auto-supervisat amb tasques de modelatge del llenguatge, requereix recursos intensius com GPU/TPU farms durant setmanes o mesos, i treballa amb escales massives on models com GPT-4 utilitzen trilions de tokens.

La segona fase, l'**ajustament fi** o *fine-tuning*, se centra en l'especialització per tasques específiques. Aquesta fase utilitza tècniques com el *Supervised Fine-tuning* (SFT) amb dades etiquetades i el *Reinforcement Learning from Human Feedback* (RLHF) per optimitzar segons preferències humanes. Les dades utilitzades són conjunts curats i

³ Les RNNs són un tipus d'arquitectura de xarxa neuronal dissenyada per processar **dades seqüencials** com text, àudio o sèries temporals. La seva característica distintiva és que tenen **memòria interna** que els permet recordar informació de passos anteriors en la seqüència.

específics del domini, i la durada és significativament menor, requerint dies o setmanes en contrast amb els mesos necessaris per al pre-entrenament.

5.1.4 Capacitats i Aplicacions

Els LLMs demostren capacitats excepcionals en tasques de processament de llenguatge natural. En generació de text, poden crear articles, codi i poesia amb coherència i creativitat notables. La seva comprensió lectora inclou anàlisi semàntica i inferència contextual avançades. En traducció automàtica, ofereixen qualitat propera a traductors humans professionals, mentre que en resums poden extreure i abstraure informació clau de manera eficient. També excel·leixen en classificació de text per anàlisi de sentiment i detecció de contingut problemàtic.

Les **capacitats emergents** representen un aspecte fascinant d'aquests models. Inclouen el *few-shot learning* per aprendre amb pocs exemples, el *chain-of-thought reasoning* per raonament pas a pas, la generació de codi en múltiples llenguatges de programació, i el raonament matemàtic per resoldre problemes aritmètics complexos.

La **multi modalitat** constitueix una frontera avançada on els LLMs moderns integren múltiples modalitats sensorials. Els models visió-llenguatge com GPT-4V processen simultàniament imatges i text, els sistemes àudio-llenguatge realitzen transcripció i generació de veu, i eines com DALL-E combinen comprensió textual amb creació visual sofisticada.

5.1.5 Exemples Destacats

5.1.5.1 Models Propietaris

Model	Empresa	Característiques Clau
GPT-4	OpenAI	Capacitats multimodals, rendiment superior
Claude	Anthropic	Context fins a 200k tokens, enfocat en seguretat
PaLM 2	Google	Integració amb ecosistema Google
GPT-3.5	OpenAI	Optimitzat per velocitat i eficiència

Taula 2. Alguns models

En el context de l'empresa i del meu treball s'usa Claude ja que manté la seguretat i eficiència

5.1.6 Avantatges

Els avantatges tècnics dels LLMs inclouen la seva versatilitat per funcionar com un model únic capaç de realitzar múltiples tasques, l'aprofitament del *transfer learning* per

utilitzar coneixement pre-entrenat, l'escalabilitat predictable que millora amb més paràmetres i dades, i la contextualització automàtica al domini d'aplicació.

Des d'una perspectiva empresarial, ofereixen automatització de tasques cognitives complexes, personalització de contingut i respostes segons necessitats específiques, acceleració significativa del desenvolupament d'aplicacions, i reducció substancial de costos en processos tradicionalment manuals.

5.1.7 Limitacions i Reptes

Els **reptes tècnics** inclouen les al·lucinacions o generació d'informació falsa amb aparença versemblant, inconsistència en les respostes per a la mateixa pregunta formulada diverses vegades, limitacions de context degut a la finestra d'atenció finita que varia entre 2k i 200k *tokens*, i coneixement estàtic limitat a la data d'entrenament sense actualització en temps real.

Els **reptes ètics** comprenen la reproducció de biaixos presents en les dades d'entrenament, el potencial per generar desinformació convincent, riscos de privacitat per possible memorització de dades sensibles, i problemes de transparència que dificulten l'explicació de les decisions del model. A més, sorgeixen preocupacions sobre la propietat intel·lectual quan els models generen contingut similar a material protegit per drets d'autor, la necessitat d'establir responsabilitat legal per decisions automatitzades errònies, i la manca de consentiment dels autors originals les obres dels quals van ser utilitzades per entrenar els models. També existeixen dilemes sobre l'autonomia humana quan la dependència excessiva d'aquests sistemes pot reduir les capacitats crítiques i creatives de les persones, així com qüestions de justícia distributiva respecte a qui té accés a aquestes tecnologies i qui en rep els beneficis.

Les **consideracions ambientals** són significatives, amb l'entrenament de GPT-3 consumint aproximadament 1,287 MWh d'energia, emissions substancials de carboni per entrenament i inferència, i dependència de hardware especialitzat extremadament car.

Els **impactes socials** inclouen el potencial desplaçament laboral en treballs basats en processament de text, la concentració de poder en grans corporacions tecnològiques, i problemes d'accessibilitat degut als costos elevats que limiten l'accés democràtic a aquesta tecnologia.

5.1.8 Tendències Futures

Els desenvolupaments tècnics futurs se centren en l'eficiència per crear models més petits amb rendiment equivalent, l'especialització amb LLMs dissenyats per dominis específics com medicina i dret, la multi modalitat avançada amb integració de més modalitats sensorials, i el desenvolupament d'agents autònoms capaços d'executar accions complexes.

Les direccions de recerca prioritàries inclouen la interpretabilitat per comprendre millor el funcionament intern dels models, la seguretat centrada en l'alineació amb valors humans i prevenció de riscos, la personalització que respecti la privacitat individual, i l'eficiència computacional per reduir els recursos necessaris per l'entrenament i operació d'aquests sistemes.

5.2 Disseny de l'Agent

En aquesta secció es descriu la metodologia adoptada per al disseny i desenvolupament d'un agent basat en models de llenguatge grans (LLMs), emprant la llibreria pròpia de l'equip, **agentic-core**. Aquesta llibreria facilita la interacció amb els LLMs i la coordinació de funcions específiques, optimitzant i estandarditzant les crides al model.

5.2.1 Llibreria Agentic-Core

Prèviament, les crides als models LLM es realitzaven de manera manual, la qual cosa implicava un procés complex i poc eficient. La llibreria **agentic-core** proporciona una capa d'abstracció que automatitza i simplifica aquestes interaccions, permetent una gestió més àgil i estructurada dels prompts i de les eines associades.

5.2.2 Components Fonamentals de l'Agent

Per a la creació d'un agent, és necessari definir diversos elements claus:

- **System-prompt:** Defineix l'orientació general, les funcions i les restriccions de l'agent, establint el marc de treball dins del qual opera.
- **Prompt d'instrucció:** Conté les indicacions específiques que dirigeixen l'execució de la tasca assignada.
- **Configuració del model:** A més de seleccionar el model LLM a utilitzar —en aquest projecte s'ha optat pel model **Claude Sonnet** a través d'Amazon Bedrock—, és fonamental ajustar paràmetres clau que controlen el comportament del model, entre els quals destaquen:
 - **max_tokens:** nombre màxim de tokens generats per resposta per controlar la seva longitud.
 - **temperature:** paràmetre que regula la creativitat o aleatorietat en les respostes; valors baixos afavoreixen coherència i determinisme, mentre que valors alts aporten més variabilitat.
 - **top_p:** llindar de probabilitat per al nucleus sampling, que afina l'equilibri entre diversitat i coherència.
 - **frequency_penalty** i **presence_penalty:** per evitar repeticions innecessàries i millorar la naturalitat de les respostes.
 - **stop_sequences:** seqüències que determinen la finalització de la resposta per evitar contingut no desitjat.

Aquests paràmetres es configuren i ajusten segons la tasca concreta i les necessitats de qualitat i estil, permetent una gran flexibilitat i adaptabilitat de l'agent.

- **Eines (tools):** Funcions o mòduls externs que l'agent pot invocar per dur a terme operacions específiques, com ara la generació de gràfiques o la formatació precisa de les respostes.

5.2.3 Gestió i Formatatge de Respostes

Les respostes obtingudes dels LLM solen ser estructures complexes en format JSON, que inclouen múltiples camps informatius. En molts casos, només una part d'aquesta informació és rellevant per a la fase següent del procés.

Per exemple, en un escenari on l'agent ha de realitzar un càlcul, la resposta textual proporcionada pel model podria ser una frase explicativa que conté el resultat, però el que realment es necessita és extreure únicament el valor numèric resultant. Per abordar aquesta necessitat, s'utilitzen classes de formatatge específiques que defineixen l'estructura desitjada de la resposta. Aquestes classes permeten a l'agent retornar només els elements necessaris, en el format adequat, facilitant així el processament i l'anàlisi posteriors.

5.2.4 Configuració i Execució Modular

La configuració final de l'agent implica la definició clara de les eines disponibles i l'ordre d'ús d'aquestes, si s'escau. El *prompt* conté les instruccions per activar aquestes eines en els moments pertinents, i el sistema gestiona la seqüència d'execució i la integració dels resultats.

Aquesta arquitectura modular garanteix flexibilitat, escalabilitat i facilita l'adaptació del sistema a diferents casos d'ús i requeriments futurs, sense necessitat de reformulacions significatives.

5.3 Arquitectura basada en AWS

El sistema està dissenyat seguint una arquitectura basada completament en els serveis d'AWS (Amazon Web Services), aprofitant les seves capacitats per a l'emmagatzematge, processament i gestió de dades.

Per accedir als recursos d'AWS, el sistema requereix una configuració prèvia de **credencials d'accés** (claus d'usuari i secrets) que permeten autoritzar les diferents operacions. Aquestes credencials s'han de configurar principalment a l'inici del sistema o entorn de desplegament, garantint així un accés segur als serveis.

En concret, per a la **persistència de les dades** s'utilitzen **buckets S3**, que són espais d'emmagatzematge en el núvol proporcionats per AWS. L'usuari ha de carregar les dades dels assajos dins d'una carpeta específica en un *bucket* S3.

Un cop les dades d'entrada estan disponibles a S3, el sistema accedeix a aquesta carpeta utilitzant el nom del *bucket* i el nom de la carpeta proporcionats. A partir d'aquí, tot el procés següent es realitza sobre aquesta mateixa ubicació:

- **Preprocessament:** Preparació i neteja de les dades per al seu correcte processament, incloent l'ús d'*AWS Textract* per extreure informació de documents escanejats o formats no estructurats.
- **Processament:** Execució dels càlculs i anàlisis, com la detecció de pics i la identificació de configuracions, fent servir també models avançats basats en AWS *Bedrock* per a tasques d'intel·ligència artificial i processament del llenguatge natural.
- **Generació de gràfiques:** Creació de les representacions visuals dels resultats.
- **Generació d'informes:** Ompliment de les plantilles Word amb dades i gràfiques per crear documents finals, utilitzant la llibreria **Aspose.Words**, que ofereix un maneig robust i flexible dels documents Word. Per a l'ús correcte d'aquesta llibreria, cal configurar la **licència Aspose** al sistema.

Totes les dades i resultats generats (gràfiques, documents parcials, informes finals) es guarden de nou a la mateixa carpeta dins del *bucket* S3, mantenint així tota la informació centralitzada i accessible.

Finalment, el sistema concatena els documents parcials per obtenir l'informe complet i realitza una anàlisi final, que també es penja al *bucket* S3. A més, es descarrega una còpia local del document final per facilitar la seva consulta i distribució.

Aquesta arquitectura permet un flux de treball **centralitzat, escalable i segur**, aprofitant la infraestructura robusta d'AWS i facilitant la gestió automatitzada dels assajos i informes.

6 Implementació

6.1 Tecnologies usades

6.1.1 Amazon Textract

Amazon Textract és un servei d'Intel·ligència Artificial (IA) completament gestionat d'Amazon Web Services (AWS) que utilitza tecnologies d'aprenentatge automàtic per extreure text, dades manuscrites i dades estructurades de documents escanejats i imatges.

6.1.1.1 Funcionalitats Principals

- **Extracció de dades estructurades** de formularis, taules i documents complexos
- **Detecció automàtica de camps** i associació amb els seus valors corresponents
- **Processament multi-format** compatible amb PDF, PNG, JPEG, TIFF
- **Anàlisi de disposició** per comprendre l'estructura del document

6.1.2 Amazon Bedrock

Amazon Bedrock és una plataforma de serveis d'Intel·ligència Artificial generativa que ofereix accés a models de llenguatge gran (LLM) de primer nivell mitjançant APIs senzilles, eliminant la complexitat de gestionar infraestructura d'IA.

6.1.2.1 Models Disponibles i Característiques

- **Claude 3.5 Sonnet (Anthropic)** - Model utilitzat en aquest projecte, destacat per la seva capacitat d'anàlisi i raonament
- **Llama (Meta)** - Model open-source amb bones prestacions
- **Titan (Amazon)** - Optimitzat per tasques empresarials

6.1.2.2 Funcionalitats Principals

- **Processament de llenguatge natural** avançat
- **Anàlisi contextual** de documents tècnics
- **Classificació intel·ligent** de contingut
- **Generació de metadades** descriptives
- **Interpretació de patrons** en dades complexes

6.1.3 Aspose.Words

Aspose.Words és una biblioteca de processament de documents comercial d'alta qualitat que permet crear, modificar, convertir i renderitzar documents de Microsoft Word de manera programàtica, sense requerir la instal·lació de Microsoft Office.

