Revisión Sistemática de la Literatura: aplicación de seguridad a requerimientos software de sistemas críticos ferroviarios

Cristian Pinto Luft, Emanuel Irrazabal, Iván Sambrana

Grupo de Investigación en Innovación de Software y Sistemas Computacionales
Departamento de Informática.- FACENA UNNE
Cristianpl777@gmail.com, emanuelirrazabal@gmail.com, sambranaivan@gmail.com

Abstract. Entre los sistemas críticos más reconocidos se encuentran los sistemas ferroviarios, ya que un fallo en los mismos puede generar daños económicos, ambientales o a la vida de las personas, por lo que su análisis y tratamiento cobra especial relevancia, sobre todo en cuanto a la seguridad. En esta revisión sistemática se hace un análisis exhaustivo de la información existente en cuanto a la gestión de requerimientos software en sistemas críticos ferroviarios con la finalidad de conocer sus principales características de implementación. La revisión sistemática se realizó sobre cuatro repositorios académicos distintos (ACM, Science Direct, Springer e IEEE), obteniéndose un total de 23 publicaciones, las cuales han sido analizadas para obtener información categorizada en 5 dimensiones distintas para ayudar a su comprensión. Como conclusión, se informan los resultados de la aplicación de dicho procedimiento, indicando los principales hallazgos obtenidos de este análisis.

Keywords: RSL, safety-critical, safety, software requirement, railway, EN-50128, IEC-61508

1 Introducción

Las Revisiones Sistemáticas de la Literatura (RSL) son herramientas fundamentales para la búsqueda, recolección, procesamiento y publicación del estado del arte de una temática particular, siguiendo una metodología sistémica, completa, explícita y reproducible, que puede ser utilizada con diversos fines. En lo que a esta RSL respecta, su foco principal se centra en investigar y apropiarse de los conocimientos más actualizados y relevantes relativos a la gestión de los requerimientos software de los sistemas críticos ferroviarios, centrándose principalmente en la seguridad, uno de los atributos de calidad más importantes en estos tipos de sistemas, ya que de ellos pueden depender en gran medida la viabilidad económica y funcional del sistema, la conservación de su entorno e incluso la preservación de vidas humanas.

Si bien en el contexto de la Ingeniería de Requerimientos Software para sistemas críticos se puede encontrar gran cantidad de material de investigación y aplicación [1][2][3][4][5], por la misma naturaleza crítica de estos sistemas, es importante conocer

puntualmente el estado del arte para cada dominio antes de comenzar a definir un nuevo procedimiento de gestión. Esto es determinante en la seguridad (y en los aspectos RAMS en sí) del sistema software. Dichas particularidades incluyen aspectos como ser normativas y estándares [6][7], técnicas de análisis y aseguramiento de la seguridad [8], metodologías de análisis [9], herramientas de soporte software [10], entre otras. En cada dominio de aplicación en particular, estos aspectos son definidos empíricamente mediante el uso y aprendizaje histórico. Lo que es útil en un tipo dominio de aplicación (como por ejemplo, en los sistemas aeroespaciales), puede no serlo en otros (como por ejemplo, en los sistemas ferroviarios), dadas las particularidades de cada dominio en sí [6].

Un aspecto importante a considerar, es que al tratarse de sistemas críticos, de no analizar los sucesos históricos se pueden llegar a producir inconsistencias y llevar al sistema a posibles ocurrencias de fallas. Esta gestión de la seguridad de la que se debe dotar a los sistemas críticos es una de sus principales diferencias con otro tipo de sistemas, en donde la criticidad de esta dimensión es menor.

El objetivo principal de la presente revisión es entonces establecer el estado del arte y marco teórico/práctico de la temática definida, con la finalidad de apoyar a la generación de un procedimiento general de gestión de requisitos software, cumpliendo con las buenas prácticas internacionales entre las que se pueden destacar la UNE-EN 50128:2012 o IEC 62279.

