# AI-FSM: Towards Functional Safety Management for Artificial Intelligence-based Critical Systems

Javier Fernández, Irune Agirre, Jon Perez-Cerrolaza, Lorea Belategi, Ana Adell, Carlo Donzella, Jaume Abella

# AI-FSM: Towards Functional Safety Management for Artificial Intelligence-based Critical Systems

Javier Fernández [†], Irune Agirre [†], Jon Perez-Cerrolaza [†], Lorea Belategi [†],
Ana Adell[†], Carlo Donzella [⋆], Jaume Abella [‡]
[†]*Ikerlan Technological Research Center, Basque Research and Technology Alliance*, Spain
[⋆]*Exida Development*, Italy
[‡]*Barcelona Supercomputing Center (BSC)*, Spain
{javier.fernandez, iagirre, jmperez, lbelategi, aadell}@ikerlan.es, carlo.donzella@exida-dev.com, jaume.abella@bsc.es

***Abstract*—This paper introduces additional systematic steps, actions, and technical considerations intending to extend conventional Functional Safety Management (FSM) for developing safety-critical systems that integrate Artificial Intelligence (AI). The proposed approach begins by outlining a safety lifecycle for safety-critical systems incorporating AI, based on the traditionally adopted V-model lifecycle. This encompasses essential phases associated with AI integration that require careful attention. To achieve this goal, the paper defines the fundamental procedures of AI-FSM, aiming to facilitate systematic failure avoidance in AI-based safety-critical systems.**

***Index Terms*—Artificial Intelligence, Functional Safety, Functional Safety Management, Lifecycle**

## I. Introduction

The development of safety-critical systems follows a well-known V-model, moving from safety goals to safety requirements, system architecture design, Software (SW) and Hardware (HW) architecture design, and implementation to obtain a system that is intended to be safe by construction. Then, the testing phase takes place from unit testing up to full system testing against its safety requirements. The Functional Safety Management (FSM) defines the required systematic approach (e.g., steps, actions, technical considerations) for developing safety-critical systems and other lifecycle phases, from concept definition up to decommissioning and disposal.

Autonomous systems often require the use of AI, and more particularly Deep Learning (DL), to perform advanced functionalities like visual perception [1], [2]. Whenever these functionalities implement safety requirements, they are also subject to the safety-critical system certification with functional safety standards such as IEC 61508 [3]. Thus, the DL subsystem that implements safety requirements must adhere to the applicable safety development and management processes [2], [4], [5]. However, the general DL-based systems development process clashes frontally with traditional safety development processes [2], [4]–[6]. For example:

1) DL SW is designed monolithically following empirical training processes with example training data, rather than implementing specific safety requirements.
2) DL SW, as opposed to any other kind of SW in safety-critical systems, cannot be considered as correct by design due to the predictive nature that comes along with mispredictions and confidence values.
3) DL SW design is no longer independent of data, and its parameters are set empirically based on training datasets.
4) DL SW imposes high-performance demands on the underlying HW and its inherent complexity (both HW and SW) entails challenges to comply with safety standards.

Moreover, there is a lack of guidance in the development process for safety-critical systems incorporating AI. To address these challenges, this paper proposes a new development process that maps the traditional lifecycle of safety-critical systems with the AI lifecycle, addressing their interactions. The aim is to reconcile the empirical data-dependent and predictive nature of DL with safety management and development processes outlined in safety standards. The proposed lifecycle extends widely adopted FSM methodologies from functional safety standards with specific needs for DL architecture specifications, data, learning and inference management, as well as appropriate testing steps.

The rest of the paper is organized as follows. Section II outlines the background. Section III outlines the steps, actions and considerations to complement a traditional FSM when the systems involves AI. Finally, in Section IV we draw conclusions and provide future research lines.

## II. Background

In this section, we provide an introduction to the foundational aspects guiding key elements of this paper.

### A. Functional Safety Management

The FSM defines a development strategy consisting of a set of procedures, guidelines, and templates that define how a project with functional safety considerations should be executed (planning, involved team, activities, documents, configuration management, modification procedures, etc.). The main goal of the FSM is to ease the definition, organization, and control of the information generated during safety-critical project development while fulfilling the requirements of functional safety standards.