L'elecció de **Aspose.Words** com a motor de generació no és casual: aquesta llibreria ofereix un conjunt de funcionalitats avançades que han resultat crucials per satisfer els requeriments del projecte:

- **Control precís sobre el format i estructura del document:** permet inserir elements complexos (taules, imatges, seccions) amb formatació detallada.
- **Support per a plantilles dinàmiques:** és possible duplicar seccions o blocs de contingut, aplicant les dades de cada configuració sense perdre l'estil original.
- **Gestió robusta de fonts, estils i disseny:** garanteix que el document manté una presentació uniforme, independentment del nombre d'elements inserits o del contingut variable.

6.1.3.1 Aplicació en el Projecte

Aspose.Words constitueix la fase final del pipeline de processament, responsable de la generació automàtica de documents EMC finals amb qualitat professional. Utilitza plantilles predefinides amb marcadors específics que es substitueixen dinàmicament amb dades de cada test, afegint gràfics, taules de resultats i metadades de manera automàtica.

6.1.3.2 Capacitats Destacades

- **Fidelitat de format** equivalent a Microsoft Word natiu
- **Control granular** sobre disseny i presentació
- **Automatització completa** del procés de generació documental
- **Support per elements complexos** com taules dinàmiques i gràfics integrats
- **Optimització de rendiment** per processos *batch* de gran volum

6.2 Desenvolupament del Sistema d'Indexació Automàtica

Hem de tenir en compte que l'input d'aquesta tasca es un document mal indexat, es el document final ja omplert.

6.2.1 Extracció de dades

En la fase inicial del projecte, es va abordar el primer objectiu principal: la implementació d'un sistema per generar **índexs dinàmics** de manera automatitzada, especialment en documents amb una estructura de títols inconsistent.

Després d'una exhaustiva investigació i anàlisi de les diferents tecnologies disponibles, es va identificar *Amazon Textract* com la solució més adequada per a les necessitats del projecte. Aquesta eina destaca per les seves capacitats avançades en el processament de documents i l'extracció intel·ligent de text.

La metodologia inicial contemplava l'extracció completa del text mitjançant una crida bàsica a Amazon Textract, seguida d'un procés de neteja i posterior processament mitjançant un *Model de Llenguatge Gran LLM*. No obstant això, aquesta aproximació resultava excessivament complexa i poc eficient en termes de recursos computacionals.

Durant la fase d'investigació, es va descobrir una funcionalitat específica d'Amazon Textract denominada "*Layout*", que ofereix capacitats avançades per a l'anàlisi estructural de documents. Aquest descobriment tècnic va conduir a una recerca més profunda en la documentació tècnica d'AWS Textract, centrant-se específicament en les capacitats d'extracció de jerarquies de títols i subtítols.

La solució òptima es va trobar en la funcionalitat *Amazon Textract – Layout*, especialment dissenyada per a l'extracció selectiva de títols i subtítols en documents. Aquesta descoberta es va realitzar a través de la documentació oficial d'Amazon, on es detallaven les especificacions tècniques i els protocols d'implementació. Després d'analitzar i adaptar els exemples de codi proporcionats, es va aconseguir implementar amb èxit la funcionalitat d'extracció selectiva de títols, complint així amb els requeriments inicials del projecte.

Amb aquesta funcionalitat, es va poder centrar l'extracció només en els títols, evitant l'extracció íntegra de tot el text del document i posterior filtratge mitjançant LLM, fet que va suposar una reducció significativa en la complexitat i en els recursos necessaris, aconseguint un sistema de baix codi (low-code/no-code).

```
def extractor_layout(input_document: str):
    textract_json = call_textract(input_document=input_document,
                                 features=[Textract_Features.LAYOUT,
                                           Textract_Features.TABLES])
    layout = get_text_from_layout_json(textract_json=textract_json,
                                       exclude_figure_text=True,
                                       exclude_page_footer=True,
                                       exclude_page_number=True,
                                       )
    return layout
```

6.2.1.1 Visualització i Anàlisi del Procés d'Extracció

Per validar i analitzar el funcionament del sistema d'extracció de títols, es va implementar una metodologia de visualització que permet examinar el processament a nivell de pàgina individual. Aquesta funcionalitat es va aconseguir mitjançant una única instrucció de codi:

```
document.pages[2].page_layout.headers.visualize()
```

L'execució d'aquesta instrucció genera una representació visual que il·lustra el procés d'identificació i classificació dels elements de títol. El sistema utilitza un esquema de marcatge diferencial que permet observar:

- **Identificació d'Elements:**
 - El sistema detecta i marca els elements classificats com a LAYOUT_HEADER
 - La funcionalitat és extensible a altres tipus de títols segons els paràmetres especificats
- **Representació Visual Dual:**
 - *Zona Blava:* Indica l'àrea completa identificada com a capçalera
 - *Zona Verda:* Delimita les regions específiques que generen salts de línia (n) en el JSON de sortida
- **Estructura de Dades:**

- El resultat es serialitza en format JSON
- L'estructura segueix un patró clau-valor on:
 - Clau: Número de pàgina
 - Valor: Col·lecció de títols identificats en la pàgina corresponent

Cal destacar que durant l'anàlisi es va observar que el sistema identifica elements com el logotip com a LAYOUT_HEADER. Aquesta particularitat ha estat documentada per a la seva posterior optimització en fases següents del desenvolupament.

Aquesta metodologia de visualització ha resultat especialment útil per a la validació i refinament del procés d'extracció, proporcionant una eina valuosa per a l'anàlisi detallada del comportament del sistema.

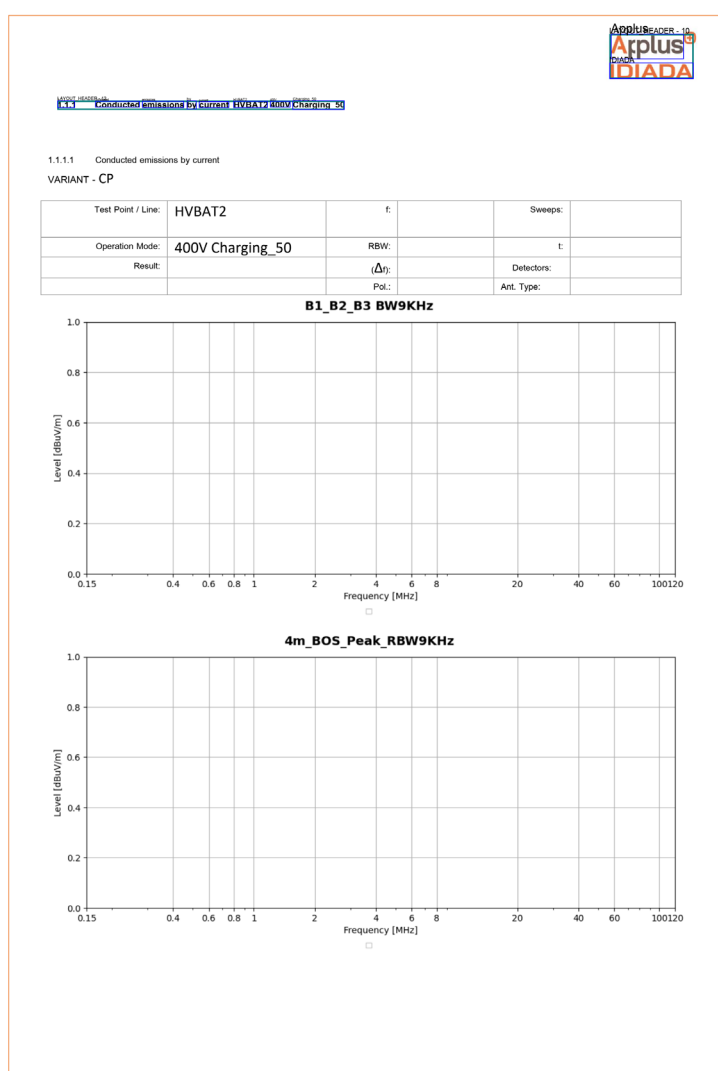


Figura 4. Extracció de títols

6.2.2 *Detecció de títols*

La fase de detecció de títols és essencial per assegurar una estructura clara i coherent del document, especialment en documents amb formats complexos o inconsistents. Aquesta estructura permet no només una presentació ordenada, sinó també una navegació eficient i una posterior generació automàtica d'índexs.

- **Processament mitjançant un agent basat en LLM**

La detecció i numeració dels títols es realitza mitjançant un agent, que utilitza un LLM per interpretar el text. Aquest agent identifica els títols basant-se en l'estructura i format característics dels elements que constitueixen títols, com ara la mida i l'estil de la font, la posició dins de la pàgina i altres atributs de formatació. A més, l'agent descarta elements que poden formar part de la capçalera, com ara logos o altres gràfics, per evitar falsos positius en la detecció de títols. Un cop identificats, s'assigna una numeració seqüencial i jerarquizada (1, 1.1, 1.1.1, etc.) que reflecteix la jerarquia del document.

- **Eina de formatació i estructuració**

L'agent té accés a una eina que transforma la resposta en un format JSON estandarditzat, facilitant la seva posterior integració i manipulació. Aquest JSON conté:

- **Text del títol:** El text complet del títol amb la numeració corresponent.
- **Pàgina:** La ubicació exacta dins del document per facilitar la referència i la navegació.
- **Percentatge de confiança:** Una mètrica que indica la seguretat de l'agent respecte a la identificació del text com a títol.

Aquest mètode, a més de garantir que la jerarquia sigui consistent i fàcilment interpretable, permet superar problemes habituals en documents on l'estructura visual o el formatatge no segueixen pautes estrictes. Així, es facilita la generació automàtica d'índexs, la navegació per seccions i la posterior anàlisi del contingut.

En el context del projecte, aquesta funcionalitat és clau per integrar-se en el flux de treball de tractament de documents, especialment en àmbits on es requereix agilitat i precisió en la gestió de grans volums d'informació amb estructura variable.

6.2.3 *Correcció de numeració al document*

En aquesta etapa s'aborda la verificació i correcció automàtica de la numeració dels títols prèviament identificats i estructurats. L'objectiu és assegurar que tots els títols del document segueixin una jerarquia coherent i actualitzada, segons la informació extreta i validada en l'apartat anterior.

El procés es desenvolupa en les següents etapes:

Cerca de coincidències:

Es realitza una exploració completa del document per identificar els títols que coincideixen amb els enregistrats al JSON generat en l'etapa anterior. Aquesta comparació es fa tenint en compte únicament el contingut textual del títol (ignorant la numeració inicial), per tal de detectar coincidències encara que la numeració original sigui incorrecta.

Anàlisi de la numeració:

Un cop localitzats els títols, es verifica si la numeració que apareix en el document coincideix amb la numeració jeràrquica correcta proporcionada pel JSON.

- Si la numeració és correcta, es manté sense modificacions.
- Si es detecta alguna discrepància, es prepara per fer la correcció automàtica.

Procés de correcció:

La substitució de la numeració incorrecta es fa utilitzant la biblioteca **Aspose.Words**, que permet modificar el contingut del document de manera programàtica sense afectar-ne el format. Aquesta eina ofereix també una funcionalitat per actualitzar l'índex del document de manera automàtica, evitant així haver de realitzar aquest pas manualment després dels canvis.

Aquest procés garanteix una validació del contingut basada en l'extracció semàntica de text. Només es modifica la numeració si realment es detecta una diferència amb el valor correcte, de manera que es manté intacte tant el contingut com l'estructura del document. Finalment, s'actualitza l'índex per reflectir els canvis realitzats i assegurar la coherència global del document.

6.3 Generació de enllaços a una *Test Matrix*

Aquest apartat descriu el procés de generació automàtica d'enllaços (enllaç) dins la *Test Matrix* del document. La finalitat principal d'aquesta funcionalitat és **eliminar la necessitat de generar manualment els enllaços**, un procés feixuc i propens a errors que es realitzava manualment fins ara. Amb la nova metodologia, s'automatitza completament la creació d'enllaços interns que connecten cada entrada de test amb la secció corresponent del document, millorant l'eficiència, la precisió i la traçabilitat dels resultats.

El sistema implementat es basa en quatre etapes principals:

6.3.1 Extracció d'informació del job⁴

El procés comença amb el *job* que conté totes les entrades de test prèviament processades. Aquest fitxer inclou informació rellevant com:

- El *test point*
- El *operation mode*
- El nom del tipus d'assaig

Aquesta informació es carrega en una estructura de dades que permet operar de forma iterativa sobre cada entrada de test.

⁴ Job: JSON amb les dades corresponent a aquell assaig, conté el nom dels modes, test points, el tipus d'assaig i d'altres

6.3.2 Manipulació del document Word amb *Aspose.Words*

Un cop carregades les dades, es processa el document mitjançant la biblioteca **Aspose.Words**, que permet una manipulació avançada de documents Word de forma programàtica. Concretament, es localitza la taula corresponent a la *Test Matrix*, identificada mitjançant patrons predefinitos com l'encapçalament o la seva estructura de columnes.

Aquest accés estructurat permet navegar de manera eficient per les cel·les de la taula i modificar-ne el contingut.

6.3.3 Generació programàtica dels enllaços

Per a cada entrada, es genera un **enllaç intern** que apunta al marcador ubicat a la secció del document corresponent al test. Trobem el marcador mitjançant el nom del assaig, *el Test Point* i el Mode. El procés segueix els passos següents:

- Verificació prèvia de l'existència del marcador
- Construcció del text de l'enllaç que ha de coincidir amb la puntuació del títol
- Assignació del marcador com a destí de l'enllaç

Aquest pas assegura que l'enllaç creat sigui vàlid i funcional dins del context del document.

6.3.4 Inserció i validació dels enllaços

Un cop generats, els enllaços es col·loquen dins la cel·la corresponent de la *Test Matrix* anomenada enllaços. Durant la inserció:

- Es comprova que la cel·la no contingui contingut conflictiu
- Es valida el correcte funcionament de l'enllaç
- S'assegura que el format del document es manté intacte

Si no es troba un marcador vàlid per una entrada concreta, es descarta la inserció i es registra l'excepció per revisió manual posterior.

