Proc. of the European Safety and Reliability Conference (ESREL 2026)
Edited by José C. Matos, Paulo B. Lourenço, Michael Beer, Daniel Oliveira,
Edoardo Patelli, Hélder S. Sousa, Rui A. Silva and Monica Santamaria

doi: 10.3850/ESREL2026061419_esrel26-p26397-cd

AI-based Requirement Processing to Support Validation of Future Railway Mobile Communication

Arianna Nocente
University of Pisa, Italy. E-mail: a.nocente@studenti.unipi.it

Riccardo Risi
Independent Researcher, Italy. E-mail: riccardorisi10@gmail.com

Francesco Flammini
University of Applied Sciences and Arts of Southern Switzerland, Switzerland, and University of Florence, Italy. E-mail: francesco.flammini@supsi.ch, francesco.flammini@unifi.it

Pasquale Donadio
Comesvil S.p.A, Villaricca, Italy. E-mail: pasquale.donadio@comesvil.com

Roberto Canonico, and Valeria Vittorini
University of Naples Federico II, Italy. E-mail: name.surname@unina.it

The Future Railway Mobile Communication System (FRMCS) represents the next-generation communication standard for European railways, aiming at ensuring interoperability, safety, and digital resilience across onboard and trackside systems. This research presents the FRMCS Requirement Processing System, an AI-based framework that has been developed to automate the extraction and structuring of technical requirements from complex and multi-source specifications. The first step is the faithful extraction of the requirement along with its important information, which includes the type of requirement and the requirement identifier. Subsequently, each requirement is processed and classified using a graph-based approach that takes contextual elements into account. Using the RAG-Anything framework, a comprehensive knowledge graph is built from multimodal document inputs (i.e., text, tables, and figures). The resulting architecture of the framework supports hybrid query modes that combine vector similarity search and graph traversal, enabling the model to retrieve contextual information across specification layers before requirement classification. Output data are generated as structured files containing FRMCS component and function metadata. System evaluation was conducted with human supervision of the output of the framework to ensure a reliable comparison of requirement identification and metadata extraction results. The framework was developed after an initial exploratory phase based on a human-in-the-loop extraction approach utilizing ChatGPT, a conversational chatbot based on a large language model (LLM). This work highlights a novel synergy between AI, prompt engineering, and systems validation, offering a replicable blueprint for the development of future standards such as FRMCS. The approach proved very useful in industrial practice in order to support the design of specific labs for FRMCS validation.

Keywords: software engineering, design, train control systems, verification and validation, testing, generative AI.

1. Introduction

In many countries all over the world, railway communications are based on the *Global System for Mobile Communications for Railways* (GSM-R) technology, which was established between 1995 and 2000 as part of the *European Rail Traffic Management System* (ERTMS) standard. After almost thirty years of successful deployment, the GSM-R technology has shown limitations and weaknesses. End of support for GSM-R is expected around 2035. To prepare the ground for a new communications standard for railways, in 2013 the *International Union of Railways* (UIC) launched an initiative aimed at defining and standardizing the *Future Rail Mobile Communications System* (FRMCS). Over the last decade, UIC has established a solid organization capable of building the FRMCS specifications, which form the basis for early commercial implementations and field trials. Operational testing is the objective of MORANE-2 (*MObile radio for RAilway Networks in Europe*)[a], a European project involving manufacturers and national railway operators.

Requirements engineering is safety-critical in the railway domain: EN 50126-1 positions the specification and demonstration of RAMS requirements as a cornerstone of the railway life cycle, because these requirements drive downstream verification and validation activities British Standards Institution (2017). More broadly, requirements-quality research highlights that low-quality requirements can propagate defects into later life-cycle stages, where they become increasingly costly to detect and correct, reinforcing the need for systematic and automatable requirement quality assurance when working with complex railway standards Frattini et al. (2023).

