Journal of Biomedical Informatics 167 (2025) 104856

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Navigating regulatory challenges across the life cycle of a SaMD

Martina Francesconi [a], Miriam Cangi [b], Silvia Tamarri [b], Noemi Conditi [c,d], Chiara Menicucci [e], Alice Ravizza [d], Luisa Cattaneo [a], Elisabetta Bianchini [f,*]

[a] Ente di Supporto Tecnico Amministrativo Regionale (ESTAR), Dipartimento Tecnologie Informatiche, Pisa, Italy
[b] Thema S.r.l., Imola, Italy
[c] Stefanelli and Stefanelli Law Firm, Bologna, Italy
[d] InsideAI S.r.l, Bologna, Italy
[e] Bureau Veritas Italia, Milano, Italy
[f] National Research Council (CNR), Institute of Clinical Physiology (IFC), Pisa, Italy

ARTICLE INFO

Keywords:
SaMD
Medical device
MDR
Regulation
Life cycle
Conformity assessment

ABSTRACT

Objective: Software as medical devices (SaMDs) have become part of clinical practice and the management of the development and control processes of the documentation associated with them are an integral part of many medical realities. The European Regulation, MDR (EU) 2017/745, introduces a classification rule (rule 11, Annex VIII) specifically for software, which provides more explicit requirements than in the past, leading to classification of many software to higher risk and therefore to more complex certification processes. In this context, planning and awareness of possible regulatory strategies and related standards are fundamental for the key stakeholders, but this complex landscape can be perceived as fragmented. The aim of this work is to provide an amalgamated overview of how the current EU normative framework integrates into the various phases of the life-cycle of a medical device software, trying to ensure its safe and effective development.
Methods: In addition to the MDR, the main normative references relevant to the medical device software sector were taken into consideration. Specifically, the IEC 62304 standard clarifies the main processes of the software life-cycle, including the analysis of problems and changes, and the IEC 82304 standard completes its management by addressing activities relating to post-market phases and requirements. In addition, the various steps include also key points such as risk identification and control (ISO 14971), design, implementation and validation of usability requirements (IEC 62366) and in general the quality of the context in which the software is developed and maintained (ISO 13485). The application of these standards can support the activities of the various stakeholders and facilitate evidence of compliance with the regulatory requirements by MDR.
Results: Based on the software life cycle, a mapping of the requirements from the entire normative framework analyzed over the various phases was implemented.
Conclusions: A detailed and integrated picture of the regulatory context behind the life cycle of a SaMD has been provided: this can facilitate the implementation of a balanced and effective approach, including key aspects, such as risk management and usability processes, and ensuring safety for the end user.

1. Introduction and related work

The digital revolution we are experiencing had a significant impact in the healthcare sector. More medical devices have become part of clinical practice, and the management of the development and control processes of the associated documentation is a challenging activity for many medical realities [1,2].

The evolution of software into medical devices has revolutionized healthcare, enhancing diagnostic accuracy, treatment efficacy, and patient outcomes. However, the complexity and critical nature of Software as a Medical Device (SaMD) necessitates stringent regulatory oversight to ensure safety and effectiveness. The software life cycle encompasses a series of phases, from initial concept and development through to deployment and maintenance, each governed by specific regulatory requirements [3456]. Software can be qualified and placed on the market as a medical device, as an accessory for a medical device or as a

* Corresponding author.
E-mail address: elisabetta.bianchini@cnr.it (E. Bianchini).

https://doi.org/10.1016/j.jbi.2025.104856
Received 10 March 2025; Received in revised form 26 April 2025; Accepted 20 May 2025
Available online 21 May 2025

part or a component of a medical device, and, accordingly, it can be subject the specific regulation.

Internationally the management of SaMDs is established through the availability of specific references aimed to facilitate the adoption of regulatory processes by the manufacturer. The IMDRF (International Medical Device Regulators Forum) has published multiple documents regarding the regulation of software, like risk categorization framework, quality management system requirements to a medical device software and principles for demonstrating the safety, effectiveness and performance of medical device software. At the European level, however, the need to define a specific and convergent regulatory thinking about SaMD has been growing in recent years. The recent European Regulation, 2017/745 MDR (EU) [789] governs the development and placement on the market of medical devices in Europe in order to ensure safety and performance. Among the various novelties, MDR introduces a clearer qualification approach [10]and a classification rule (rule 11, annex VIII) specifically for software which provides more explicit requirements than in the past, and leads to the transition of many systems to classes of higher risk than under the previous legislative framework and therefore to more complex certification processes. In this complex and fragmented context, planning and awareness of possible regulatory strategies and supporting guidance are fundamental for the various stakeholders[11–13]. This requires a multidisciplinary approach with a detailed vision of the SaMD lifecycle together with knowledge of the main references and the key regulatory requirements to be considered.