Aquest sistema automatitzat **substitueix completament el procés manual anterior**, estalviant temps, minimitzant errors humans i garantint la coherència i integritat del document final. A més, assegura una millor navegació i traçabilitat, especialment útil en entorns on es manipulen grans volums de proves i informes.

6.4 Implementació de la detecció de pics mitjançant un Agent

Aquest apartat descriu el procés de detecció automàtica de pics en conjunts de dades utilitzant un **agent basat en un model de llenguatge (LLM)**, com a alternativa a la detecció clàssica mitjançant programació estructurada. L'objectiu principal d'aquest enfocament és valorar el potencial dels models LLM per dur a terme anàlisis quantitatives simples mitjançant instruccions semàntiques, eliminant la necessitat d'implementar lògica condicional o estructures de càlcul tradicionals.

6.4.1 Detecció tradicional vs detecció amb LLM

La detecció de *pics* en sèries temporals és una tècnica habitual per identificar punts que representen canvis destacables o màxims relatius. Tradicionalment, aquesta anàlisi es duia a terme mitjançant algoritmes programats que treballaven sobre els valors numèrics de la sèrie.

6.4.1.1 Detecció tradicional

En la metodologia tradicional, un *pic* es defineix com un punt que presenta una **alça o davallada significativa** respecte als seus valors veïns. Aquest patró es pot detectar mitjançant:

- **Gradients o derivades:** es calcula la diferència entre punts consecutius; un pic es detecta quan es produeix un canvi de signe en el pendent (positiu a negatiu o viceversa).
- **Comparació amb la mitjana local:** un valor es considera pic si se separa clarament de la mitjana d'una finestra de punts propers.
- **Lindars mínims:** s'estableix una diferència mínima respecte als veïns perquè un punt sigui considerat rellevant.
- **Ordenació per intensitat:** es seleccionen els N pics amb major intensitat (positiva o negativa).

Aquest enfocament és efectiu, però requereix configurar paràmetres concrets (mida de finestra, toleràncies, etc.) i adaptar el codi segons la naturalesa de les dades. A més, no permet capturar definicions més qualitatives de què és un *pic* si no s'especifiquen de forma explícita.

6.4.1.2 Detecció mitjançant un Agent

Amb l'ús d'un Agent, es proposa una alternativa en què la detecció de pics no es basa en càlculs matemàtics, sinó en una **descripció textual i semàntica** del concepte. En aquest cas, s'utilitza un *prompt* que conté una definició com la següent:

“Un *pic* és un punt dins una sèrie de valors numèrics que mostra una forma característica: un augment sobtat seguit d'un descens, o un descens seguit d'un augment. També es considera la magnitud del canvi respecte als punts adjacents. Els *pics negatius* són aquells amb un valor de *margin* negatiu. Els més rellevants són els que tenen el *margin* més negatiu.”

Aquesta descripció permet que el model **interpreti la forma global del comportament temporal** i seleccioni aquells punts que s'ajusten millor a la definició proporcionada.

A diferència de l'enfocament tradicional, aquest mètode no requereix paràmetres numèrics explícits ni programació condicional, sinó una formulació clara i ben estructurada del comportament esperat. Això aporta flexibilitat i permet adaptar el criteri simplement modificant el llenguatge del *prompt*.

6.4.2 Definició del prompt per a l'Agent

Es va definir un *prompt* específic amb les següents instruccions:

- Descripció de la forma característica d'un pic (augment sobtat seguit de descens, o viceversa)

- Criteri numèric explícit: considerar només els punts amb *margin* negatiu
- Ordenació per importància: seleccionar els N punts amb valor més negatiu
- Una *tool* amb el format de sortida estructurat, que pot ser posteriorment validat o processat.

6.4.3 Execució del model i generació dels resultats

Un cop inicialitzat l'agent amb el *prompt* adequat i la configuració del model, el sistema processa la sèrie de dades rebuda i retorna una llista de punts identificats com a pics. Cada resultat inclou:

- Identificador del pic
- Valor numèric del *margin*
- Valor numèric del *Level*
- Valor numèric de la freqüència
- Classificació segons la seva intensitat, ordre descendent
- Format de resposta validat mitjançant una classe de format que assegura l'estructura JSON

Els resultats obtinguts mitjançant el model van ser comparats amb els generats pel sistema original i van coincidir de manera exacta els resultats.

És fonamental que els resultats obtinguts en el procés de detecció de pics siguin exactament iguals als generats per la lògica tradicional, ja que aquests resultats tenen un ús crític: formen part dels informes tècnics que s'utilitzen en processos d'homologació de vehicles.

Les homologacions són procediments regulats i estrictes que exigeixen una alta precisió i traçabilitat en les dades presentades. Qualsevol desviació, per mínima que sigui, pot comportar la no acceptació de l'informe per part de les autoritats competents o inclús problemes legals i tècnics en la comercialització del vehicle.

Per aquest motiu, la comparació entre el sistema original i el sistema basat en LLM no és només una prova qualitativa, sinó una validació crítica que garanteix que el model de llenguatge pot substituir de manera fiable el mètode tradicional, sempre i quan es mantingui la coherència i exactitud dels resultats.

En resum, no es tracta només de trobar els mateixos pics, sinó de garantir que el nou sistema és robust, reproduïble i acceptat en entorns regulats, especialment en aplicacions com l'automoció, on les dades tècniques tenen un impacte directe en la seguretat i la conformitat legal del producte final.

6.4.4 Validació i control de qualitat

Tot i la qualitat dels resultats, es van tenir en compte diversos aspectes crítics:

- Els LLM's **no realitzen càlculs matemàtics reals**, sinó que **interpreten el comportament esperat** segons el *prompt*.
- La **formulació precisa del *prompt*** és essencial per evitar ambigüitats o resultats incorrectes

- Es va implementar un sistema de **validació creuada**, comparant la sortida de l'agent amb la sortida del mètode tradicional.

Aquest enfocament híbrid garanteix la fiabilitat de la detecció, mantenint la flexibilitat d'ús dels models de llenguatge, però amb control estricte de la qualitat.

6.5 Identificació de configuracions mitjançant un Agent

Un dels reptes detectats durant el projecte va ser la identificació automàtica de configuracions tècniques que, tot i referir-se al mateix concepte, es trobaven escrites de formes diferents segons la font. Aquest fet dificultava la classificació i agrupació consistent d'informació relacionada. Les diferències incloïen l'ordre dels termes, l'ús d'abreviacions, o variacions menors en valors com la freqüència.

Per resoldre aquesta situació, es va implementar un **Agent basat en LLM** dissenyat per **detectar similituds semàntiques** entre configuracions. El funcionament consisteix a:

- Proporcionar al model una **configuració de referència**.
- Passar-li un conjunt de **configuracions candidates** provinents de fonts diverses.
- Donar-li un **prompt** específic que indica al model com determinar quines configuracions poden considerar-se equivalents, tenint en compte possibles diferències en l'estructura textual, l'ordre dels elements o petites variacions de valor.

Aquest enfocament permet al model identificar correctament coincidències com ara:

- Band_4_1800MHz
- 1800_B4
- B4-Freq1800

com a representacions equivalents d'una mateixa entitat tècnica.

L'ús d'un model de llenguatge en aquest cas ha demostrat ser una alternativa eficient a la comparació tradicional de cadenes, ja que ofereix:

- **Major flexibilitat** davant formats diversos.
- **Reducció de la necessitat de regles específiques**.
- **Capacitat d'adaptació** a nous casos sense modificar el codi, només ajustant el *prompt*.

Gràcies a aquesta solució, es va aconseguir millorar de manera notable la robustesa i fiabilitat del sistema en la gestió de configuracions tècniques heterogènies.

6.6 Generació de gràfiques personalitzades amb programació tradicional

Un dels components essencials del projecte ha estat la generació de gràfiques a partir de dades experimentals o de simulació. Aquestes gràfiques formen part de la documentació tècnica final que es lliura com a resultat dels assajos, i per tant, han de complir criteris molt estrictes tant pel que fa a l'estètica com a la presentació de la informació. A diferència d'altres funcionalitats que han estat delegades a models de llenguatge o a agents intel·ligents, en aquest cas es va optar deliberadament per una implementació mitjançant programació tradicional.

Tot i que els LLM tenen la capacitat de generar codi dinàmic per crear gràfiques (per exemple, en llenguatges com Python amb biblioteques com Matplotlib, Seaborn o Plotly), aquesta aproximació no era adequada per a aquest cas concret, ja que la sortida d'aquests models pot variar lleugerament d'una execució a una altra, especialment si el *prompt* no està absolutament determinat o si el context canvia. Aquesta variabilitat, tot i ser acceptable en processos més exploratoris, és **incompatible amb la necessitat de produir gràfics totalment estables i homogenis en un entorn de producció formal**.

6.6.1 Requisits específics de les gràfiques

Les gràfiques generades han de respectar una sèrie de requisits estrictes, que inclouen:

- **Criteris visuals predefinits:**
 - Paleta de colors coherent amb l'estil corporatiu o tècnic.
 - Tipografies específiques per als textos, etiquetes i llegendes.
 - Estils de línia i punts diferenciats per tipus de dades (simulació, mesura real, límits d'error, etc.).
- **Format numèric dels eixos:**
 - Evitació explícita de notació exponencial en valors petits (ex: 1000 en lloc de $1e3$).
 - Control precís sobre la granularitat de les divisions dels eixos (*ticks*), tant horitzontals com verticals.
- **Dimensió i resolució controlades:**
 - Sortida a mida fixa en píxels per garantir integració directa en documents, sense requerir redimensionament.
 - Resolució mínima per a impressió en PDF d'alta qualitat (ex. 300 DPI o superior).
- **Estabilitat i predictibilitat de la sortida:**
 - Les gràfiques han de ser 100% reproduïbles, és a dir, idèntiques entre execucions amb les mateixes dades.
 - No poden dependre d'estils per defecte de biblioteques ni de cap valor aleatori o heurístic.

6.6.2 Implementació i integració amb l'Agent

Per complir aquests requisits, es va dissenyar i codificar una **funció personalitzada en programació tradicional**, basada en biblioteques de visualització gràfica estàndard (com Matplotlib en Python). Aquesta funció està pensada com una **eina estable**, que pot ser **instrumentada des de l'Agent General** del sistema, però que es manté **fora del seu control directe** per garantir la consistència.

Això significa que:

- **L'Agent pot sol·licitar la generació d'una gràfica** mitjançant una crida estructurada a aquesta eina.
- **No se li permet modificar els paràmetres estètics o tècnics**, assegurant que no hi hagi variabilitat entre assajos.

- Es defineix una **interfície clara** d'entrada: dades normalitzades i metadades associades (tipus de prova, unitats, rangs).
- El resultat és una **imatge exportada en formats estàndard** (PNG, SVG, o PDF), llesta per ser inclosa en l'informe.

6.6.3 Consideracions finals

Tot i el potencial dels LLMs per generar codi flexible, en aquest cas s'ha prioritzat la **seguretat, robustesa i control total de la sortida**. Aquesta decisió també permet un millor procés de validació, ja que qualsevol canvi en l'estètica o estructura del gràfic passa pel codi font, que pot ser versionat i auditat fàcilment. Això contrasta amb l'ús de *prompts*, que poden ser més difícils de gestionar en entorns on es requereix una traçabilitat estricta.

6.7 Generació i ompliment automàtic del document final

Una de les funcionalitats troncal del sistema desenvolupat és la generació automàtica de l'informe tècnic final. Aquest document recull, de manera estructurada i visualment estandarditzada, els resultats obtinguts durant l'anàlisi de les diferents configuracions tècniques. El procés de generació d'aquest informe es basa en una **funció automatitzada que integra dades, gràfiques i metadades dins d'una plantilla Word predefinida**, mitjançant l'ús de la llibreria **Aspose.Words**, una eina potent i orientada a la manipulació programàtica de documents.

6.7.1 Procés general de generació

Aquest mòdul s'activa un cop completada la fase de processament prèvia (anomenada *process_mode*), en què s'analitzen totes les configuracions d'un mode determinat. Com a sortida, aquesta fase produeix:

- **Un fitxer JSON** estructurat, que conté totes les mètriques tècniques i valors rellevants associats a cada configuració (valors de potència, límits, errors detectats, etc.).
- **Un conjunt de gràfiques** generades automàticament a partir de les dades, les quals es desen en un repositori remot (p. ex. Amazon S3), seguint una nomenclatura i estructura que permet l'enllaç directe amb les dades corresponents.

Amb aquesta informació disponible, la funció de generació del document realitza els següents passos:

1. **Càrrega de la plantilla Word:** es fa ús d'un document base amb l'estil i estructura ja definits (capçaleres, taules, seccions, fonts, colors, etc.), garantint la coherència visual amb la resta de documentació del projecte.
2. **Inserció de dades tècniques:** mitjançant **Aspose.Words**, es recuperen les dades del fitxer JSON i s'integren automàticament en les seccions del document corresponents (taules, paràgrafs, etc.).
3. **Enllaç i col·locació de gràfiques:** es vinculen les imatges gràfiques generades a les configuracions pertinents, inserint-les en l'ordre correcte dins del document.
4. **Adaptació final de l'estil:** es revisa programàticament l'alineació, els estils, el format de les taules i el contingut textual, per garantir la uniformitat visual i la qualitat formal del document final.