Compared with prior automated approaches that mainly perform requirement extraction as sentence-level detection/classification over standard text Luttmer et al. (2023) and with LLM-based RE assistance that still faces recurring challenges in context coverage, reliability, and validation Norheim et al. (2024); Ferrari and Spoletini (2025), this paper contributes an end-to-end FRMCS Requirement Processing System (FRMCS-RPS) that (i) faithfully extracts requirements together with identifiers and types and (ii) enables context-aware classification via a multimodal knowledge graph and hybrid retrieval (vector similarity + graph traversal) built from heterogeneous specification artifacts Guo et al. (2025). FRMCS-RPS structured FRMCS component/function metadata, effectively supporting reproducible validation-lab design for FRMCS, while a human-in-the-loop process based on a ChatGPT-assisted baseline has been found to be time-consuming and error-prone.

2. FRMCS standard and motivation

The *Future Railway Mobile Communication System* (FRMCS) is standardized by UIC in a process that also involves ETSI, the *European Telecommunications Standards Institute*, with its *Technical Committee on Rail Telecommunications* (TC RT). In early 2020, UIC decided to align the new communication system to the 3GPP's Fifth Generation architecture (5G) and a *FRMCS User Requirements Specification* (URS) was published (UIC FU-7100 (2020)) with the purpose of defining a set of technology-independent user requirements. Such user requirements were later translated into functional and system requirements organized in a set of different documents, undergoing a continuous revision process. The most relevant specification documents are the following:

- FRMCS FRS - Functional Requirements Specification (UIC FU-7120 (2025))
- FRMCS TOBA FRS - On-Board Functional Requirements Specification (UIC TOBA-7510 (2025))
- FRMCS SRS - System Requirements Specification (UIC AT-7800 (2025))
- FRMCS FFFIS - Form Fit Functional Interface Specification (UIC FFFIS-7950 (2025))
- FRMCS FIS - Functional Interface Specification (UIC FIS-7970 (2025))

A first set of UIC FRMCS Specifications (V1) was produced and published at the end of 2023. A second version of the specifications was published in December 2024. An update version (v2.1.0) was issued in April 2025. MORANE-2 is conducting large-scale testing of V2 specifications since 2025 and gathering feedback to let UIC finalize a third version of FRMCS specs, which is expected to be released in 2027.

Figure 1 shows the organization of the document corpus for the FRMCS specifications. It shows that UIC specifications also refer to documents issued by other institutions, such as ETSI and 3GPP. Moreover, FRMCS specs take into account the requirements dictated by railway systems that will communicate through FRMCS, including the *European Train Control System* (ETCS) as well as Mission Critical voice communication systems.

2.1. *Requirements categorization*

All UIC FRMCS specifications are PDF documents structured in sections and subsections. Sec-

[a]https://cordis.europa.eu/project/id/101196125

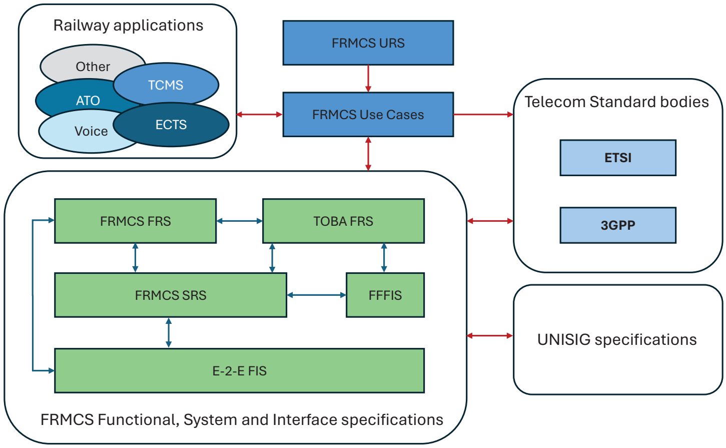

Fig. 1. FRMCS Specifications from UIC AT-7800 (2025)

tions (and subsections) contain individual sentences (*clauses*), numbered as sub-sub-sections, each marked at the end with a marker, as follows.