2 Trabajos relacionados

Anteriormente a la confección de la presente RSL, y conjuntamente con el estudio de diversos artículos de variada índole, se analizó el siguiente trabajo: en [A26] se presenta una RSL sobre la integración de la ingeniería de requerimientos y las técnicas de análisis de seguridad existentes. El objetivo principal de este trabajo es analizar los distintos enfoques de integración entre ambas metodologías utilizados en el mundo, con sus distintas variantes en cuanto a aplicabilidad.

En la RSL citada, se hace hincapié en la importancia de aplicar técnicas de análisis de seguridad en etapas tempranas del diseño de un sistema, integrándola a la gestión de sus requerimientos, para intentar disminuir la cantidad de errores posibles durante las siguientes etapas de su ciclo de vida. A diferencia de la anterior, la presente RSL intenta analizar específicamente la gestión de requerimientos software en sistemas críticos, centrándose principalmente en aquellos del tipo ferroviario y sus aspectos relativos a la seguridad, enfocándose en casos de aplicaciones exitosas de estos procedimientos.

3 Planificación de la RSL

El objetivo de esta sección es definir la metodología de trabajo a utilizarse para la realización de la presente RSL, especificando cada una de las etapas que la componen, y siguiendo el modelo propuesto por Barbara Kitchenham [11].

3.1 Elección de preguntas de investigación

En la Tabla 1 se definieron las preguntas de investigación (PI) a formularse para llevar a cabo la búsqueda de información relevante existente acerca de sistemas y metodologías de gestión de los requerimientos software en sistemas críticos ferroviarios, haciendo foco en la seguridad. Para puntualizar las respuestas a las interrogantes, se definieron para cada una de ellas una dimensión y un conjunto de atributos que corresponden a los aspectos más significativos a indagar en la construcción de la RSL.

Tabla 1. Preguntas de Investigación

PI	Descripción	Dimensiones
PI-1	¿Qué tipos o módulos de sistemas software se implementan siguiendo metodologías de gestión de requerimientos software en sistemas críticos ferroviarios?	**Sistemas**: implementación, módulo, sistema, sub-sistema.
PI-2	¿Qué metodologías se utilizan para la gestión de requerimientos software en sistemas críticos ferroviarios?	**Metodologías usadas**: métodos ágiles, formales, estructurados, mixtos.
PI-3	¿Qué software o herramientas se utilizan y cómo se logra la integración de las mismas en la gestión de requerimientos software en sistemas críticos ferroviarios?	**Aplicaciones**: software, manual, integración, modelo.
PI-4	¿Cómo se gestiona la seguridad de los sistemas de gestión de requerimientos software en sistemas críticos ferroviarios?	**Seguridad**: amenazas, peligros, fallas, nivel de seguridad.
PI-5	¿Bajo qué normativas se implementan sistemas de gestión de requerimientos software en sistemas críticos ferroviarios?	**Normativas:** normativa, estándar, buenas prácticas, ley.

3.2 Formulación de cadenas de búsqueda

Para generar las cadenas de búsqueda, se definieron en la Tabla 2 las palabras claves y un conjunto de palabras relacionadas a cada una de ellas.

Tabla 2. Palabras clave

Palabras clave	Palabras relacionadas
Software	Application, firmware, program
Requirement	Safety-critical software requirement, requisite, SRS
Safety	Hazard, failure, fault, threat, risk, security level, SIL, SSIL, RAMS
Railway	Train, rails, metro, monorail, subway, tracks, 50128, 62279, 61508

Luego de haber definido las palabras claves y sus relaciones, se las unieron usando los conectores lógicos AND y OR, obteniendo la siguiente cadena de búsqueda:

(software OR application OR firmware OR program) AND (requirement OR "safety-critical software requirement" OR requisite OR SRS) AND (safety OR hazard OR failure OR fault OR threat OR risk OR "security level" OR SIL OR SSIL OR RAMS) AND (railway OR train OR rails OR metro OR monorail OR subway OR tracks OR 50128 OR 62279 OR 61508)