Aquesta automatització aporta beneficis notables tant a nivell de productivitat com de qualitat del entregable final:

- **Reducció significativa del temps de generació d'informes**, especialment en modes amb moltes configuracions o assajos.
- **Eliminació del risc d'errors humans**, com ara l'assignació incorrecta de gràfiques o la duplicació de dades.
- **Escalabilitat**: el sistema pot generar informes per a centenars de modes amb mínimes modificacions.
- **Uniformitat i consistència professional**: cada document manté exactament el mateix estil, estructura i qualitat, independentment de qui el generi o quan es generi.

6.8 Processament complet de les dades d'un assaig

Per tal de generar un informe final complet i coherent, ha estat necessari implementar un sistema capaç de processar totes les combinacions possibles de modes d'operació i *test points* disponibles dins un assaig. Inicialment, el sistema es va desenvolupar per treballar amb una sola combinació concreta (és a dir, un únic fitxer CSV corresponent a un mode i un punt de test), però per escalar la solució a casos reals, es va ampliar la funcionalitat per tal de poder tractar tots els fitxers de prova d'una sola vegada.

6.8.1 Entrada de dades

Com a entrada, el sistema rep una carpeta que conté tots els fitxers `.csv`, cadascun representant una prova diferent amb una combinació específica de mode d'operació i *test point*. Cada fitxer és processat de manera iterativa mitjançant un bucle automatitzat que aplica el mateix flux d'anàlisi i generació de documents per a cada cas.

6.8.2 Generació i concatenació d'informes

Per cada fitxer processat, es genera un document parcial amb els resultats corresponents. Un cop tots els fitxers han estat tractats, es procedeix a la concatenació d'aquests documents en un únic informe final. Aquest procés permet centralitzar tota la informació de l'assaig en un sol document de sortida.

6.8.3 Gestió automàtica de la indexació

Un dels punts crítics durant la concatenació és la gestió correcta de la numeració dels apartats i de l'índex del document. Per solucionar aquest aspecte, s'ha implementat un sistema de control que actualitza automàticament la indexació cada vegada que s'afegeix un document parcial al conjunt. D'aquesta manera, s'evita la necessitat de realitzar ajustos manuals posteriors i es garanteix que la jerarquia i l'estructura del document final sigui correcta i clara per al lector.

Aquesta automatització del processament massiu i la integració dels resultats permet una generació d'informes molt més eficient, escalable i fiable, adequada per a assajos amb múltiples configuracions i alta variabilitat de dades.

6.9 Anàlisi d'un assaig

Una de les funcionalitats més avançades del sistema és la capacitat d'analitzar de forma automatitzada un informe complet d'assaig, que habitualment pot tenir una extensió d'unes 150 pàgines. Aquest procés es realitza mitjançant un agent intel·ligent que interpreta el contingut del document i extreu informació rellevant segons uns criteris analítics prèviament definits.

6.9.1 Entrada del sistema

L'entrada consisteix en un document complet d'assaig, que conté resultats per a múltiples bandes, *test points* i modes d'operació. Aquest document pot incloure gràfiques, taules de dades, seccions tècniques i comentaris de prova.

6.9.2 Objectius principals de l'anàlisi

L'agent realitza el processament del document amb l'objectiu de generar un resum estructurat i comprensible, enfocat en dos eixos principals:

1. Descripció de l'objecte d'avaluació per banda:

L'agent identifica quines bandes s'han avaluat i proporciona una breu descripció de què s'està mesurant o provant en cadascuna. Això permet contextualitzar millor la naturalesa de les proves i entendre la seva importància tècnica dins l'àmbit de compatibilitat electromagnètica (EMC) o de seguretat funcional.

2. Identificació de resultats crítics:

A continuació, l'agent analitza quines proves han fallat, tot centrant-se en els següents aspectes:

- Els pitjors resultats observats a nivell de mètrica (per exemple, màxims valors d'emissions o mínims de marge).
- Els *test points* més crítics on s'han detectat falles de conformitat.
- Els *operation modes* que han presentat comportaments problemàtics.

6.10 Generació de Dades Sintètiques

Es va desenvolupar una solució basada en **variació percentual global** que aplica el mateix factor matemàtic a totes les dades numèriques del *dataset* simultàniament. Aquest enfocament garanteix que les relacions proporcionals entre valors es mantinguin intactes.

La funcionalitat implementada segueix un flux estructurat:

- **Anàlisi de dades:** Identificació automàtica de valors numèrics versus textuals dins dels fitxers CSV
- **Generació de factor:** Creació d'un factor aleatori únic dins del rang percentual especificat per tot el *dataset*.
- **Aplicació uniforme:** El mateix factor s'aplica a tots els valors numèrics identificats.
- **Preservació de format:** Manteniment de la precisió decimal, separadors i estructura originals dels fitxers.
- **Processament massiu:** Capacitat de processar carpetes senceres de fitxers CSV de forma recursiva.

- **Integració cloud:** Exportació automàtica dels resultats al *bucket* S3 del sistema.

6.10.1 Característiques Tècniques

La implementació utilitza *seeds* únics per cada fitxer per garantir reproductibilitat, manté l'encoding UTF-8 amb separadors de punt i coma propis dels fitxers CSV europeus, i aplica el factor percentual només als valors numèrics, deixant intactes camps textuais, dates i identificadors.

El sistema genera automàticament noms de fitxer descriptius i organitza els resultats en estructures de carpetes coherents dins de l'emmagatzematge *cloud*, facilitant la posterior utilització de les dades sintètiques generades.

6.11 Implementació del *Substream* al Front-end de DOCS

Durant el desenvolupament d'aquest projecte en col·laboració amb l'equip de DOCS, es va produir una circumstància que va condicionar l'abast de la implementació. L'equip estava en procés de **reenginyeria completa de la plataforma DOCS**, la qual cosa va impossibilitar la implementació integral del *stream* complet al *front-end*, ja que aquest es trobava en una fase activa de creació i modificació estructural.

6.11.1 Estratègia d'Implementació Adoptada

Degut a les limitacions temporals i l'estat de desenvolupament de la plataforma, es va optar per implementar la funcionalitat completa al *back-end*, deixant com a **objectiu post-TFG** la migració i implementació del *stream* complet amb el nou front-end de DOCS una vegada finalitzada la seva reestructuració.

Com a primera fase d'aquesta futura implementació, s'ha desenvolupat el ***substream detector de pics***, que constitueix un component modular del sistema general de detecció d'anomalies.

6.11.2 Arquitectura del *Stream* a DOCS

DOCS és una eina que permet crear *streams* de processament de dades de manera **transparent a la programació tradicional**, utilitzant un sistema de caixes visuals

que substitueixen les funcions convencionals i faciliten la construcció de *pipelines* de processament de manera intuïtiva.

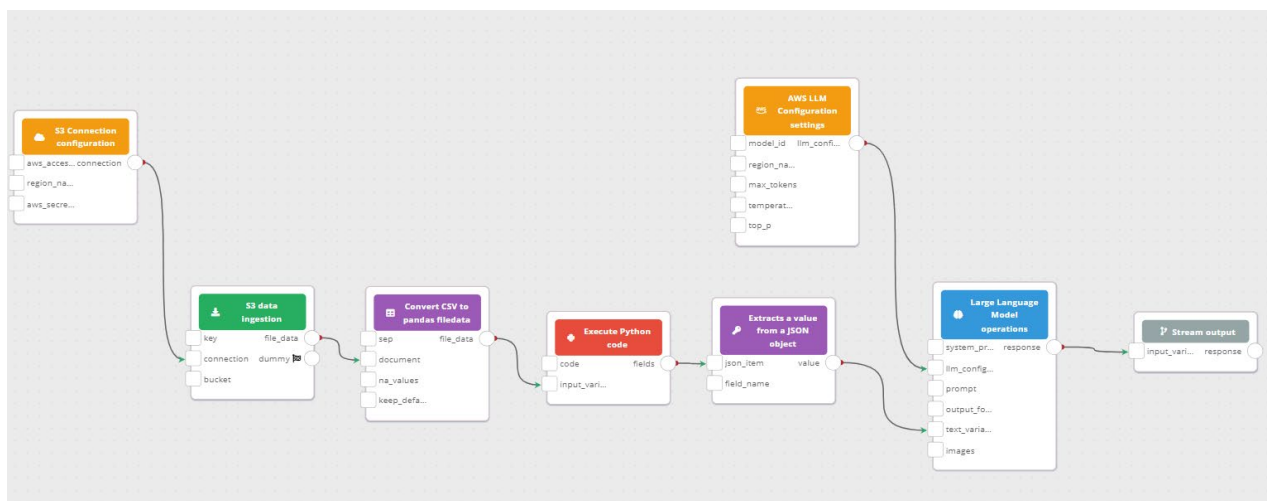


Figura 5. *Substream* detector de pics

El *substream* desenvolupat segueix l'arquitectura modular de DOCS i està compost pels següents components:

1. **Mòduls de connectivitat:** Caixes especialitzades per a la connexió i ingestió de dades des d'Amazon S3
2. **Processament de dades:** Conversió automàtica del fitxer CSV d'entrada a format *DataFrame* per al seu tractament
3. **Preparació de dades:** Caixa d'execució de codi Python personalitzat per adequar les dades als requeriments de l'LLM
4. **Detecció de pics:** Integració amb el model de llenguatge per a la identificació de pics en les dades
5. **Extracció de resultats:** Processament i formatació de la sortida generada pel sistema

Aquest enfocament modular permet una fàcil **mantenibilitat i escalabilitat** del sistema.

7 Avaluació

7.1 Desenvolupament del Sistema d'Indexació Automàtica

7.1.1 Entrada

Document mal indexat

TABLE OF CONTENTS	
Contenido	
1.1.1	Conducted emissions by current HVBAT2 400V Charging_50 3
1.1.1	Conducted emissions by current HVBAT2 Stand by_50 8
1.1.1	Conducted emissions by current HVDCDC1 400V Charging_50 13
1.1.1	Conducted emissions by current HVBAT2 Driving_50 18
1.1.1	Conducted emissions by current HVBAT1 400V Charging_50 23
1.1.1	Conducted emissions by current HVDCDC1 Driving_50 28
1.1.1	Conducted emissions by current HVBAT2 800V Charging_50 33
1.1.1	Conducted emissions by current HVBAT1 Stand by_50 38
1.1.1	Conducted emissions by current HVBAT1 800V Charging_50 43
1.1.1	Conducted emissions by current HVDCDC1 800V Charging_50 48

Figura 6. Exemple d'entrada amb mala indexació

1.1.1	Conducted emissions by current HVBAT2 Stand by_50
1.1.1.1	Conducted emissions by current

Figura 7. Exemple de títol mal indexat

7.1.2 Sortida

Document idèntic corregint la indexació:

TABLE OF CONTENTS

Contenido

1.1.1 Conducted emissions by current HVBAT2 400V Charging_50	3
1.1.2 Conducted emissions by current HVBAT2 Stand by_50	8
1.1.3 Conducted emissions by current HVDCDC1 400V Charging_50	13
1.1.4 Conducted emissions by current HVBAT2 Driving_50	18
1.1.5 Conducted emissions by current HVBAT1 400V Charging_50	23
1.1.6 Conducted emissions by current HVDCDC1 Driving_50	28
1.1.7 Conducted emissions by current HVBAT2 800V Charging_50	33
1.1.8 Conducted emissions by current HVBAT1 Stand by_50	38
1.1.9 Conducted emissions by current HVBAT1 800V Charging_50	43
1.1.10 Conducted emissions by current HVDCDC1 800V Charging_50	48

Figura 8. Índex corregit**1.1.2 Conducted emissions by current HVBAT2 Stand by_50**

1.1.2.1 Conducted emissions by current

VARIANT - CP

Figura 9. Títol corregit**7.1.3 Debug:**

La funció que retorna els títols extrets, amb un JSON on la clau indica el numero de pagina, i el valor, son tots els títols que considera

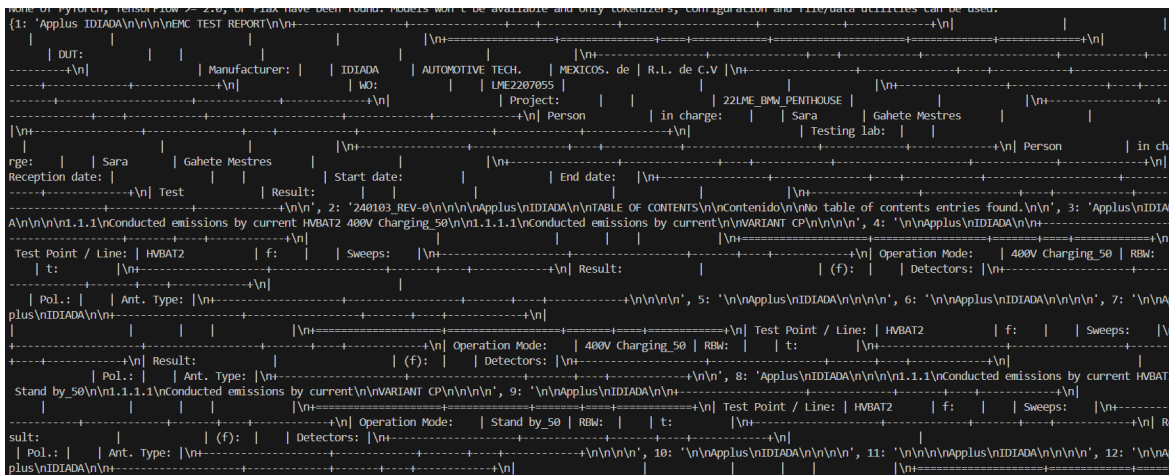


Figura 10. Debug extreuere títols

L'agent que retorna el títols que considera amb les puntuacions corregides.