(a) **Mandatory for the System** (indicated by '(M)' at the end of the clause). These clauses refer to a condition that must be met without exception in order to deliver a system ensuring the fulfillment of essential functional and system needs, compliance to relevant standards and technical integration. The mandatory requirements are identified as sentences using the keyword "shall".
(b) **Optional for the System** (indicated by '(O)' at the end of the clause). These clauses refer to requirements that may be used based on the implementers' choice. When an optional requirement is selected, the related requirement(s) of this specification becomes mandatory for the system. The optional requirements are identified as sentences using the keyword "should".
(c) **Information** (indicated by '(I)' at the end of the clause). These statements provide additional information to help the reader understanding a requirement.

FRMCS specifications, in their current version include some clauses, marked as (M-V3), (O-V3) or (I-V3), that refer to requirements planned to become effective in the third specification version on which the first commercial implementations will be based. Indications (M-Vx), (O-Vx) and (I-Vx) are also used for clauses for later versions of the specification. These clauses are kept in the specification only for readability and consistency purposes.

2.2. *Issues and motivation*

Currently, no complete and interoperable commercial FRMCS solutions are available. For establishing field trials and private testbeds it is crucial for companies prioritize the requirements dictated by FRMCS specs, so that these systems can be engineered and these companies may gain the necessary expertise to deliver FRMCS solutions to railway operators.

This work describes the framework we developed to analyze FRMCS specs, extract mandatory requirements, and associate them to system functions. Those are enablers in the development of innovative systems and products, whose implementation and commercialization depend on the evolution of the relevant standards. The dichotomous treatment of mandatory and optional requirements makes it possible to obtain a system version that is compliant with the standard while being simpler to implement, significantly reducing the time-to-market. The FRMCS-RPS framework is currently used by Comesvil S.p.A. to design and implement an FRMCS v2-compliant private testbed.

3. Baseline LLM Approaches

The analysis of FRMCS specifications poses significant challenges due to their size, heterogeneity, and continuous evolution across versions. Although the documents follow a well-defined hierarchical structure and clearly label requirements as mandatory, optional, or informative, they also exhibit formatting inconsistencies, extensive cross-referencing, and extensive use of architectural information encoded in tables and figures.

In recent practice, Large Language Models (LLMs) have increasingly been adopted as general-purpose tools to support requirement extraction and interpretation from technical documentation. While LLMs demonstrate strong linguistic and summarization capabilities, their direct application to safety-critical systems such as FRMCS raises substantial concerns Flammini et al. (2026). When used without explicit structural constraints, LLMs tend to exhibit uncontrolled generative behaviour, including paraphrasing of normative clauses, omission of requirements, and the generation of plausible but non-existent statements that are stylistically consistent with the source documents.

3.1. *Baseline GPT-Based Processing*

To assess these limitations, a baseline approach based on direct interaction with GPT-based models was initially investigated. Several iterative experiments were conducted by varying prompt formulations, temperature settings, and input strategies, including both exact text reproduction and semantic classification of copied requirement clauses in order to avoid the generation of incorrect extracted text that, while coherent with the surrounding context, appeared highly credible despite not being faithful to the original specification. Although acceptable results were obtained for isolated requirements, systematic issues emerged when scaling the approach to entire specification chapters. Minor human-induced formatting style or defects in the source document, such as line breaks, misplaced labels, or nested bullet lists were sufficient to trigger inconsistent model behavior. In particular, the models frequently rewrote normative text, skipped requirements, or produced outputs that could not be reliably traced back to specific clauses in the specifications.

From a computational perspective, the baseline approach also proved inefficient. Sequential processing of requirements required several hours of work by a dedicated unit of personnel for a single document.