The research leading to these results has received funding from the European Union's Horizon Europe Programme under the SAFEXPLAIN Project (www.safexplain.eu), grant agreement num. 101069595. Jaume Abella has also been partially supported by the Spanish Ministry of Science and Innovation under grant PID2019-107255GB-C21/AEI/10.13039/501100011033.

In the system realization phase, safety-critical systems typically adhere to a V-model and are often organized into the following phases: System Concept Specification (Ph1), System Architecture Specification (Ph2), Module Detailed Design (Ph3), Implementation (Ph4), Module Testing (Ph5), Integrations testing (Ph6) and Validation Testing (Ph7). Traditional functional safety standards such as IEC 61508 and ISO 26262 [7] define separate V-model based procedures for safety-related SW and HW.

### B. Artificial Intelligence Lifecycle

A key distinction in the DL development process, in contrast to that followed by traditional functional safety systems, lies in phases related to preparation of datasets and their utilization to create models by optimizing error functions derived from requirement specifications. Fig. 1 illustrates a simplified DL lifecycle, as proposed by [2]. Following the definition of the requirements specification, the DL workflow progresses through data management, model training, and model verification steps, culminating in the deployment of the model on the target inference platform.

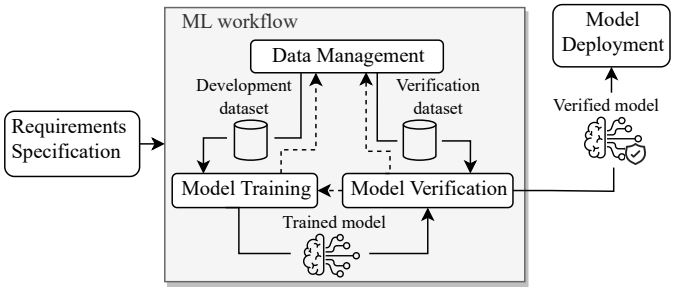


Fig. 1. Simplified DL Lifecycle [2]

Data Management is one of the most labor-intensive and crucial processes in DL development and includes activities such as data collection and data preparation. During this process, the development dataset (comprising training and validation[1]) and the verification dataset are generated. Subsequently, the model undergoes an iterative training process using the training dataset until the results on the validation dataset meet specific requirements in the model training step. Following successful training, the model verification evaluates the model's ability to properly extrapolate results with new data, utilizing the verification dataset. Lastly, the verified model is prepared in the model deployment phase for its implementation on the final inference platform. For an in-depth description guiding the AI lifecycle processes, we refer the reader to the ISO/IEC 5338 standard [8].

## III. AI-FSM

The AI-FSM is grounded in state of the art practices from functional safety according to standards such as IEC 61508 and ISO 26262 as well as in emerging initiatives in the topic of AI safety, such as EASA Concept Paper [9], AMLAS [10], A-SPICE for ML [11] and ISO/IEC TR 5469 [5]. AI-FSM is publicly available at [12].

The V-based lifecycle, traditionally followed by FSM, has been expanded considering the peculiarities of the AI lifecycle, as depicted in Fig. 2. Conventional lifecycle is represented by white boxes, while DL phases, along with those requiring slight modifications, are illustrated using colored boxes.

The traditional FSM defines procedures to be adhered to throughout the development of each task within the entire lifecycle, but it does not specifically address the nuances of the AI lifecycle [2]. The subsequent sections delve into defining procedures relevant to DL tasks. The established phases of the traditional V-based development model as adopted as they are, while for the new processes/tasks, a neutral "linear" model is adopted. There is currently no reconciled way of straightforwardly adopting a V-model arrangement for such new processes/tasks, as witnessed e.g. by [9], [11].

### A. Overall Lifecycle – Phase 0 (Ph0)

The Phase 0 (Ph0) is a transversal phase that collects all the generic project information, e.g., project document list, organizational chart, or tools selection. It must consider information collected in an AI-related safety project. When addressing a safety-critical project, all documents generated along the project must be versioned and controlled. When AI-related items are part of the project, their associated documents must also be included in the management system for its versioning and control. Similarly, when justifying the qualification of the people involved in a safety project, describing their roles, and ensuring independence between teams, AI-related skills and qualifications must be considered for specific tasks.