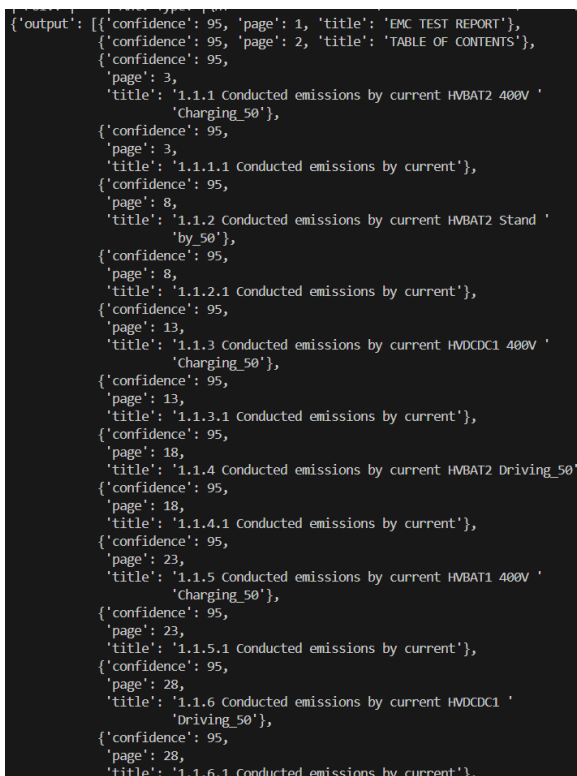


Figura 11. Debug detector de títols

7.2 Generació de enllaços a una Test Matrix

S'ha generat automàticament una *Test Matrix* amb enllaços directes a les seccions corresponents del document. Cada enllaç es construeix a partir de la combinació del nom del

tipus d'assaig, el *Test Point* i el *Operation Mode*, permetent una navegació eficient i contextual dins del document final.

7.2.1 Entrada

OP. MODE	BAND	FREQUENCY RANGE (MHz)	DETECTOR	RBW (kHz)	TEST POINT	SECTION	RESULT
«OperationMode»					«TestPoint»	«linkGraphic»	

Figura 12. Taula de plantilla

7.2.2 Sortida

OP. MODE	BAND	FREQUENCY RANGE (MHz)	DETECTOR	RBW (kHz)	TEST POINT	SECTION	RESULT
Single Rail					DUT FRONT	1.1.1	
					CABLE HARNESS	1.1.2	

OP. MODE	BAND	FREQUENCY RANGE (MHz)	DETECTOR	RBW (kHz)	TEST POINT	SECTION	RESULT
Double Rail					DUT FRONT		
					CABLE HARNESS		

OP. MODE	BAND	FREQUENCY RANGE (MHz)	DETECTOR	RBW (kHz)	TEST POINT	SECTION	RESULT
Ambient					DUT FRONT		
					CABLE HARNESS		

OP. MODE	BAND	FREQUENCY RANGE (MHz)	DETECTOR	RBW (kHz)	TEST POINT	SECTION	RESULT
Stand by					DUT FRONT		
					CABLE HARNESS		

Figura 13. *Test Matrix*

S'ha de tenir en compte que les taules on no hi ha l'apartat de *Section* es perquè no hi ha la gràfica que correspon al punt. En aquest cas es deixa en blanc.

1.1.1 Radiated emissions DUT FRONT Single Rail

1.1.1.1 Radiated emissions

Figura 14 Títol secció 1.1.1

En el cas de l'enllaç identificat com a “1.1.1”, aquest redirigeix automàticament a la secció del document on es troba la gràfica corresponent, juntament amb el títol i les dades associades, confirmant així la coherència entre la *Test Matrix* i el contingut del document.

7.3 Implementació de la detecció de pics mitjançant un Agent

7.3.1 Sortida

Detecció de pics, amb i sense tolerància de un test

```
peaks.py
None of PyTorch, TensorFlow >= 2.0, or Flax have been found. Models won't be available and only tokenizers, configuration and file/data ut
can be used.
=== DETECCIÓ AUTOMÀTICA AMB TOLERÀNCIA BAIXA (0.5) ===
Pics detectats: 5
Estadístiques: {'strict_peaks': 16, 'tolerant_peaks': 28, 'approximate_peaks': 46, 'total_detected': 90, 'tolerance_used': 0.5}

=== DETECCIÓ AUTOMÀTICA AMB TOLERÀNCIA MITJANA (0.2) ===
Pics detectats: 5
Estadístiques: {'strict_peaks': 16, 'tolerant_peaks': 11, 'approximate_peaks': 9, 'total_detected': 36, 'tolerance_used': 0.2}

=== DETECCIÓ AUTOMÀTICA AMB TOLERÀNCIA ALTA (0.1) ===
Pics detectats: 5
Estadístiques: {'strict_peaks': 16, 'tolerant_peaks': 6, 'approximate_peaks': 5, 'total_detected': 27, 'tolerance_used': 0.1}

=== MILLORS 3 PICS DETECTATS AUTOMÀTICAMENT (tolerància 0.1) ===
Pic 1: Freq=86.90, Margin=-1.691, Confiança=high (1.69), Mètode=strict
Pic 2: Freq=87.30, Margin=-1.676, Confiança=high (1.68), Mètode=strict
Pic 3: Freq=86.87, Margin=-1.637, Confiança=medium (1.15), Mètode=tolerant
```

Figura 15. Detector de pics

De un csv amb dades es prova que passant-l'hi al detector de pics ens retorna els pics esperats, en aquest cas troba 16 pics estrictes, es a dir que si es consideren pics, però com a la configuració hem posat 5 pics ens agafa sols els 5 pitjors, de tolerants, també en troba bastants segons la tolerància que li indiquem, els pics tolerants son els que acceptem per un marge x que ens indica l'usuari.

7.4 Anàlisi d'un assaig

L'anàlisi realitzat es divideix en dos blocs principals:

1. **Explicació de les bandes:** Interpretació dels resultats obtinguts per a cada banda processada, destacant els valors rellevants i la seva relació amb els criteris tècnics establerts.
2. **Comparació de configuracions:** Avaluació entre diferents configuracions d'assaig per identificar diferències significatives en el comportament dels resultats o possibles patrons comuns.

Per garantir la fiabilitat del procés, **s'ha verificat que els resultats de l'anàlisi coincideixen amb les dades reals del document**, assegurant així la coherència i la validesa de les conclusions obtingudes.

EMC TEST ANALYSIS

Generated: 2025-07-04 12:32:59

1. FREQUENCY BANDS EXPLANATION

B1-B4 (0.1 Hz - 30 Hz)

What it tests: Low frequency emissions

Automotive systems: Power electronics, motor controllers

NFC (13 Hz - 14 Hz)

What it tests: Near-field communications interference

Automotive systems: Keyless entry, wireless charging

4m_BOS (80 Hz - 90 Hz)

What it tests: 4-meter band emissions

Automotive systems: Emergency services communication

B5 (30 Hz - 125 Hz)

What it tests: Mid-range frequency emissions

Automotive systems: Various electronic control units

FM (70 Hz - 125 Hz)

What it tests: FM radio band interference

Automotive systems: In-vehicle entertainment systems

2. TEST CONFIGURATION COMPARISON

2.1 Test Point Performance

HVDCDC1+: CRITICAL (Worst: -28.83 dB)

Issue: Highest emissions in multiple bands, especially in MODE-3

HVDCDC1-: CRITICAL (Worst: -28.66 dB)

Figura 16. Definició bandes

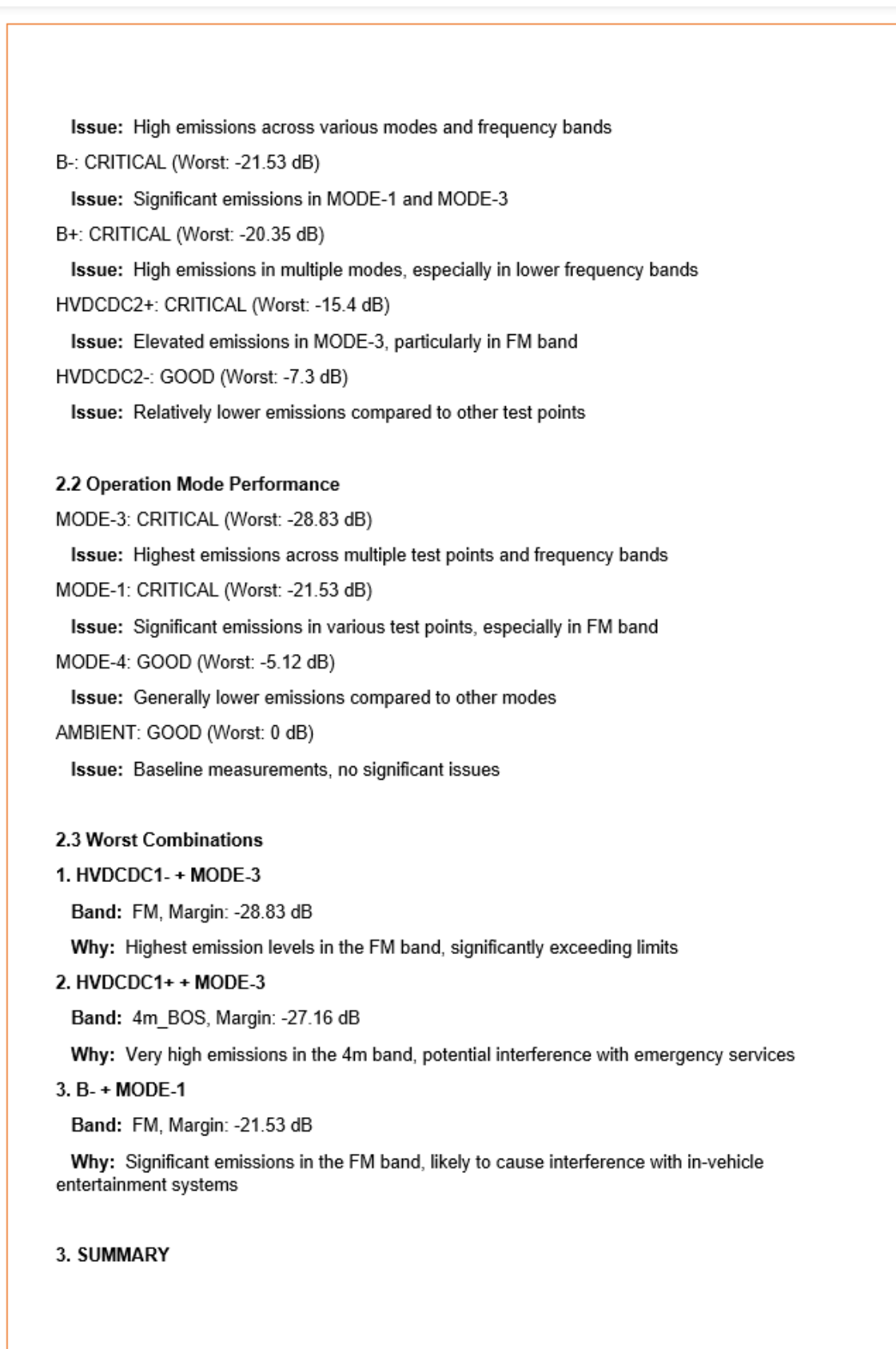


Figura 17. Comparativa

The EMC test results reveal critical issues across multiple test points and operation modes. HVDCDC1+ and HVDCDC1- show the most severe emissions, particularly in MODE-3. The FM band (70-125 Hz) is consistently problematic across various test points. MODE-3 is the worst-performing operation mode, while MODE-4 shows relatively better performance. Immediate mitigation strategies are needed, focusing on HVDCDC1+/- in MODE-3 and addressing FM band emissions across all critical test points.