Como aclaración: la cadena de búsqueda tuvo que ser refinada para los distintos repositorios, debido a las restricciones que poseen, quedando:

- **IEEE** (15 términos como máximo): (software OR application) AND (requirement OR "safety-critical software requirement" OR requisite OR SRS) AND (safety OR hazard OR fault OR SIL OR RAMS) AND (railway OR 50128 OR 62279 OR 61508)
- **Science Direct** (250 caracteres como máximo): (software OR application OR program OR firmware) AND (requirement OR "safety-critical software requirement" OR requisite) AND (safety OR hazard OR fault OR risk OR SIL OR SSIL OR RAMS) AND (railway OR train OR rails OR 50128 OR 62279 OR 61508)
- **Springer** (computer science and software engineering): (software OR application OR firmware) AND (requirement OR "safety-critical software requirement" OR requisite OR SRS) AND (safety OR hazard OR failure OR fault OR "security level" OR SIL OR SSIL OR RAMS) AND (railway OR subway OR 50128 OR 62279 OR 61508)

3.3 Selección de fuentes de búsqueda

La búsqueda de información fue realizada sobre las fuentes electrónicas que se mencionan a continuación, usando las siguientes funcionalidades/secciones:

- Science Direct (advanced search - title, abstract, keywords)
- ACM (advanced search)
- Springer
- IEEE (command search - metadata only - Metadata Includes the abstract, index terms, and bibliographic citation data (such as document title, publication title, author, etc.)).

3.4 Determinación de criterios de selección

Los criterios de inclusión de la información encontrada en la confección de la RSL fueron: artículos relacionados con la gestión de requerimientos software en sistemas críticos ferroviarios, en inglés y publicados en congresos, workshops, revistas, libros y/o capítulos.

4 Ejecución de la RSL

Para llevar a cabo la presente RSL, por cada fuente de búsqueda definida se llevaron a cabo los siguientes pasos, con tal de obtener los estudios primarios que se analizaron:

1. Realizar la búsqueda utilizando la cadena de búsqueda definida.
2. En los artículos resultantes, decidir cuáles incluir y cuáles excluir leyendo el título y el abstract.
3. De los artículos resultantes del paso anterior, decidir cuáles incluir y cuáles excluir leyendo el texto completo.
4. Los artículos resultantes fueron indicados y clasificados de acuerdo a las dimensiones definidas en el Anexo y numeradas con referencias desde A1 hasta A23. En la Tabla 3 se indica la distribución de los artículos primarios incluidos de acuerdo a sus fuentes.

Table 3. Distribución de artículos

Fuente	Artículos encontrados	Filtrados por título	Filtrados por abstract	Filtrados por texto	Porcentaje por fuente
Science Direct	3196	17	8	1	4
ACM	506	9	3	3	13
Springer	5039	38	19	11	48
IEEE	368	20	13	8	35
Total	**9109**	**84**	**43**	**23**	**100%**

Además de los artículos encontrados en la búsqueda definida, se analizaron 3 más de especial importancia para el estado del arte, por lo que se los incluye en la presente RSL. Los mismos son:

- CENELEC 50128 and IEC 62279 Standards [A24]
- NASA Software Safety Guidebook [A25]
- Integration between Requirements Engineering and Safety Analysis: A Systematic Literature Review [A26]

5 Reporte de resultados

En la presente sección se muestran los resultados obtenidos del análisis de los estudios primarios obtenidos, teniendo en cuenta las distintas PI a las que responden y se referencian los artículos en los que se encuentran.

5.1 PI-1 - ¿Qué tipos o módulos de sistemas software se implementan siguiendo metodologías de gestión de requerimientos software en sistemas críticos ferroviarios?

En la Fig. 1 se muestra la distribución de aplicación de gestión de requerimientos software a distintos sistemas, módulos o subsistemas ferroviarios.