These observations highlighted a critical limitation of unconstrained LLM-based workflows: despite their expressive power, they lack intrinsic awareness of the formal structure, architectural semantics, and validation requirements inherent to FRMCS specifications. This motivated the development of a structured and auditable processing framework, in which requirement extraction and candidate identification are governed by explicit rules and graph-based reasoning, and the role of the LLM is deliberately confined to a final, and constrained classification step.

4. Proposed Methodology

We describe the two-phase methodology underlying the FRMCS Requirement Classification System designed to decouple document understanding from requirement classification. This separation allows FRMCS specifications to be processed once, and subsequently queried multiple times without re-ingestion, ensuring scalability, consistency, and traceability.

Phase 1 focuses on transforming multimodal PDF specifications into a persistent knowledge graph. Phase 2 leverages the knowledge graph to classify individual requirements through a multi-strategy retrieval process followed by graph-guided LLM reasoning.

4.1. *Phase 1: Document Ingestion and Graph Construction*

Figure 2 illustrates the logical architecture of the multimodal ingestion functionality of the framework. The objective of Phase1 is to transform FRMCS specification documents from unstructured PDF format into a structured, queryable knowledge graph. A layout-aware PDF parser is employed to extract textual content, tables, and images while preserving document hierarchy and

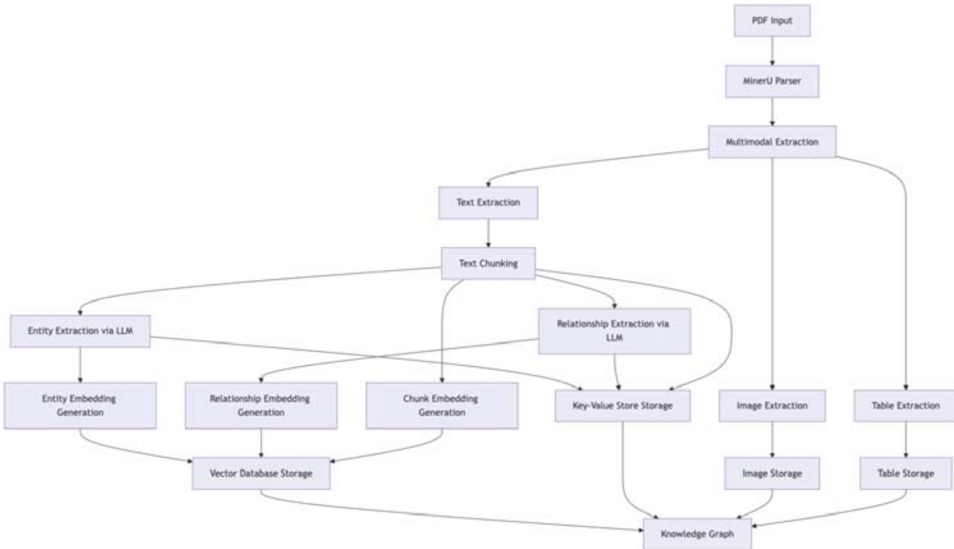

Fig. 2. Phase 1: Architecture for multimodal document ingestion and knowledge graph construction from FRMCS specifications.

page-level references.

Textual content is segmented using page-based, context-aware chunking, ensuring that each chunk maintains semantic coherence with its surrounding sections, headers, and captions. Entity extraction is performed using an LLM-assisted approach to identify FRMCS-relevant concepts, including functional blocks, components, interfaces, and services. In parallel, semantic relationships between entities are extracted, capturing associations such as component–function mappings and architectural dependencies. For each extracted entity, relationship, and text chunk, dense vector embeddings are generated and stored in a vector database to support semantic similarity search. In parallel, complete metadata are stored in key–value repositories to guarantee deterministic retrieval and auditability, which is a key requirement for requirement extraction in safety-related domains Arianna et al. (2024). Multimodal artifacts such as diagrams and tables are stored separately and linked to the corresponding document pages and semantic entities.