The purpose of this work is to provide a balanced and consistent overview of how the current regulatory framework is integrated into the various phases of the life cycle of medical device software, with a focus on the EU, trying to ensure its safe and effective development and the maintenance of its performance and safety once placed on the market.

STATEMENT OF SIGNIFICANCE	
Issue	Complex and fragment normative scenario around the SaMD lifecycle. Need of a convergent thinking.
What is Already Known	Availability of various international guidelines/standards aimed to facilitate the adoption of regulatory processes by the manufacturer.
What this Paper Adds	Balanced and consistent overview of how the current normative framework can be integrated into the various phases of the life cycle of medical device software, with a focus on the EU market.
Who would benefit from the new knowledge in this paper	Innovators, developers, manufacturers of SaMD.

2. Methods

A multidisciplinary team including healthcare system managers, developers, and regulatory experts (experience in the field = 13,9 ± 8,5 years) involved in the SaMD development, maintenance and use processes was created in order to provide an overview of the related regulatory framework. The activity of the team included: i) agreement about key stages of the SaMD lifecycle; ii) identification, description and deep discussion about the key references considered useful within this context; iii) brainstorming and deep discussion about the links and intersection among these references; iv) summary and mapping of the output. More specifically, as described in the subsequent paragraphs, the European Regulation, MDR (EU) 2017/745 [7] and the main references relevant to the sector were taken into consideration with a focus on their role within the software lifecycle.

2.1. Overview of medical device software lifecycle stages

The team considered that the management of SaMD involves a set of structured processes to ensure safety, effectiveness and regulatory compliance. A scheme for this context, according to the IEC 62304[14] and IEC 82304 [15] vision, is reported in Fig. 1 and it can be divided into three main blocks, **development, maintenance and post-market**, whose key activities are described below.

Customer Needs and Regulatory Requirements, including preliminary risk analysis: The initial phase includes a comprehensive analysis of user and patient needs and regulatory requirements. Understanding user and patient needs involves gathering detailed information about the intended use of the software, user workflows, and specific functionalities required. Regulatory requirements depend on the jurisdiction in which the software will be available and the specific nature of the software. After qualification as a medical device, determining the software's classification based on its intended use and risk to patients is important to define the regulatory pathway. This step is crucial in identifying the scope and objectives of the software development project. Additionally, core risks need to be identified, and their mitigation measures are included in the input requirements. Besides the manufacturer, the key stakeholders involved in this stage are the end-users (e.g. medical operators and/or patients), developers, regulatory experts, healthcare professions (e.g. clinical engineering).

Software Development and Configuration Management: This phase ensures that medical device software is developed systematically and maintained effectively throughout its lifecycle. Key activities are: i) software design, implementation and testing according to predefined requirements; ii) control, tracking and maintaining of all aspects of the software and related artifacts throughout the software lifecycle. IEC 62304 underlines the importance of verifying during the development stage that the software is built according to specifications and design ensuring that it meets all requirements for functionality, quality, and safety. Verification involves activities like code reviews, unit testing, integration testing, and system testing [17]. Moreover, the application of IEC 62304 within this phase pivots on the definition of the architectural structure of the device, where the architecture is described as composed of MODULES that are subsequently subdivided into ITEMS. To each ITEM, developers may associate a specific functionality and therefore clearly identify which ITEMS actually provide the intended clinical use. Furthermore, specific identification of software developed by third parties, the so-called SOUP (Software of Unknown Provenance) is required by IEC 62304. Some development approaches, once all software requirements have been verified, might include the creation of a release candidate for the validation phase. Besides the manufacturer, the key stakeholders involved in this stage are the developers and regulatory experts.

Software Validation: According to IEC 82304 and IEC 62304, while verification ensures the software works as intended per the design, validation activities ensure that it meets the need for the user and performs safely in real-world scenarios. This phase includes tasks to check that the software performs safely and effectively for its target users; validation is linked to clinical evaluation and risk assessment in order to ensure that the software is effective for its intended use in a health context.