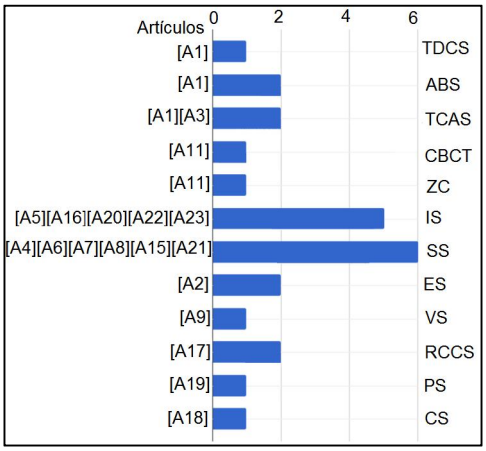

Fig. 1. Distribución por tipo de sistema

Se puede observar la prevalencia de integración de métodos de ingeniería del software principalmente en la definición de sistema de señalamiento (SS) y de interbloqueo (IS). Se hace mayor énfasis en los SS y en los IS porque son sistemas sumamente complejos, integran hardware y software, y generalmente son de tipo SIL 4.

5.2 PI-2 - ¿Qué metodologías se utilizan para la gestión de requerimientos software en sistemas críticos ferroviarios?

Los principales tipos de metodologías utilizadas fueron: los métodos formales (53.1%), métodos estructurados (34.4%) y métodos mixtos (12.5%). Cabe apreciar que, a pesar de su auge en la actualidad, no se encontraron metodologías ágiles para esto, y prevalecieron los métodos formales por sobre los demás.
De los métodos formales, los más mencionados fueron: métodos integrados, método formal B (lenguaje B), método formal Z (lenguaje Z), entre otros, como se puede observar en la Fig. 2. En cuanto a los primeros, se clasificó en esta categoría a aquellos que se valen del producto software con el que se implementan para definir las especificaciones de los requerimientos software. Se puede apreciar una prevalencia de estos, ya que las herramientas que los soportan ayudan a gestionar la complejidad del uso de dichos métodos, a la vez que brindan otras funcionalidades, como ser la generación automática de diagramas y código, que aportan al análisis de los requerimientos y su documentación de manera automatizada y organizada.

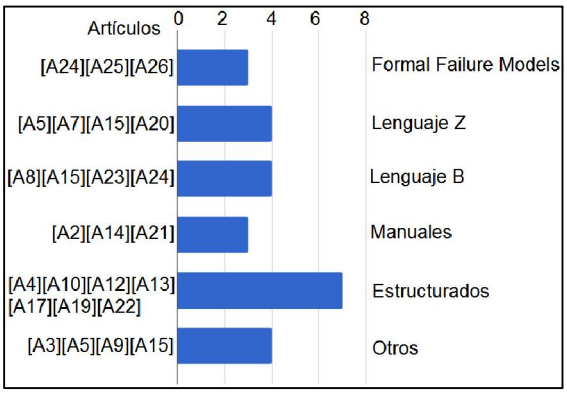

Fig. 2. Métodos formales usados

Cabe destacar que la implementación de métodos formales se dio casi siempre de manera parcial, integrando otras metodologías y herramientas a las mismas, debido a que de esta forma se logra una comprensión más completa del problema, y su uso es altamente recomendado por los distintos estándares a cumplir (como el EN-50128).

5.3 PI-3 - ¿Qué software o herramientas se utilizan y cómo se logra la integración de las mismas en la gestión de requerimientos software en sistemas críticos ferroviarios?

Los principales tipos de herramientas utilizadas fueron: software (52,6%), modelos (26,3%) e integración de herramientas varias (21,1%). Cabe resaltar que no se encontró ninguna propuesta que haga referencia a métodos manuales de gestión, es decir que se tiende hacia una gestión automatizada por medio de herramientas software especialmente diseñadas para estos fines. La más utilizada fue SCADE (usada en 4 trabajos), la misma representa una amplia suite de gestión de software de seguridad crítica, que integra distintas aspectos de la ingeniería del software de este tipo de sistemas. Se encontró que, dada la complejidad de gestión de este tipo de requerimientos, sin la utilización de herramientas software que ayuden a automatizar los procesos, esta tarea sería sumamente compleja de ser llevada a cabo.