4.2. *Phase 2: Graph-Guided Requirement Classification*

Figure 4 depicts the workflow adopted for requirement classification. A multi-strategy search phase is first executed to identify candidate entities within the knowledge graph. Candidate entities are identified through a multi-strategy retrieval process that combines complementary search mechanisms. Direct name search performs exact and fuzzy matching against normalized entity names to capture explicit or near-explicit textual references within the requirement. Semantic vector search complements this step by retrieving conceptually related entities based on embedding similarity, enabling the identification of relevant components even when they are not explicitly mentioned. Finally, chunk-based search exploits the structural organization of the specifications by retrieving entities occurring in document sections located in close proximity to the requirement page, thereby leveraging contextual locality.

Candidates from all strategies are merged, deduplicated, and ranked according to a unified scoring function that combines similarity scores,

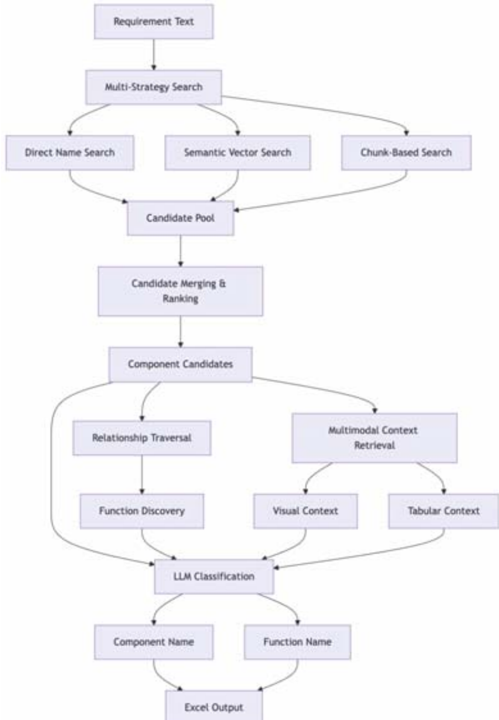

Fig. 3. Phase 2: Workflow for graph-guided requirement classification and Excel-based output generation.

match type, and entity frequency within the graph. The highest-ranked component candidates are then used to traverse the knowledge graph through single-hop relationships to discover associated functions. In parallel, multimodal context retrieval is performed to collect relevant diagrams and tables that may provide additional architectural or functional information. A graph guided LLM classification stage receives as input the ranked candidates, relationship-derived functions, and multimodal context. The LLM is constrained by structured prompts and predefined output schemas, ensuring that results are limited to valid FRMCS components and functions or explicitly marked as undefined when confidence is insufficient.

The final output is generated as an enriched Excel file containing, for each requirement, the identified component and function, while detailed metadata are stored in structured JSON files for traceability and validation.

5. Knowledge Graph Model

The knowledge graph represents the central semantic representation of FRMCS-RCS. It provides a structured and persistent abstraction of the FRMCS specification corpus, enabling controlled reasoning over system architecture, functional relationships, and document context.

The graph is composed of four tightly integrated elements: entity nodes, relationship edges, contextual text chunks, and explicit links to multimodal artifacts such as tables and architectural figures. Together, these elements transform the FRMCS specifications from static documents into an operational semantic model that can be queried.

Entity nodes represent FRMCS-relevant concepts, including functional blocks, system components, interfaces, services, and architectural reference points. Entities are extracted from textual and tabular content using LLM-assisted recognition and are subsequently normalized and deduplicated to ensure consistency across the document corpus. Each entity is associated with a canonical normalized name, possible lexical variants, frequency information, vector embeddings for semantic similarity search, and references to the originating text chunks and document pages.

Relations edges encode semantic associations between entities as they emerge from the specifications. These associations include component-function dependencies, functional containment, and architectural interactions. Relationships are stored explicitly as pairs of linked entities, enabling graph traversal during requirement classification. This explicit relational structure allows the system to infer function candidates from component candidates even when such functions are not directly mentioned in the requirement text.