### B. DL-Related Concept Specification – Phase 1 (Ph1)

Besides the traditional description of the use case and the definition of the operation reflected in the requirements, the use of AI involves the specific definition of the DL Operational Design Domain (ODD) and DL operational scenarios in which the DL will operate. The definition of the DL ODD and the DL operational scenarios requires a more extensive engineering effort compared to traditional systems. The definition of ODD is application-specific, while the operational scenarios must be formulated with due consideration to the ODD. This process involves constraining the environment within which the DL operates and specifying the operational conditions.

### C. DL-Requirements Specification – Phase 2 (Ph2)

This phase entails the allocation of the software requirements specification to the DL component. These requirements are refined and shall encompass safety, operational, functional and non-functional requirements specifications as well as interface requirements. Additionally, this phase encompasses the definition of a set of metrics to assess whether the data requirements specification has been fulfilled, the test definitions, and their corresponding outcomes.

[1]Despite maintaining the term "validation" in line with AI nomenclature, it does not correspond to validation in the safety context.

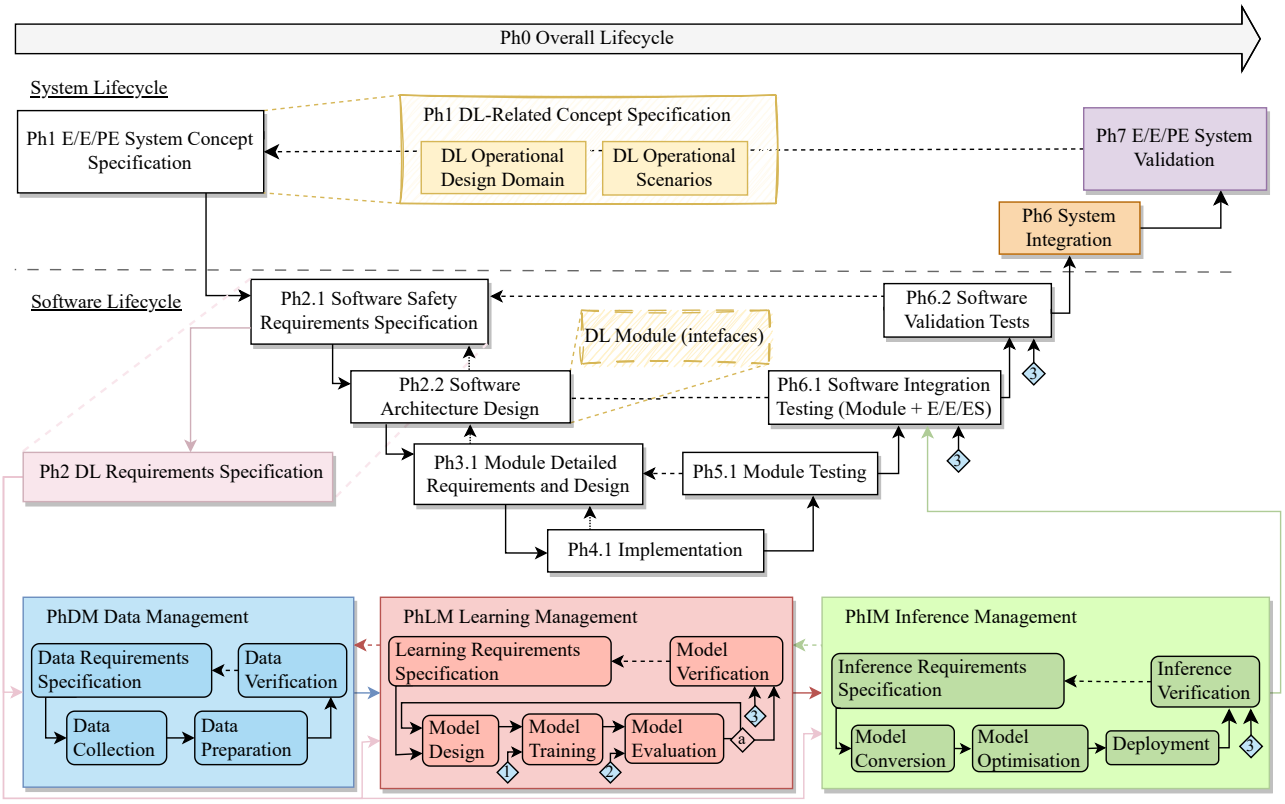


Fig. 2. Proposed AI Lifecycle

### *D. Data Management – Phase DM (PhDM)*

The PhDM is responsible for ensuring the collection and preparation of data. This phase is crucial since the behavior of the DL components is determined by extracting patterns from data. Therefore, this phase entails the proper formulation of data requirements refined and allocated from DL requirements specification. Since each of the datasets serves a different purpose, the data requirements will not always be identical. This paper proposes to differentiate between dataset requirements (i.e., the format or data characteristics) and data requirements associated with each dataset. The latter relate but are not limited to completeness, representativeness, balance, volume, or data origin. Additionally, a degree of differentiation between the datasets should be defined. For example, the verification dataset's purpose is to evaluate how the trained model extrapolates results against the training, therefore the differentiation between them is crucial. Finally, since the data usually are subject to preparation, there shall be a set of requirements related to this process such as labeling annotation, permissible data augmentation techniques, data cleaning requirements, or data pre-processing requirements such as the use of normalization or feature selection. The formulation of the data requirements would relate to Ph3 in a traditional FSM. Additionally, in the Data Requirements Specification step, the data requirements verification tests shall be defined. Defining the test corresponds with Ph3 of traditional FSM while the implementation and the results collection relate to Ph4 and 5.