5.4 PI-4 - ¿Cómo se gestiona la seguridad de los sistemas de gestión de requerimientos software en sistemas críticos ferroviarios?

Las técnicas de gestión de seguridad más mencionadas fueron: Software Failure Tree Analysis (SFTA), Software Failure Modes and Effect Analysis (SFMEA), Preliminary Hazard Analysis (PHA), Hazard and Operability study (HAZOP), Software Effect and Criticality Analysis (SFMECA), entre otras menos frecuentes (y algunas relativas a la seguridad del sistema en general), como se puede observar en la Fig. 3.

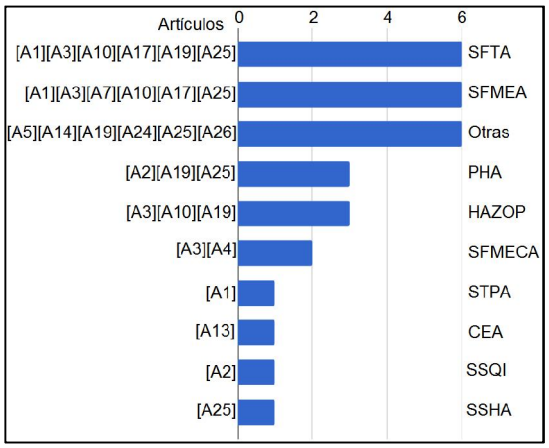

Fig. 3. Técnicas usadas

Se puede apreciar puntualmente el predominio de las técnicas SFTA y SFMEA, debido a que las mismas son técnicas complementarias: las primeras permitiendo descubrir las posibles causas de las fallas del software, y la segunda analizar los efectos, causas, detección, mitigación, prevención y demás características de dichas fallas.

5.5 PI-5 - ¿Bajo qué normativas se implementan sistemas de gestión de requerimientos software en sistemas críticos ferroviarios?

En la Fig. 4 se pueden observar las principales normativas encontradas en los artículos seleccionados. Se destaca la prevalencia de artículos mencionando a la normativa EN-50128, por sobre otras más genéricas, como ser la IEC-61508.

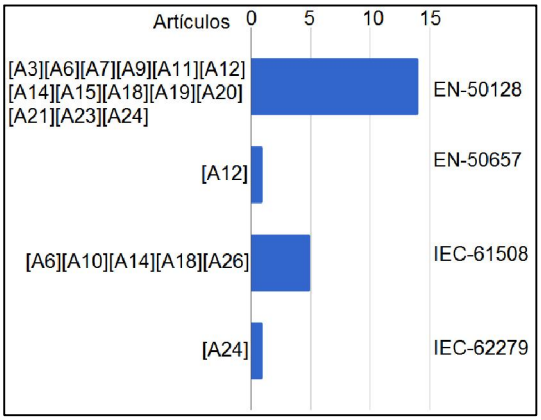

Fig. 4. Normativas para el desarrollo de software crítico ferroviario.

5.6 Resultados adicionales

En la Fig. 5 se puede observar la evolución cronológica de las publicaciones encontradas, mostrándose la cantidad de publicaciones realizadas por año, pudiendo verse una tendencia a la investigación de estos temas en lo que va de los últimos siete años.