Contextual text chunks anchor the content to the original document structure, preserving page- and section-level references as well as surrounding headers and captions to ensure traceability. Contextual grounding also enables proximity-based reasoning, exploiting the fact that requirements are often semantically related to entities defined in nearby sections of the document.

In addition to textual content, the knowledge graph maintains explicit links to multimodal artifacts extracted from the specifications, including tables and figures. Tables frequently encode parameter constraints and functional mappings, while figures convey architectural relationships that are not fully expressed in text. By associating multimodal artifacts with entities and text chunks, the system is able to retrieve visual and tabular context to support disambiguation during requirement classification.

The knowledge graph adopts a hybrid symbolic-semantic representation. Symbolic graph topology enables rule-driven reasoning and explicit relationship traversal, while vector embeddings provide robustness to linguistic variability and conceptual similarity. This combination overcomes the limitations of purely vector-based retrieval approaches, which lack explicit relational semantics and contextual anchoring, and purely symbolic approaches, which are brittle in the presence of terminological variation. As a result, the knowledge graph enables graph-guided LLM reasoning while constraining uncontrolled generative behaviour and preserving auditability.

6. Operational Effort

The proposed framework has been implemented and applied to the analysis of the FRMCS specifications, version V2, specifically to the five documents reported in Section 2. Figure 4 shows an excerpt of the resulting requirement matrix in Excel format. Table 1 highlights the operational effort required by the two processing phases introduced in Section 4. The reported values correspond to real executions on the PDF documents.

Phase1 exhibits a growth trend mainly related to the number of pages actually processed to extract structural, semantic, and multimodal information to be persistently stored in the knowledge graph.

Phase2 shows an increasing effort as the number of requirements grows. This reflects the fact that each requirement triggers a multi-strategy retrieval process over an expanding knowledge graph, followed by graph-guided reasoning and constrained LLM-based classification.

7. Discussion and conclusion

Unconstrained use of general-purpose LLMs is unsuitable for safety-critical standard analysis. Without structural guidance, LLMs exhibit high hallucination rates, uncontrolled paraphrasing, and omission of requirements. By contrast, the proposed graph-guided approach constrains LLM reasoning within a constrained pipeline, significantly reducing semantic drift while preserving flexibility. The two-phase approach enables efficient re-use of ingested specifications and supports scalable analysis across evolving document versions.

From an industrial perspective, the system has proven effective in supporting the design of FRMCS-compliant testbeds and validation laboratories. More broadly, the methodology is replicable for other complex standards characterized by layered architectures, multimodal documentation, and long-term evolution.

The results highlight a fundamental principle: in safety-critical domains, LLMs must be treated

reqID	type	text	page	component_name	subfunc/refpoint
6.1.1.5	M	In order to interconnect two FRMCS Domains, the FRMCS System shall support this interconnection using FSNNI reference point (see 6.4.4.2).	31	5G Core and Access Network (Transport Layer)	FSNNI
6.1.1.6	M	In order to interwork with GSM-R, the FRMCS System shall support this interworking using FSIWF reference point (see 6.4.4.1).	31	GSM-R systems (via IWF)	FSIWF
6.1.1.7	O-V3	The FRMCS System should support interworking with non-GSM-R / non-FRMCS systems using FSONI reference point (see 6.4.4.3).	31	FRMCS Trackside Gateway (Service Layer)	FSONI
6.1.1.9	M	An FRMCS Domain shall encompass the minimum necessary components: a. In the Transport Domain: a. A 5G Core Network b. A 5G Access Network supporting the possible spectrum options as defined in section [8], b. In the Service Domain: a. A MCX infrastructure, including a SIP Core.	31	MCX Infrastructure (Service Layer)	SIP Core