Besides the manufacturer, the key stakeholders involved in this stage are the end-users (e.g. medical operators and/or patients), developers, regulatory experts, Information Technology (IT) departments.

Software Placing on the market: After the completion of verification and validation steps a final software release is approved and it can be placed on the market. It is worth noting that different technical and business strategies to reach the end-users can be implemented ranging e.g. from locally installed software to web-based applications and the manufacturer needs to consider several aspects in terms of safety and security related to the different types of architecture. MDR specifies the requirement of general conformity for this step (Article 5 (1)), the obligation to refer to an identification system, based on a unique device identifier (UDI), ensuring traceability of the medical device on the marked and the need for registration of the involved economic operators. Besides the manufacturer and any other relevant economic

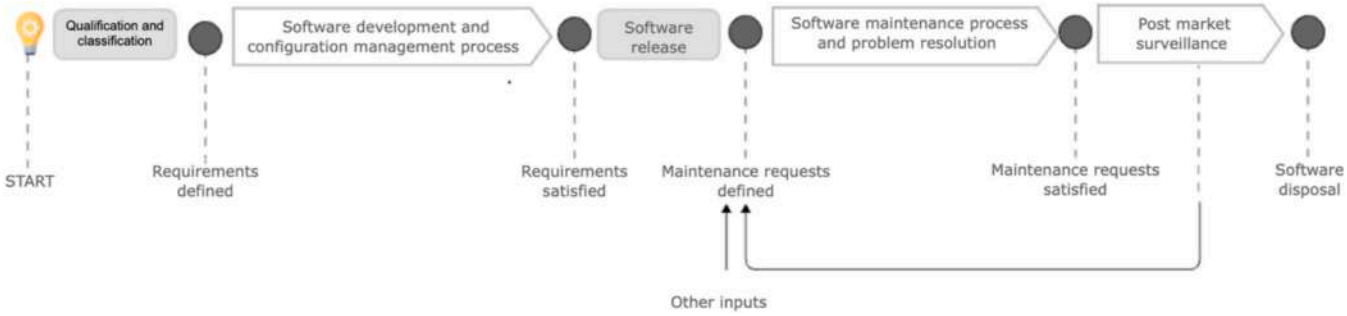

Fig. 1. The software medical product processes. Useful definitions: SaMD = software that is intended to be used, alone or in combination, for a purpose as specified in the definition of a "medical device" in the medical device regulation [16]; software lifecycle = all phases in the life of medical device software, from the definition of its requirements to final decommissioning and disposal; post-market surveillance = systematic process to collect and analyze experience gained from medical devices that have been placed on the market, according to MDR.

operator, the key stakeholders involved in this stage are the notified bodies.

Maintenance Requests: maintenance involves a comprehensive analysis, monitoring and management of e.g., adaptive, perfective and preventive requests addressing evolving safety, security, and functional issues, as well as for keeping the software up to date with regulatory and industry standards. Besides the manufacturer, the key stakeholders involved in this stage are the end-users (e.g. medical operators and/or patients), developers, regulatory experts, IT departments.

Software maintenance and Problem resolution: the software maintenance involves regular updates and modifications to the software to address the above mentioned requests. In addition to this process, problem resolution focuses on identifying, diagnosing, and rectifying issues that arise during the software's operational phase. Besides the manufacturer, the key stakeholders involved in this stage are the end-users (e.g. medical operators and/or patients), developers, regulatory experts, clinical engineering.

Post-Market Activities: post-market surveillance activities are conducted to monitor and maintain the quality, safety and performance of the medical device software, to comply with applicable regulatory requirements, and to contribute to the management of the life cycle of devices. Post-market surveillance activities involve systematically collecting, recording and investigating information on the device use in the market. Besides the manufacturer, the key stakeholders involved in this stage are the end-users, the Notified Bodies, and the clinical engineering.

Software Disposal: Software disposal is the final phase of the medical device software life cycle, encompassing the systematic decommissioning and secure removal of software that is no longer in use. This phase is critical for protecting patient data, maintaining compliance with regulatory requirements, and ensuring that the software lifecycle management processes are complete and secure. Besides the manufacturer, the key stakeholders involved in this stage are the developers and the end-users.