Figura 18. Resum

7.5 Resultats Finals

7.5.1 Entrada

L'entrada al sistema pot ser una carpeta de AN, carpeta de CP o carpeta de RE, depenen de l'assaig. Amb diferents csv per les dades

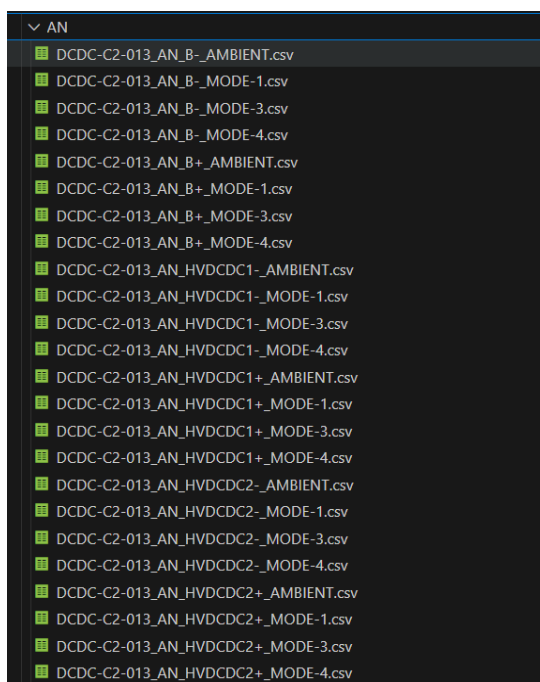


Figura 19. Exemple dades d'assaig AN

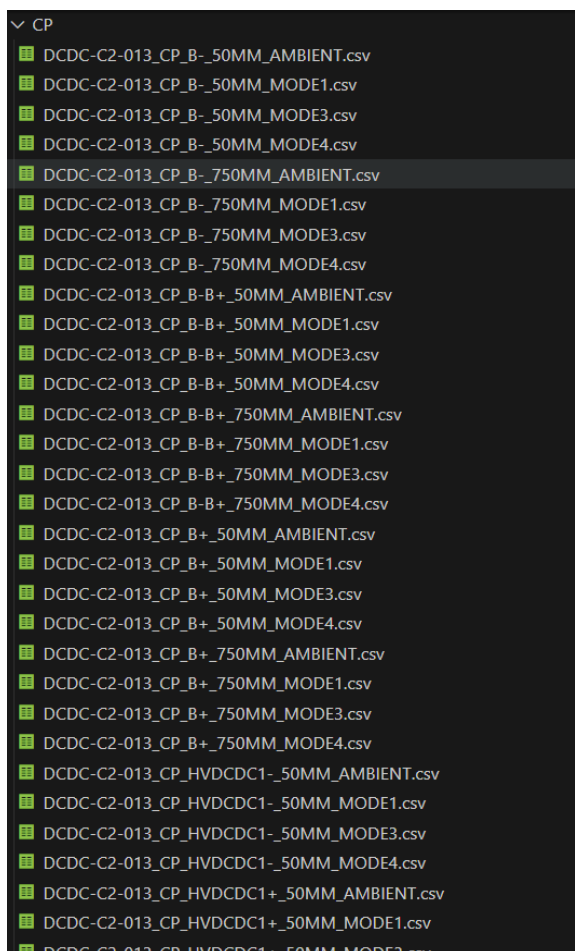


Figura 20. Exemple dades d'assaig CP

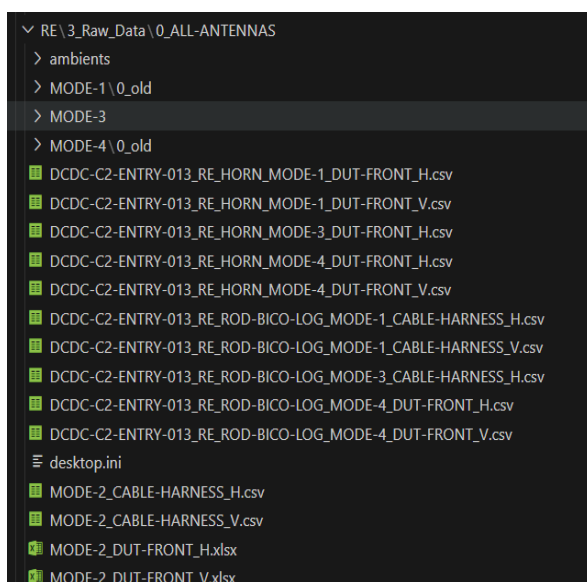


Figura 21. Exemple dades d'assaig RE

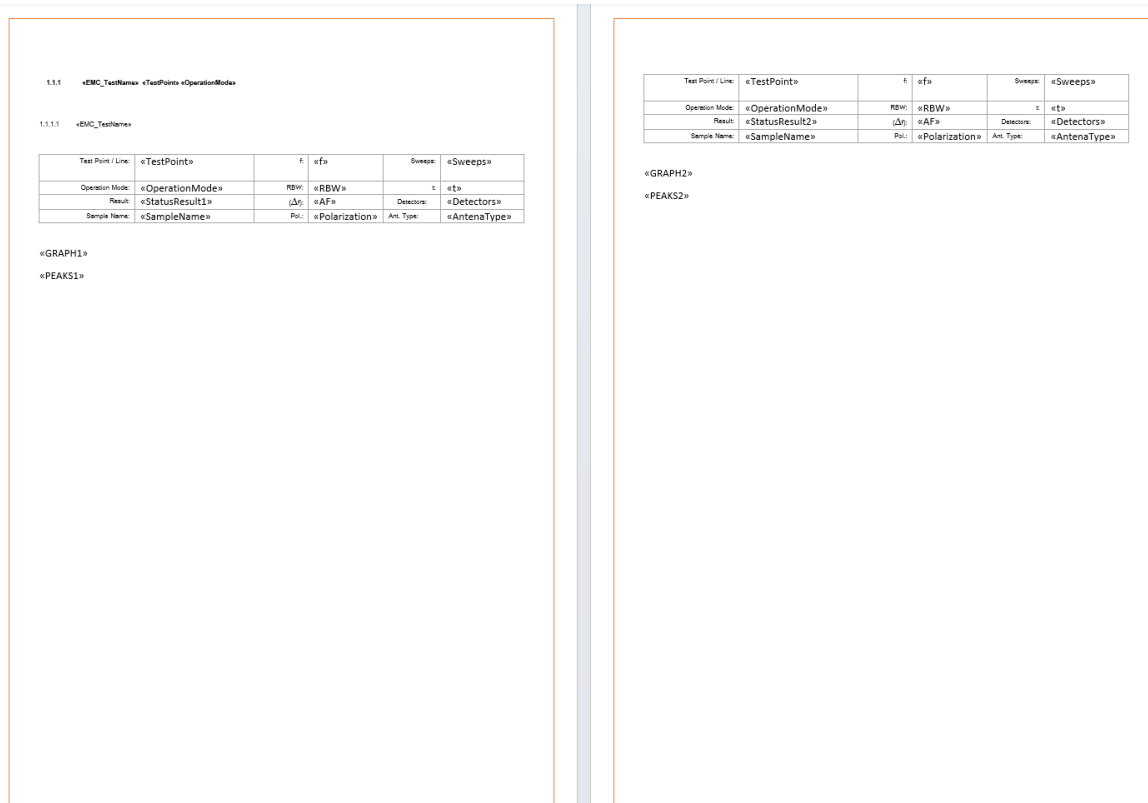


Figura 23. Plantilla entrada pàgina 1 i 2

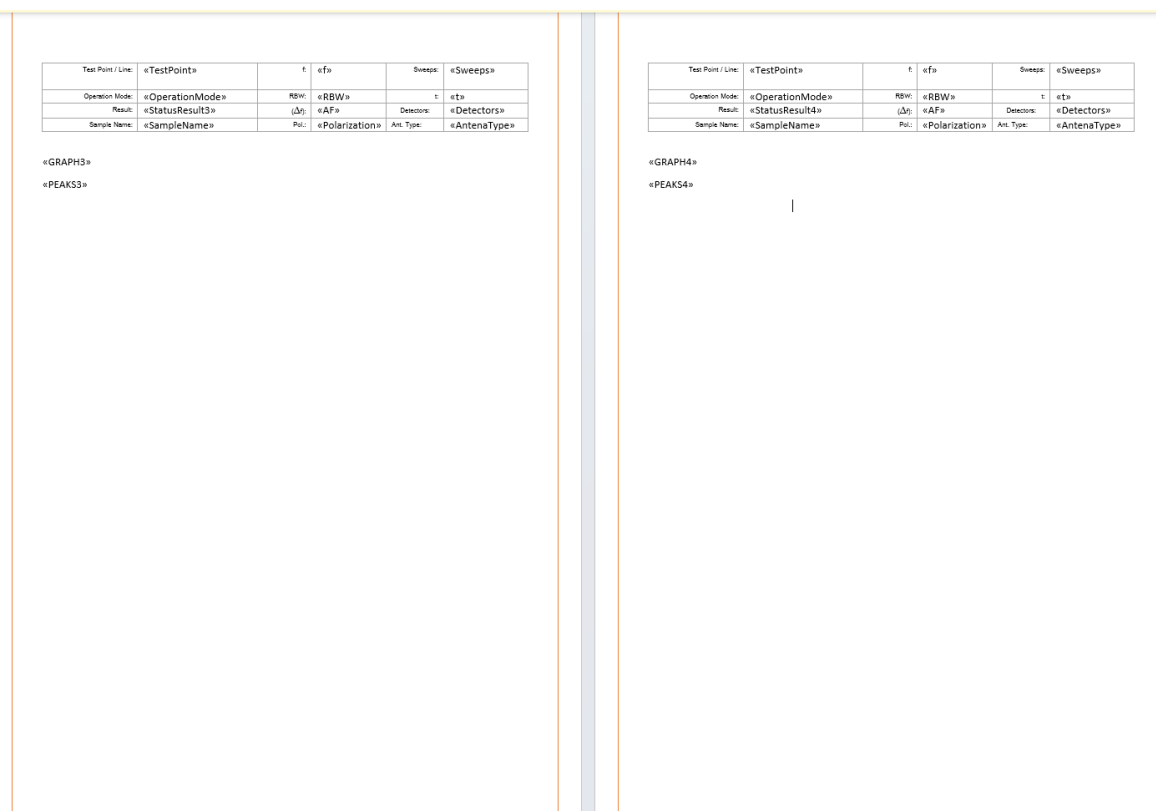


Figura 24. Plantilla d'entrada pagina 3 i 4

Test Point / Line:	«TestPoint»	f:	«f»	Sweeps:	«Sweeps»
Operation Mode:	«OperationMode»	RBW:	«RBW»	ε:	«ε»
Result:	«StatusResult5»	(Δf):	«AF»	Descors:	«Detectors»
Sample Name:	«SampleName»	Pol:	«Polarization»	Ant. Type:	«AntenaType»

«GRAPH5»
«PEAK5»

Test Point / Line:	«TestPoint»	f:	«f»	Sweeps:	«Sweeps»
Operation Mode:	«OperationMode»	RBW:	«RBW»	ε:	«ε»
Result:	«StatusResult6»	(Δf):	«AF»	Descors:	«Detectors»
Sample Name:	«SampleName»	Pol:	«Polarization»	Ant. Type:	«AntenaType»

«GRAPH6»
«PEAK6»

Figura 25. Plantilla d'entrada pàgina 5 i 6

Un cop implementat tot el sistema, es va procedir a la seva validació i prova mitjançant dades sintètiques, generades a partir de casos reals, per tal d'avaluar el comportament del procés complet. L'objectiu era verificar que, a partir d'un conjunt de dades d'entrada, el sistema fos capaç de generar automàticament un document tècnic final amb la mateixa qualitat, rigor i estructura que els informes tradicionals generats manualment.

7.5.2 Validació funcional

Inicialment, es va provar el sistema amb un únic fitxer .csv corresponent a un *operation mode* i un *test point* específics. Aquest primer test va permetre avaluar el flux sencer, des del processament de dades fins a la generació del document final i la inserció de gràfiques. Un cop confirmat el bon funcionament en aquest cas simple, es va procedir a escalar la prova per a la totalitat dels fitxers disponibles.

El sistema es va executar sobre un conjunt de **24 fitxers .csv**, els quals contenien dades de diferents assajos i configuracions. Aquests fitxers es divideixen en **6 bandes**, generant en total **144 configuracions analitzades**. Aquesta execució completa va permetre validar el comportament del sistema en condicions més realistes i complexes.

7.5.3 Validació de la sortida generada

L'aspecte més crític d'aquest projecte no és tant la varietat dels jocs de proves com la **correcta generació del document final**. En aquest sentit, es va verificar que:

- **Els camps del document s'omplen correctament** en funció de les dades d'entrada.
- **Les gràfiques es generen i s'insereixen correctament** dins del document, mantenint l'ordre i correspondència amb les dades.
- **Els estils, formats i estructures** es mantenen coherents i professionals, tal com defineix la plantilla original.
- El sistema incorpora **mecanismes de depuració (debug)** que permeten seguir l'estat d'execució pas a pas, facilitant la identificació de problemes puntuals.

Una part clau per garantir aquest bon funcionament és l'ús adequat de **plantilles Word prèviament definides**. Aquestes plantilles han de tenir la **estructura i marques de substitució correctes**, ja que qualsevol desajust entre plantilla i dades pot provocar errors d'execució o resultats incorrectes. Això posa de manifest la importància de cuidar meticulosament el disseny de la plantilla i l'alineació amb les dades d'entrada.

7.5.4 Comparació amb el sistema original

Per tal de validar la fidelitat dels resultats, es va realitzar una comparació entre els informes generats amb aquest nou sistema i els informes oficials generats anteriorment amb el sistema **DARWIN**. Els resultats van ser **exactament iguals**, tant pel que fa a les mètriques com a les gràfiques. Aquesta equivalència confirma que el nou sistema automatitzat **manté el mateix nivell de rigor i precisió**, però amb un alt grau d'autonomia i escalabilitat.

7.5.4.1 Comparativa d'eficiència en el desenvolupament

És rellevant destacar les **diferències significatives** en els recursos necessaris per a la implementació de cada sistema:

- **Sistema DARWIN:**
 - Temps d'implementació: **12 mesos**
 - Recursos humans: 1 persona a jornada completa + 1 persona a mitja jornada
 - Total d'esforç: **18 mesos-persona**
- **Sistema amb IA:**
 - Temps d'implementació: **3 mesos**
 - Recursos humans: 1 persona a mitja jornada
 - Total d'esforç: **1,5 mesos-persona**

Aquesta comparativa evidencia **una reducció del 92%** en l'esforç requerit, demostrant una major eficiència en el procés de desenvolupament. **Cal destacar especialment que l'eina actual ha estat desenvolupada per un estudiant a mitja jornada**, fet que posa en relleu la potència i accessibilitat de les tecnologies d'intel·ligència artificial actuals per a la resolució de problemes complexos.

A més, el sistema actual ha resolt totes les limitacions identificades en el sistema predecessor i ha incorporat funcionalitats analítiques avançades que proporcionen valor afegit significatiu.

7.5.4.2 Rendiment d'execució

Pel que fa al **temps d'execució**:

- **Sistema DARWIN**: 50 segons per fitxer
- **Sistema amb IA**: 2 minuts per fitxer

L'increment en el temps d'execució (+140%) s'atribueix principalment a les **crides als models de llenguatge** (LLMs) necessàries per a les funcionalitats analítiques avançades. Aquest augment temporal és **justificat** per l'increment substancial en les capacitats analítiques i la qualitat dels informes generats.