Fig. 4. Excerpt from the generated SRS requirement matrix

Document	# Pages	# Req.	Ph1 (s)	Ph2 (s)	Ph1 (h)	Ph2 (h)
FRMCS TOBA (UIC TOBA-7510 (2025))	52	161	10010	1923	2.78	0.53
FRMCS FFFIS (UIC FFFIS-7950 (2025))	55	203	7876	4576	2.19	1.27
FRMCS FIS (UIC FIS-7970 (2025))	89	322	18797	5116	5.22	1.42
FRMCS SRS (UIC AT-7800 (2025))	152	662	26336	10759	7.32	2.99
FRMCS FRS (UIC FU-7120 (2025))	224	1602	34952	25451	9.71	7.07

as reasoning components embedded within engineered systems, rather than as autonomous analytical tools.

Acknowledgement

Work of authors affiliated to University of Napoli Federico II and Comesvil S.p.A. was partly funded by the Italian Ministry of Enterprises and Made in Italy (MIMIT) as Project no. F/350285/01-02/X60 (FRMCS Lab). The work of Francesco Flammini was partly supported by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contracts no. 23.00321 (Academics4Rail) and 24.00528 (PhDs EU-Rail). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the funding agencies, which cannot be held responsible for them.

References

Arianna, N., R. Riccardo, P. Gabriele, and R. Giulio (2024). From unstructured documents to annotated information: An optimized pipeline to process industrial requirements. In *Proceedings of the 2024 8th International Conference on Natural Language Processing and Information Retrieval*, pp. 272–278.

British Standards Institution (2017). Railway applications — the specification and demonstration of reliability, availability, maintainability and safety (rams) — part 1: Generic rams process. Technical Report BS EN 50126-1:2017, BSI Standards Limited.

Ferrari, A. and P. Spoletini (2025, May). Formal requirements engineering and large language models: A two-way roadmap. *Information and Software Technology 181*, 107697.

Flammini, F., A. Nocente, C. Bernardeschi, and V. Vittorini (2026). Opportunities and risks of Generative AI in model-based engineering of railway systems. In M. H. ter Beek, S. Gnesi, A. E. Haxthausen, and L. Semini (Eds.), *Journeys Between Formal Methods and the Railway Industry*, Volume 16470 of *LNCS*. Cham: Springer. Forthcoming (book due March 2026).

Frattini, J., L. Montgomery, J. Fischbach, D. Mendez, D. Fucci, and M. Unterkalmsteiner (2023). Requirements quality research: A harmonized theory, evaluation, and roadmap. *Requir. Eng. 28*, 507–520.

Guo, Z., X. Ren, L. Xu, J. Zhang, and C. Huang (2025). Rag-anything: All-in-one rag framework.

Luttmer, J., V. Prihodko, D. Ehring, and A. Nagarajah (2023). Requirements extraction from engineering standards — systematic evaluation of extraction techniques. *Procedia CIRP 119*, 794–799.

Norheim, J. J., E. Rebentisch, D. Xiao, L. Draeger, A. Kerbrat, and O. L. de Weck (2024). Challenges in applying large language models to requirements engineering tasks. *Design Science 10*, e16.

UIC AT-7800 (2025, April). FRMCS System Requirements Specification (v2.1.0). Tech. specification, International Union of Railways.

UIC FFFIS-7950 (2025, April). FRMCS Form-Fit Functional Interface Specification (v2.1.0). Tech. specification, International Union of Railways.

UIC FIS-7970 (2025, April). FRMCS Functional Interface Specification (v2.1.0). Tech. specification, International Union of Railways.

UIC FU-7100 (2020, February). FRMCS User Requirements Specification. Tech. specification, International Union of Railways.

UIC FU-7120 (2025, April). FRMCS Functional Requirements Specification (v2.1.0). Tech. specification, International Union of Railways.

UIC TOBA-7510 (2025, April). On-Board FRMCS Functional Requirements Specification (v2.1.0). Tech. specification, International Union of Railways.