Throughout each stage of the lifecycle, feedback loops are incorporated to facilitate continuous improvement and ensure compliance with customer expectations and regulatory requirements. Medical device software life consists of an iterative process with a cyclical approach where the different phases are refined through analysis, evaluation, testing, and feedback, that allow to align the product with clinical needs and regulatory requirements. The design and development just as the maintenance of the product are break down into manageable iterations, allowing for analysis, design, implementation, testing, and review until the continuous changing requirements are satisfied, and the user experience and feedback from clinicians and patients are gathered to refine the design for ensuring usability. This approach allows for changes to the product scope or requirements as the project progresses and helps identify areas for improvement in the development process.

2.2. Key regulatory requirements

Together with the European Regulation, MDR (EU) 2017/745 [1], the main normative standards relevant to the medical device software sector were identified and considered by the team (Fig. 2). Standards can be harmonized with MDR (i.e., developed by a recognised European Standards Organisation following a request from the European Commission in order to facilitate demonstration that products, services, or processes comply with relevant EU legislation) or not, and in general, since they provide the state-of-the-art of a specific context, they are means to provide presumption of conformity to safety and performance requirements. Specifically, as already introduced in the previous paragraph, the IEC 62304 [14] standard clarifies the main processes of the software life cycle, including the analysis of problems and changes, and the IEC 82304 [15] standard completes its management by explaining activities relating to the requirements and post-market phases. The aspects implemented in the various phases also include key points such as risk identification and control (ISO 14,971 [18]), design, implementation and validation of usability requirements (IEC 62,366 [19]) and in general the quality of the context in which the software is developed and maintained (ISO 13,485 [20]). Some of these interconnections are also reported in Annex C of IEC 62304 [14].

The above reported standards are interrelated and their simultaneous application can support the activities of the various stakeholders and facilitate evidence of compliance with the regulatory requirements by MDR. The team, starting from the software life cycle scheme, elaborated an integrated overview based on those points, of the above described fragmented normative framework, that were considered most specifically interconnected.

A summary of the main topic for each standard is reported below.

2.3. Medical device Regulation (MDR) 2017/745

A software is qualified as a medical device when it has a medical purpose; this can include diagnosis, prevention, monitoring, prediction, prognosis, treatment or alleviation of disease or diagnosis, monitoring, treatment, alleviation of, or compensation for, an injury or disability. For example a software with diagnostic or therapeutic purpose, a software that provides support for healthcare professionals in making a therapeutic or diagnostic decision or control a medical device should be considered a medical device. Therefore, to place on the market or put into service a medical device software, the manufacturer must comply with the Medical Device Regulation (MDR) 2017/745.

MDR 2017/745 establishes clear rules to be complied with in order to lawfully place any medical device on the market aiming to ensure, when medical devices are used, the maximum level of safety and protection of patients, users and other person's health. These norms are not only safety and performance requirements of the product, to consider when designing and manufacturing it, but also obligations for all the

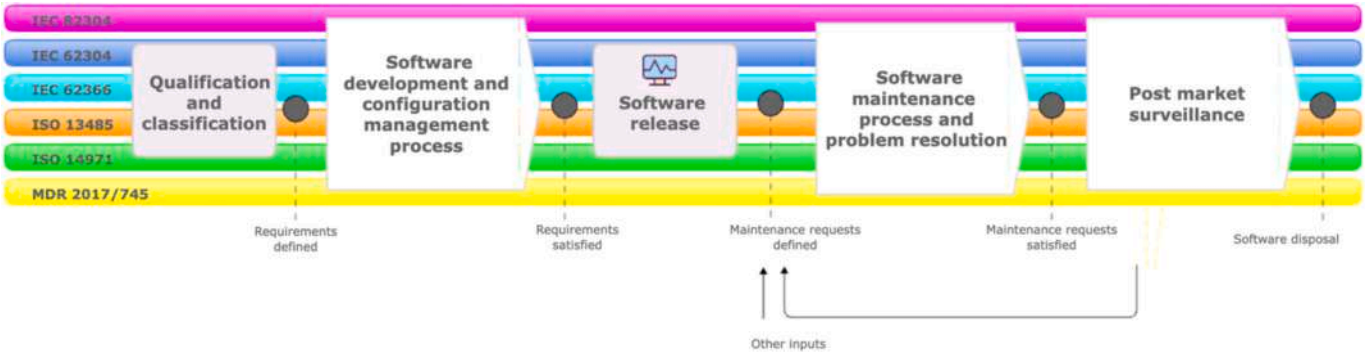

Fig. 2. The normative framework behind the software medical product processes.

economic operators involved, especially the manufacturer. As for the latter, in particular, the MDR provides for the duty to have in place a quality management system.