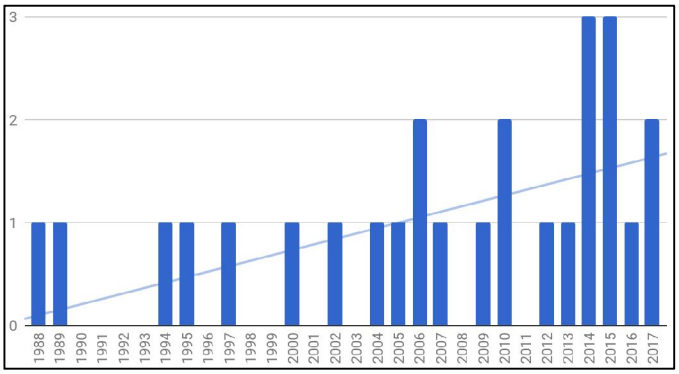

Fig. 5. Distribución por año de publicación.

6 Conclusiones

En este documento se ha realizado una RSL acerca de la aplicación de seguridad a requerimientos software de sistemas críticos ferroviarios en distintos contextos alrededor del mundo. Se han encontrado un total de 23 publicaciones de carácter relevante sobre esta temática (más las otras 3 mencionadas), las cuales han sido clasificadas y analizadas, encontrando como resultados principales los siguientes:

- La mayoría de las publicaciones de esta índole se encuentran en el repositorio de Springer, teniendo en cuenta los 4 utilizados, por lo que se observa que es en donde más se publican cuestiones relativas a la temática tratada.
- Los sistemas software ferroviarios que más se investigan con respecto a sus requerimientos son los de señalamiento e interbloqueo.
- Los métodos más utilizados para la investigación y concreción de este tipo de procedimientos suelen ser métodos formales, con el uso del lenguaje B o Z.
- Muchas publicaciones mencionan al menos una herramienta software para la gestión de este tipo de procedimientos (como SCADE), o la integración de varias con distintas metodologías.
- Las principales técnicas de aseguramiento de la seguridad de los requerimientos software encontradas fueron SFTA y SFMEA, junto con otras que integran el análisis de seguridad del sistema entero
- La mayoría del software crítico ferroviario se desarrolla bajo la norma EN-50128.
- En los últimos siete años se ha incrementado significativamente el estudio de este tipo de problemáticas.

Estos resultados serán utilizados en la confección del estado del arte de la gestión de requerimientos software en sistemas críticos ferroviarios, en el marco del desarrollo de un procedimiento que será utilizado como parte de una tesis de maestría del autor.

Como líneas futuras de mejora, se propone la profundización de los distintos aspectos que componen a la presente RSL, obteniendo mejores indicadores, formas de presentación de la información y resultados adicionales, que lleven a un mayor entendimiento del estado del arte de la temática en cuestión.

Agradecimientos

El financiamiento de este trabajo ha sido realizado a partir del proyecto PI-F17-2017 “Análisis e implementación de tecnologías emergentes en sistemas computacionales de aplicación regional.”, acreditado por la Secretaría de Ciencia y Técnica de la Universidad Nacional del Nordeste (UNNE) para el periodo 2018-2021. Parte de la investigación es desarrollada en el marco de la tesis del maestrando Cristian Pinto Luft, de la Maestría de Tecnologías de la Información Rs. 764/14 CS UNNE.