7.5.5 Temps d'execució i limitacions

Tot i l'èxit funcional, cal destacar que el sistema presenta **limitacions de rendiment**. El processament complet de les 144 configuracions esmentades pot requerir **un temps considerable**, a causa de dos factors principals:

- **Ús intensiu d'Agents i crides a serveis d'IA com Amazon Bedrock**, que impliquen una latència no negligible per cada operació.
- **Gestió d'un gran volum de dades** i la seva inserció seqüencial dins del document, especialment quan hi ha dependències entre seccions o processos.

Aquest comportament fa que el sistema sigui percebut com a lent, sobretot en execucions completes. Malgrat això, cal entendre que **l'automatització d'aquest procés redueix de forma dràstica la feina manual** i els errors associats, tot i que això impliqui consumir més temps de màquina i recursos computacionals.

7.5.6 Millores futures

Com a possible millora, es contempla la **paral·lelització de l'execució per fitxer** (és a dir, tractar diversos .csv en paral·lel), fet que podria reduir notablement el temps total d'execució. Tanmateix, no totes les fases són paral·lelitzables: certes operacions requereixen una execució seqüencial per mantenir la coherència del procés global.

En qualsevol cas, l'arquitectura modular del sistema facilita aquesta evolució futura, i es podrà adaptar fàcilment per incorporar mètodes d'execució paral·lela en fases específiques quan sigui necessari.

Sortida de l'execució sobre un csv que contenen un mode amb un *Test Point* :

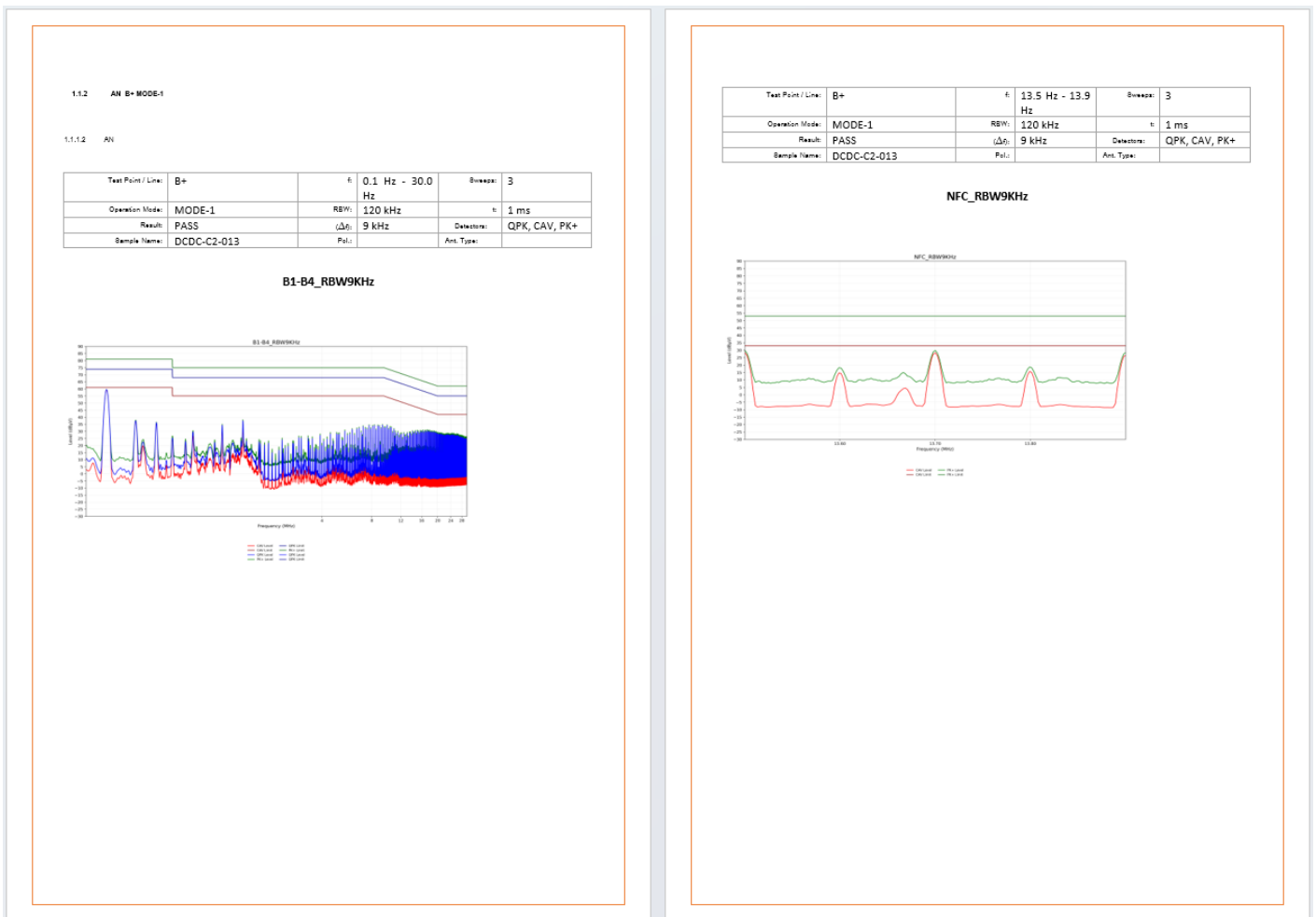


Figura 26. Bandes B1-B4 i NFC

Figura 27. Bandes B5 i FM **Figura 28.** Bandes B1-B4 i NFC

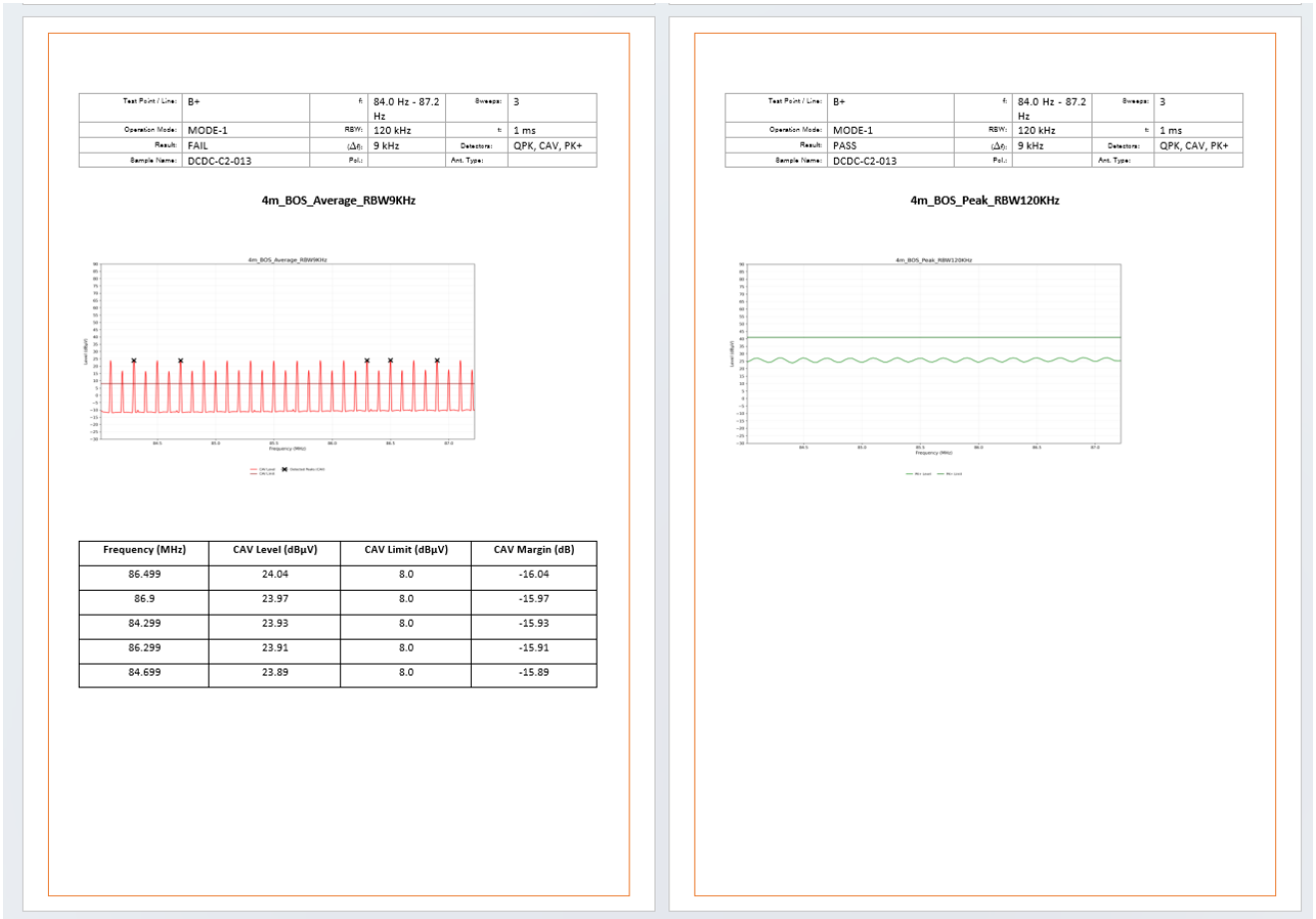


Figura 32. Bandes 4m Bos Average i Peak

Figura 33. Bandes 4m Bos Average i Peak

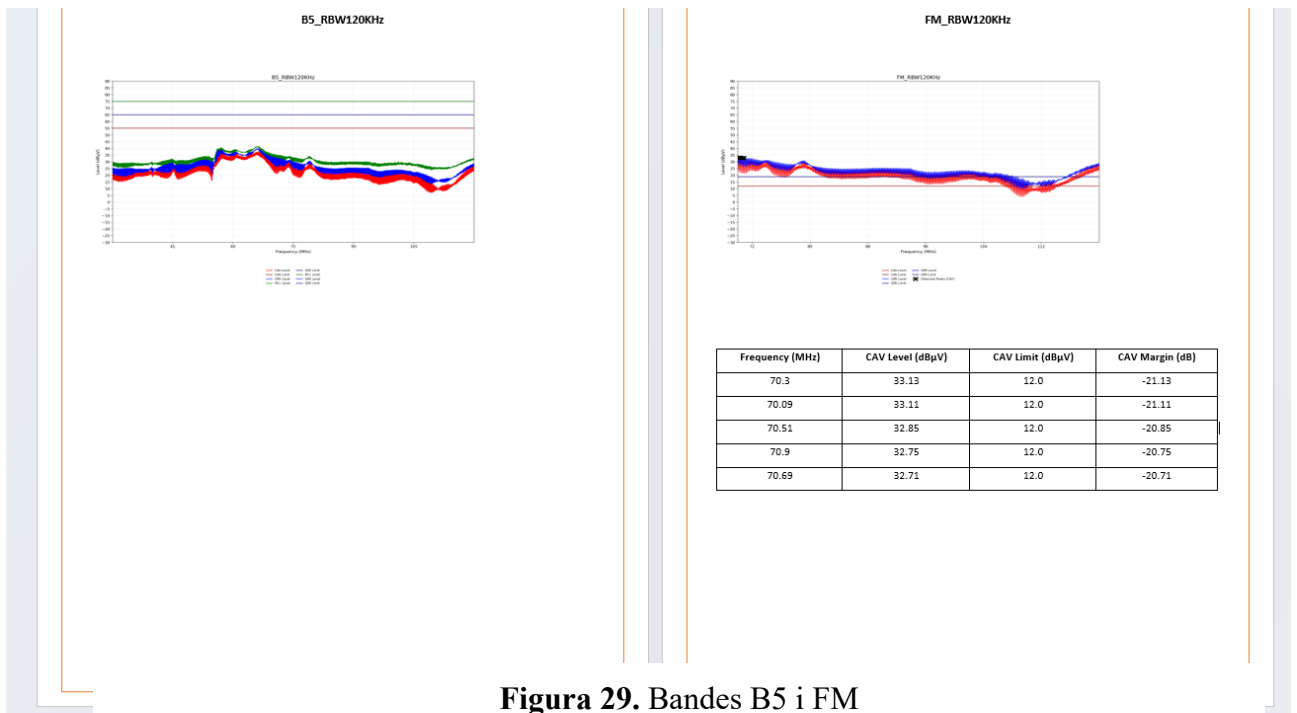


Figura 29. Bandes B5 i FM

Figura 30. Bandes 4m Bos Average i Peak
Figura 31. Bandes B5 i FM

Aquest resultat correspon a una prova individual del tipus **test AN**, realitzada sobre el **Test Point B+** i amb el **Mode-3** seleccionat. En aquest cas, s'ha utilitzat la **configuració automàtica**, que és la més habitual en el procés, i s'ha indicat que es retornin un màxim de **5 pics**, també un valor típicament emprat.

Tot i això, en la banda NFC només es retorna **1 pic**, ja que el sistema no ha detectat altres valors que compleixin els criteris definits per considerar-se un pic vàlid.

Aquest comportament demostra que el sistema s'adapta de manera dinàmica a les dades d'entrada, retornant únicament els resultats que compleixen les condicions analítiques establertes.

8 Conclusions:

Aquest treball ha desenvolupat amb èxit un sistema intel·ligent per a l'automatització de la generació d'informes d'assajos de EMC assolint tots els objectius plantejats inicialment. La solució implementada combina de manera eficaç la intel·ligència artificial generativa amb programació tradicional, utilitzant models de llenguatge natural LLM com *Claude Sonnet* integrats amb serveis d'AWS per crear una arquitectura robusta i escalable.