Once requirements are set, data must be collected in the data collection step, which can be decomposed into i) data gathering, which involves the collection of data, and ii) data generation, which relates to generating new data to complete i). The data collection step encompasses storing all the information pertinent to the description of these intermediate steps. Completing this step is analogous to Ph4 in a traditional FSM.

However, collected data is often not a valid input data as it is. It must be cleaned (i.e., removing anomalies), processed (i.e., performing normalization, scaling, or feature selection), or annotated (i.e., labeling) to assure consistency with the expected requirements. All actions and decisions taken when preparing data shall be documented.

After generating and preparing the datasets, it is essential to verify the fulfillment of the requirements defined in the Data Requirements Specification step. This is done in the Data Verification step and corresponds with the Ph5 of traditional FSM. All the results obtained shall be documented.

PhDM generates two artifacts: a development dataset, which includes training and validation datasets (Rhombus 1 and 2 in Fig. 2), and a verification dataset (Rhombus 3 in Fig. 2). All development data, even in the traditional models, are subject to the "CIA principle" of Confidentiality, Integrity, and Accessibility, especially so for Configuration Parameters. ML data are so vast and critical that extra levels of Confidentiality and Security are to be defined and granted to them.

### *E. Learning Management – Phase LM (PhLM)*

The Learning Management phase aims to optimize and get a DL model that meets the specified DL requirements. To achieve this, learning requirements must be specified, derived from DL requirements. We propose to decompose the learning requirements into qualitative and quantitative requirements.

Concerning the first, we can list, among others, a methodology for searching the hyperparameters of the model or defining the initial parameters of the model. Quantitative requirements allow the evaluation of quantifiable properties of DL models. For example, those associated with the model bias requirements to avoid underfitting, performance requirements (such as accuracy, precision, or recall), and robustness requirements (i.e., the model shall perform within the performance thresholds within unseen data in the training dataset).

In addition to this, the model selection criteria shall be defined for the model verification step in those cases in which several candidate models provide the required performance. For example, the model criteria can prioritize accuracy for specific classes, the highest robustness under particular environmental conditions or prioritize models with the highest levels of explainability. The formulation of learning requirements would relate to Ph3 in a traditional FSM.

Once the learning requirements specifications are defined, the Model Design step focuses on the specification of a set of DL models that best suit the application. All the decisions conducted during this step and the different models shall be documented. This process begins with analyzing well-known models that have succeeded in similar task domains and it is often subject to modifications since Learning Management is an iterative process. Consequently, successive iterations result in incremental versions of this documentation.