7 Referencias

1. B. Malone, A. Siraj, “Tracking requirements and threats for secure software development”, ACM-SE 46 Proceedings of the 46th Annual Southeast Regional Conference on XX, Pp. 278-281, Auburn, Alabama, Marzo 28 - 29, 2008, ISBN: 978-1-60558-105-7.
2. H. Hwang and Y. B. Park, "Safety - Critical Software Quality Improvement Using Requirement Analysis," 2017 International Conference on Platform Technology and Service (PlatCon), Busan, 2017, pp. 1-4. doi: 10.1109/PlatCon.2017.7883725.
3. S. Li, S. Duo, “Safety Analysis of Software Requirements: Model and Process”, Procedia Engineering, Vol. 80, Pp. 153-164, 2014, ISSN 1877-7058.
4. A. Saeed, R. de Lemos, T. Anderson, “On the safety analysis of requirements specifications for safety-critical software”, ISA Transactions, Vol. 34, Issue 3, Pp. 283-295, 1995, ISSN 0019-0578.
5. A.P. Ravn, H. Rischel, V. Stavridou, “Provably Correct Safety Critical Software”, IFAC Proceedings Volumes, Vol. 23, Issue 6, Pp. 13-18, 1990, ISSN 1474-6670.
6. R. Shaw, “Safety-critical software and current standards initiatives”, Computer Methods and Programs in Biomedicine, Vol. 44, Issue 1, Pp. 5-22, 1994, ISSN 0169-2607.
7. M. J. Squair, “Issues in the application of software safety standards”, SCS '05 Proceedings of the 10th Australian workshop on Safety critical systems and software - Vol. 55, Pp. 13-26, Sydney, Australia, 2006, ISBN:1-920-68237-6.
8. S. Tiwari, S. S. Rathore, S. Gupta, V. Gogate, A. Gupta, “Analysis of Use Case Requirements Using SFTA and SFMEA Techniques”, ICECCS '12 Proceedings of the 2012 IEEE 17th International Conference on Engineering of Complex Computer Systems, Pp. 29-38, Julio 18 - 20, 2012, ISBN: 978-2-9541-8100-4.
9. P. J. Bryan, "Software safety and dependability for railway control systems," IET 13th Professional Development Course on Electric Traction Systems, London, 2014, pp. 1-21. doi: 10.1049/cp.2014.1445.

10. J. Brummer, M. Kersken, J. Märtz, “Tools for software safety analysis”, Reliability Engineering & System Safety, Vol. 46, Issue 2, Pp. 123-138, 1994, ISSN 0951-8320.
11. Kitchenham, B., “Procedures for Performing Systematic Reviews”, Technical Report TR/SE-0401, Keele University (UK), 2004.

Anexo: artículos seleccionados RSL

[A1]A. Abdulkhaleq y S. Wagner, «A controlled experiment for the empirical evaluation of safety analysis techniques for safety-critical software», 2015, pp. 1-10.
[A2]S. Chandrasekaran, T. J. Madhumathy, M. Aparna, y R. S. Jain, «A safety enhancement model of software system for railways», 2009, pp. P2-P2.
[A3]J. Du, J. Wang, y X. Feng, «A Safety Requirement Elicitation Technique of Safety-Critical System Based on Scenario», en Intelligent Computing Theory, vol. 8588, D.-S. Huang, V. Bevilacqua, y P. Premaratne, Eds. Cham: Springer International Publishing, 2014, pp. 127-136.
[A4]K. A. H. Nakamura y Y. Hirao, «A strategic approach to railway signalling software», presentado en International Conference on Main Line Railway Electrification 1989, York, 1989, pp. 327-331.
[A5]J. Luo, S. Liu, Y. Wang, y T. Zhou, «Applying SOFL to a Railway Interlocking System in Industry», en Structured Object-Oriented Formal Language and Method, vol. 10189, S. Liu, Z. Duan, C. Tian, y F. Nagoya, Eds. Cham: Springer International Publishing, 2017, pp. 160-177.
[A6]T. L. Johnson, H. A. Sutherland, B. Ingleston, y B. H. Krogh, «DEPENDABLE SOFTWARE IN RAILWAY SIGNALLING», IFAC Proceedings Volumes, vol. 38, n.º 1, pp. 42-49, 2005.
[A7]T. Cichocki y J. Górski, «Failure Mode and Effect Analysis for Safety-Critical Systems with Software Components», en Computer Safety, Reliability and Security, vol. 1943, F. Koornneef y M. van der Meulen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 382-394.
[A8]B. Dehbonei y F. Mejia, «Formal methods in the railways signalling industry», en FME ’94: Industrial Benefit of Formal Methods, vol. 873, M. Naftalin, T. Denvir, y M. Bertran, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 26-34.
[A9]K. Hartig, J. Gerlach, J. Soto, y J. Busse, «Formal Specification and Automated Verification of Safety-Critical Requirements of a Railway Vehicle with Frama-C/Jessie», en FORMS/FORMAT 2010, E. Schnieder y G. Tarnai, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 145-153.
[A10]M. Wilikens, M. Masera, y D. Vallero, «Integration of Safety Requirements in the Initial Phases of the Project Lifecycle of Hardware/Software Systems», en Safe Comp 97, P. Daniel, Ed. London: Springer London, 1997, pp. 83-97.
[A11]J. Qian, J. Liu, X. Chen, y J. Sun, «Modeling and Verification of Zone Controller: The SCADE Experience in China’s Railway Systems», 2015, pp. 48-54.
[A12]Y. Chen, «Non-safety-related software in the context of railway RAMS standards», 2017, pp. 1-5.
[A13]A. El-Ansary, «Requirements Definition of Safe Software Using the Behavioral Patterns Analysis (PBA) Approach: The Railroad Crossing System», 2006, pp. 80-80.
[A14]P. Lüley, M. Franeková, y M. Hudák, «Safety and Functionality Assessment of Railway Applications in Terms of Software», en Telematics in the Transport Environment, vol. 329, J. Mikulski, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 396-405.