El sistema ha demostrat la seva capacitat per automatitzar processos complexos com la indexació automàtica de documents, la generació d'enllaços a gràfiques, la detecció intel·ligent de pics mitjançant llenguatge natural i la identificació semàntica de configuracions equivalents. Aquesta aproximació híbrida ha permès superar les limitacions dels mètodes tradicionals basats en líndars fixos, oferint una flexibilitat notable per adaptar-se a diferents tipus d'assajos i configuracions.

El valor clau d'aquesta implementació rau en la seva escalabilitat, contrastant significativament amb la programació fixa existent anteriorment. Hem aconseguit desenvolupar un sistema escalable que permet una adaptació dinàmica i flexible a nous requisits i configuracions, eliminant les limitacions dels sistemes rígids tradicionals.

L'impacte empresarial del projecte és significatiu, ja que la solució redueix considerablement el temps de generació d'informes, minimitza els errors humans en la documentació i alleugereix la càrrega de treball manual dels enginyers.

Des d'una perspectiva d'innovació, aquest treball estableix les bases per a futures millores com l'extensió a altres tipus d'assajos automotrius, la implementació de capacitats d'aprenentatge continu i la integració amb sistemes de gestió de qualitat més amplis. El desenvolupament d'aquest projecte ha permès aplicar coneixements teòrics en un context professional real, demostrant la capacitat de la intel·ligència artificial per resoldre problemes complexos en entorns industrials especialitzats.

9 Referències

- [1] Data Camp ¿Qué es un LLM? Guía sobre los grandes modelos lingüísticos y su funcionamiento [¿Qué es un LLM? Guía sobre los grandes modelos lingüísticos y su funcionamiento | DataCamp](#)
- [2] Amazon Web Services – Amazon Textract Layout [Amazon Textract’s new Layout feature introduces efficiencies in general purpose and generative AI document processing tasks | Artificial Intelligence](#)
- [3] AWS Samples Git Hub – Amazon Textract Texttraction [Using Layout Analysis — amazon-textract-textractor 1.0.0 documentation](#)
- [4] Aspose Words [Aspose.Words for Python via .NET|Documentation](#)
- [5] Aspose Words Tables [Working with Tables in Python|Aspose.Words for Python via .NET](#)
- [6] Aspose Words Hyphenation [Working with Hyphenation in Python|Aspose.Words for Python via .NET](#)
- [7] Aspose Words Markdown [Working with Markdown Features|Aspose.Words for Python via .NET](#)
- [8] Aspose Words Headers [Working with Headers and Footers|Aspose.Words for Python via .NET](#)
- [9] Aspose Words Table of Contents [Working with Table of Contents|Aspose.Words for Python via .NET](#)
- [10] Aspose Words Styles and Themes [Working with Styles and Themes|Aspose.Words for Python via .NET](#)
- [11] Aspose Words Fonts [with Fonts in Python|Aspose.Words for Python via .NET](#)
- [12] Aspose Words Sections [Working with Sections in Python|Aspose.Words for Python via .NET](#)
- [13] Docs AWS Bedrock [What is Amazon Bedrock? - Amazon Bedrock](#)
- [14] Draw.io [Untitled Diagram - draw.io](#)
- [15] AWS S3 [AWS | Almacenamiento de datos seguro en la nube \(S3\)](#)
- [16] AWS Anthropic Claude 3.5 [Anthropic’s Claude 3.5 Sonnet model now available in Amazon Bedrock: Even more intelligence than Claude 3 Opus at one-fifth the cost | AWS News Blog](#)
- [17] AWS Docs Claude 3.5 [Anthropic Claude 3.5 Sonnet v2 - Amazon Bedrock](#)
- [18] Agentic-core Tutorial - Documentació interna de l’equip
- [19] Python Documentation [3.13.5 Documentation](#)
- [20] Pandas [pandas - Python Data Analysis Library](#)
- [21] Matplotlib [Matplotlib — Visualization with Python](#)
- [22] Boto3 [boto3·PyPI](#)

10 Annexes:

10.1 Posada en marxa

10.1.1 Configuració de l'Emmagatzematge (AWS S3)

1. Connexió a S3:

- Establir la connexió amb el *bucket* d'Amazon S3
- Configurar les credencials d'accés (Access Key, Secret Key)
- Verificar els permisos de lectura i escriptura

2. Càrrega de Fitxers:

- Penjar la carpeta amb els documents d'assajos EMC a S3
- Organitzar l'estructura de directoris segons els requeriments del sistema
- Verificar la integritat dels fitxers penjats

10.1.2 Configuració de Llicències

Gestió de Llicència Aspose:

- Obtenir la llicència vàlida d'Aspose
- Configurar la llicència dins l'entorn de desenvolupament
- Verificar l'activació correcta per evitar limitacions de funcionalitat

10.1.3 Instal·lació del Framework

Instal·lació d'Agentic-core:

- Descarregar i instal·lar el *framework* agentic-core
- Configurar les dependències necessàries
- Verificar la compatibilitat amb la versió de Python utilitzada

10.1.4 Configuració del Sistema

Configuracions Personalitzades:

- Aplicar les configuracions específiques del client
- Establir els paràmetres de connexió amb els sistemes d'IDIADA
- Configurar les regles de processament dels informes EMC
- Definir els *templates* i formats de sortida

10.1.5 Execució i Verificació

6. Posada en Funcionament:

- Executar el sistema per primera vegada
- Realitzar proves de funcionament amb documents de mostra
- Verificar la generació correcta d'informes
- Validar la integració amb els sistemes existents

10.2 Requeriments

10.2.1 *Requeriments de programari*

10.2.1.1 Lenguatges de Programació

- **Python 3.8+:** Lenguatge principal per al desenvolupament del sistema intel·ligent
 - Compatible amb les llibreries d'IA i processament de documents
 - Suport per a *frameworks* de *machine learning*

10.2.1.2 Entorns de Desenvolupament

- **Visual Studio / Visual Studio Code:**
 - IDE recomanat per al desenvolupament i depuració
 - Extensions per a Python i integració amb AWS
 - Facilita la gestió de projectes i control de versions

10.2.1.3 *Frameworks* i Llibreries Especialitzades

- **Agentic-core:**
 - *Framework* específic per al desenvolupament d'agents intel·ligents
 - Proporciona les funcionalitats base per a la IA del sistema
 - Facilita la integració amb processos de generació d'informes

10.3 Requeriments de llicències

10.3.1 *Llicències Comercials*

- **Llicència Aspose:**
 - **Finalitat:** Processament avançat de documents (PDF, Word, Excel)
 - **Funcionalitats:** Extracció de dades, manipulació de formats, generació d'informes
 - **Tipus:** Llicència comercial amb suport tècnic
 - **Importància:** *Essencial* per al processament dels documents tècnics EMC

10.4 Requeriments de infraestructura

10.4.1 *Serveis en el Núvol*

- **Accés a AWS (Amazon Web Services):**
 - **AWS S3:** Emmagatzematge de documents i fitxers del sistema
 - **AWS IAM:** Gestió d'identitats i permisos d'accés

- **Credencials d'accés:** *Access Key* i *Secret Access Key* amb permisos adequats

10.5 Codis

10.5.1 Llistem csv's en un bucket de s3

```
def list_csv_files_in_s3(bucket_name, prefix):
    try:
        aws_session = boto3.Session(region_name="eu-central-1")
        s3_client = aws_session.client('s3')

        csv_files = []

        response = s3_client.list_objects_v2(Bucket=bucket_name,
        Prefix=prefix)

        if 'Contents' in response:
            for obj in response['Contents']:
                file_key = obj['Key']
                if file_key.lower().endswith('.csv') and not
file_key.endswith('/'):
                    csv_files.append(file_key)

        return csv_files

    except Exception as e:
        return []
```

10.5.2 Llistem documents en un bucket de s3

```
def list_docx_files_in_s3(bucket_name, prefix):
    try:
        aws_session = boto3.Session(region_name="eu-central-1")
        s3_client = aws_session.client('s3')

        docx_files = []

        paginator = s3_client.get_paginator('list_objects_v2')
        page_iterator = paginator.paginate(Bucket=bucket_name,
        Prefix=prefix)

        for page in page_iterator:
            if 'Contents' in page:
                for obj in page['Contents']:
                    file_key = obj['Key']
                    if file_key.lower().endswith('.docx') and not
file_key.endswith('/'):
                        docx_files.append(file_key)

        return docx_files

    except Exception as e:
        return []
```

```

        docx_files.append({
            'key': file_key,
            'name': file_key.split('/')[-1],
            'last_modified': obj['LastModified'],
            'size': obj['Size']
        })

    docx_files.sort(key=lambda x: x['key'])

    return docx_files, s3_client

except Exception as e:
    return [], None

```

10.5.3 Obtenir credencials amb diferents opcions

```

def get_credentials():
    try:
        set_environment()
        if os.environ.get('ACCESS_KEY_AWS') and
os.environ.get('SECRET_KEY_AWS'):
            return {
                'aws_access_key_id':
os.environ.get('ACCESS_KEY_AWS'),
                'aws_secret_access_key':
os.environ.get('SECRET_KEY_AWS'),
                'region_name': 'eu-central-1'
            }
    except Exception:
        pass

    if os.environ.get('ACCESS_KEY_AWS') and
os.environ.get('SECRET_KEY_AWS'):
        return {
            'aws_access_key_id': os.environ.get('ACCESS_KEY_AWS'),
            'aws_secret_access_key':
os.environ.get('SECRET_KEY_AWS'),
            'region_name': 'eu-central-1'
        }

    if os.environ.get('AWS_ACCESS_KEY_ID') and
os.environ.get('AWS_SECRET_ACCESS_KEY'):
        return {
            'aws_access_key_id':
os.environ.get('AWS_ACCESS_KEY_ID'),
            'aws_secret_access_key':
os.environ.get('AWS_SECRET_ACCESS_KEY'),
            'region_name': os.environ.get('AWS_REGION', 'eu-central-
1')
        }

```

```
return None
```

10.5.4 *Configurem client bedrock i sessió S3*

```

if s3_credentials is None:
    s3_credentials = get_credentials()

model_id = "anthropic.claude-3-5-sonnet-20240620-v1:0"

aws_client = set_client(
    service_name="bedrock-runtime",
    region_name=s3_credentials.get('region_name', 'eu-central-1') if s3_credentials else "eu-central-1",
    aws_access_key_id=s3_credentials.get('aws_access_key_id') if s3_credentials else None,

aws_secret_access_key=s3_credentials.get('aws_secret_access_key') if s3_credentials else None
)

aws_session = boto3.Session(
    region_name=s3_credentials.get('region_name', 'eu-central-1') if s3_credentials else "eu-central-1",
    aws_access_key_id=s3_credentials.get('aws_access_key_id') if s3_credentials else None,

aws_secret_access_key=s3_credentials.get('aws_secret_access_key') if s3_credentials else None
)

llm_config = AWSLLMConfig(session=aws_session,
llm_model_id=model_id)
llm_params = LLMParameters(max_tokens=8192, temperature=0.7,
top_p=1, top_k=1)
tool_config = ToolConfig(tool_list=[emc_folder_analysis_tool])

```

10.5.5 *Codi de un agent amb una Foramatter Tool:*

```

system_prompt = """You are expert guessing titles or headers of a document"""
prompt = """

## Input Format
<data>
List of texts that the position are the page number, and in the texts are the titles that we are searching
</data>

```

```

## Rules and Constraints
2. DO NOT modify or split the original text strings
3. Fix in the structure of the titles, and the output have to be
similar, or equal
4. ONLY use complete text strings from the input
5. Maintain the original text order when possible
6. Use proper numerical hierarchical notation (1.1.1, 1.1.1.1, etc.)

## Title Detection Criteria

## Hierarchy Guidelines
1. Test sections: 1.1.1
2. Test subsections: 1.1.1.1
3. Each new test maintains this two-level structure
4. Numbering increases for each new test (1.1.1, 1.1.2, 1.1.3, etc.)

## Input Data to Process
{input}

## Expected Output Format
<data>
[
  "title": "1.1.1 title1", "page": x,
  "title": "1.1.1.1 subtitle1", "page": x,
  "title": "1.1.2 title2", "page": x,
  "title": "1.1.2.1 subtitle1", "page": x
  ...
]
</data>

## Pattern Validation

## Additional Requirements
1. Maintain consistent hierarchical numbering
2. Preserve original text content exactly as found
3. Maintain the sequence of tests as found in the input
"""

def detect_titles(layout) -> dict:
    #Configuration
    model_id = "anthropic.claude-3-5-sonnet-20240620-v1:0"
    aws_session = boto3.Session(
        region_name="eu-central-1",
    )
    llm_config = AWSLLMConfig(session=aws_session,
llm_model_id=model_id)
    llm_params = LLMParameters(max_tokens=8192, temperature=1,
top_p=1, top_k=1)

```

```

llm = LLM(llm_config=llm_config)

class OutputDict(BaseModel):
    title: str = Field(description="titles")
    page: int = Field(description="page")
    confidence: int = Field(description="confidence")

class TestOutputSchema(BaseModel):
    output: List[OutputDict]

    formater_tool_config =
FormatterTool.get_formatter_tool_config(output_schema=TestOutputSchema)

roles_agent = Agent(
    llm_config=llm_config,
    system_prompt=system_prompt,
    prompt_template=prompt,
    tool_config=formater_tool_config,
    name="Detection of titles",
)

llm_response = roles_agent.run(
    input_variables={"input":layout},
    llm_params= llm_params
)

    formatted_output =
au.extract_tool_use(llm_response.response_message).input_params
return formatted_output

```