Subsequently, the model is generated from the training dataset and evaluated employing the validation dataset, both from PhDM. The result of the evaluation is checked (red rhombus with letter "a" in Fig. 2). On the one hand, a scenario may arise in which none of the previous candidates achieves the expected performance. In such cases, an iterative repetition of the model design, training, and evaluation becomes necessary. These iterations continue until the stipulated performance requirements are successfully met. Otherwise, a new iteration of the PhDM should be carried out. On the other hand, if one or multiple candidate models demonstrate the anticipated performance levels, they will be verified in the next step.

Finally, the model verification step assesses the generalization capabilities, identifies potential issues using the verification dataset, and verifies compliance with requirements specifications. The verified model, generated by the PhLM, is then utilized as input in the PhIM. Decisions regarding its selection should be documented and aligned with the model selection criteria. Iterations of PhLM or even the PhDM are necessary until requirements are achieved.

### *F. Inference Management – Phase IM (PhIM)*

The purpose of this phase is to ensure that the target inference model still fulfills the specified DL requirements after adapting and even optimising the model for its deployment on the target HW. As with the PhDM and PhLM, the initial PhIM step lies in the definition of the inference requirement specification, refining those from the PhLM and the Ph2, and generating the verification tests. These requirements shall encompass aspects associated with the model conversion, optimization and deployment. The formulation of inference requirements would relate to Ph3 in a traditional FSM.

Then, the verified model undergoes a conversion process to transform it into a format suitable for deployment (i.e., elimination of training-specific operations) that shall ensure compatibility with a specific target inference platform in the Model Conversion step.

In the Model Optimisation step, the model may undergo optimizations to enhance its performance, reduce its size, or adapt it for resource-constrained environments. Optimization aims to maintain or improve the model's performance while making it more efficient for deployment.

Finally, the optimized model is deployed on the target platform (Deployment step), which is subsequently subject to a comprehensive verification process (Inference Verification step). This involves checking the optimized model (or the converted model in cases where the optimization step is not performed) against specified criteria to ensure that the model adheres to the inference requirements specification.

Information relative to the process of converting and optimizing the model, as well as the inference verification results, shall be explicitly documented.

## IV. Conclusions and Future Work

In recent years various emerging initiatives and standards have been developed to align the use of AI in safety-critical systems, but it remains an open research challenge [2]. In this context, this paper addresses the updates required in the realization phase of traditional functional safety standards to support the safe development of AI in safety-critical systems.

In the future, the AI-FSM may be updated to align with upcoming iterations of emerging standards. Examples include ISO/CD PAS 8800 [4], IEC TS 6254 [13] or ISO/IEC 5338 [8], currently in development during the creation of this paper.

## References


[1] L. Jiao *et al.*, "A Survey of Deep Learning-Based Object Detection," *IEEE Access*, vol. 7, pp. 128 837–128 868, 2019.

[2] J. Perez-Cerrolaza *et al.*, "Artificial Intelligence for Safety-Critical Systems in Industrial and Transportation Domains: A Survey," *ACM Comput. Surv.*, 2023.

[3] "IEC 61508(-1/7): Functional safety of electrical / electronic / programmable electronic safety-related systems," 2010.

[4] "ISO/CD PAS 8800 Road Vehicles — Safety and artificial intelligence," 2023.

[5] ISO/IEC, "ISO/IEC CD TR 5469 Artificial intelligence — Functional safety and AI systems."

[6] A. Brando *et al.*, "On Neural Networks Redundancy and Diversity for Their Use in Safety-Critical Systems," *Computer*, vol. 56, 2023.

[7] "ISO 26262(-1/11) Road vehicles – Functional safety," ISO, 2018.

[8] "ISO/IEC WD 5338 Information technology — Artificial intelligence — AI system life cycle processes," 2023.

[9] European Union Aviation Safety Agency (EASA), "EASA Concept Paper: guidance for Level 1 & 2 machine learning applications," 2023.

[10] R. Hawkins *et al.*, "Guidance on the Assurance of Machine Learning in Autonomous Systems (AMLAS)," 2021.

[11] "Automotive SPICE® Process Assessment / Reference Model Version 4.0," 2023.

[12] J. Fernández *et al.*, "AI-FSM," 2024. [Online]. Available: https://doi.org/10.5281/zenodo.10964402

[13] "IEC TS 6254 - Information technology — Artificial intelligence — Objectives and approaches for explainability of ML models and AI systems," Under Development.