[A15]U. Foschi, M. Giuliani, A. Morzenti, M. Pradella, y P. San Pietro, «Software procurement and methods for specification and validation in the railway transportation industry», 2002, vol. vol.6, p. 6.
[A16]R. C. Short, «Software requirements for railway signalling systems», presentado en IEE Colloquium on Software Requirements for High Integrity Systems, London, 1988, pp. 4/1-4/3.
[A17]B. S. Medikonda y P. S. Ramaiah, «Software Safety Analysis to Identify Critical Software Faults in Software-Controlled Safety-Critical Systems», en ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India- Vol II, vol. 249, S. C. Satapathy, P. S. Avadhani, S. K. Udgata, y S. Lakshminarayana, Eds. Cham: Springer International Publishing, 2014, pp. 455-465.
[A18]P. J. Bryan, «Software Safety and Dependability for Railway Control Systems», 2014, pp. 16 (21 .)-16 (21 .).
[A19]S. Patra, «Software safety assurance process for railway platform software», 2007, vol. 2007, pp. 72-77.
[A20]A. J. Harrison y I. D. R. Shannon, «The Application of Formal Methods to Railway Signalling Systems Specification and the Esprit III Project CASCADE», en Safe Comp 95, G. Rabe, Ed. London: Springer London, 1995, pp. 101-112.
[A21]A. Lewiński y K. Trzaska–Rycaj, «The Safety Related Software for Railway Control with Respect to Automatic Level Crossing Signaling System», en Transport Systems Telematics, vol. 104, J. Mikulski, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 202-209.
[A22]K. W. W. Johnston, P. S. P. Robinson, y L. van den Berg, «Tool support for checking railway interlocking designs», presentado en SCS '05 Proceedings of the 10th Australian workshop on Safety critical systems and software, Sydney, Australia, 2006, vol. 55, pp. 101-107.
[A23]A. Fantechi, «Twenty-Five Years of Formal Methods and Railways: What Next?», en Software Engineering and Formal Methods, vol. 8368, S. Counsell y M. Núñez, Eds. Cham: Springer International Publishing, 2014, pp. 167-183.
[A24]J. L. Boulanger, "CENELEC 50128 and IEC 62279 Standards", Control, Systems and Industrial Engineering Series, John Wiley & Sons, Inc., 2015, p. 13.
[A25]NASA Software Safety Guidebook. NASA Technical Standard. NASA-GB-8719.13. Marzo, 2004.
[A26]J. Vilela, J. Castro, L. E. G. Martins, T. Gorschek, "Integration between Requirements Engineering and Safety Analysis: A Systematic Literature Review", The Journal of Systems & Software, Vol. 125, Pp. 68-92, Marzo, 2017.