To comply with the mentioned requirements and obligations, it is particularly useful to apply the standards already mentioned and further described in the following paragraphs. In particular, when it comes to the product's requirements, article 8 MDR establishes that "devices that are in conformity with the relevant harmonised standards, or the relevant parts of those standards, the references of which have been published in the Official Journal of the European Union, shall be presumed to be in conformity with the requirements of this Regulation covered by those standards or parts thereof." ISO 13485 and ISO 14971 are widespread harmonised standards. Specifically for software life cycle management, at the present time no harmonised standards are available, but the presented references are considered state of the art in the industry.

2.4. ISO 14971: Risk management

ISO 14,971 [18] is an international standard that supports medical device software manufacturers to establish, document and maintain a systematic risk management process at all stages of a medical device's life cycle. The standard helps identify risks associated with medical device software, estimate and evaluate those risks, identify risk control measures, and monitor the effectiveness of those controls.

Within this context two further documents are worth mentioning: i) the ISO TR 24971 [21], a companion guidance supporting the development, implementation and maintenance of a risk management system for medical devices according to ISO 14971; ii) the IEC/TR 80002–1 [22], a technical report aimed to facilitate the application of ISO 14971 to medical device software.

2.5. ISO 13485: Quality management system

ISO 13,485 [20]is an international standard that outlines the requirements for a comprehensive quality management system (QMS) for medical devices. It includes requirements for software development processes, documentation, validation, distribution, maintenance, and post-market surveillance. An organization must design and implement a QMS that can demonstrate its ability to deliver a medical device software that consistently meets customer requirements and applicable regulatory requirements.

2.6. IEC 62366: Usability requirements

The usability of medical device software is governed by the IEC 62366 [19]standard, available in its two parts: IEC 62366–1 and IEC TR 62366–2 containing basic information and application guidelines. The application of an appropriate usability process makes it possible to ensure that a medical device software is suitable for its intended use, that it is safe and effective and that the risks associated with their use are acceptable compared to the benefits brought to the patient.

2.7. IEC 62304: Software lifecycle requirements

The international standard IEC 62304 [14] defines the requirements for the life cycle of medical software or software within a medical device, from the development phase to the maintenance and process control phase, to risk management. The standard sets out the general requirements for developing software that consistently meets customer requirements and applicable regulatory requirements.

2.8. IEC 82304: Safety and security of healthcare software

The international standard IEC 82304 [15] outlines safety criteria that SaMD must meet to prevent harm to users or patients, related to entire software life cycle, including design, development, validation, maintenance and disposal. This standard is applicable only for medical device software that is made available as a standalone product.

Besides the mapping of requirements derived from the above described standards, mainly focused on safety/effectiveness requirements according to MDR, the team decided to identify and report specific further inputs that within the software life cycle can be considered according to e.g., innovation or security.

3. Results

3.1. Software life cycle requirements

The team, starting from the proposed software life cycle scheme, provided an integrated mapping of the requirements deriving from the fragmented normative landscape described in the previous sections. The results of the analysis are reported according to the three main blocks identified for the product processes (**A — development, B — maintenance and C — post-market**, Figs. 3, 4, 5, respectively): for each phase the key interconnected points related to the analyzed standards and regulation have been identified and reported.

To enhance the readability and the utility of the interconnected standards and regulation, a summary table outlining the specific references associated with each phase is available in Annex A.

3.1.1. PART A – Software development

Medical device design focuses on developing software that diagnose, treat, or prevent medical conditions. Operating in a high-risk environment requires safety, efficiency, and user satisfaction. To achieve these objectives, multidisciplinary collaboration is crucial throughout the development process, ensuring the device is safe, meets user needs, enhances usability, and complies with regulatory requirements.

Besides qualification and classification (*Annex VIII — rule 11, 4.3 Software safety classification*) based on the defined intended use for medical device software (*Article 2*), the manufacturer shall consider the

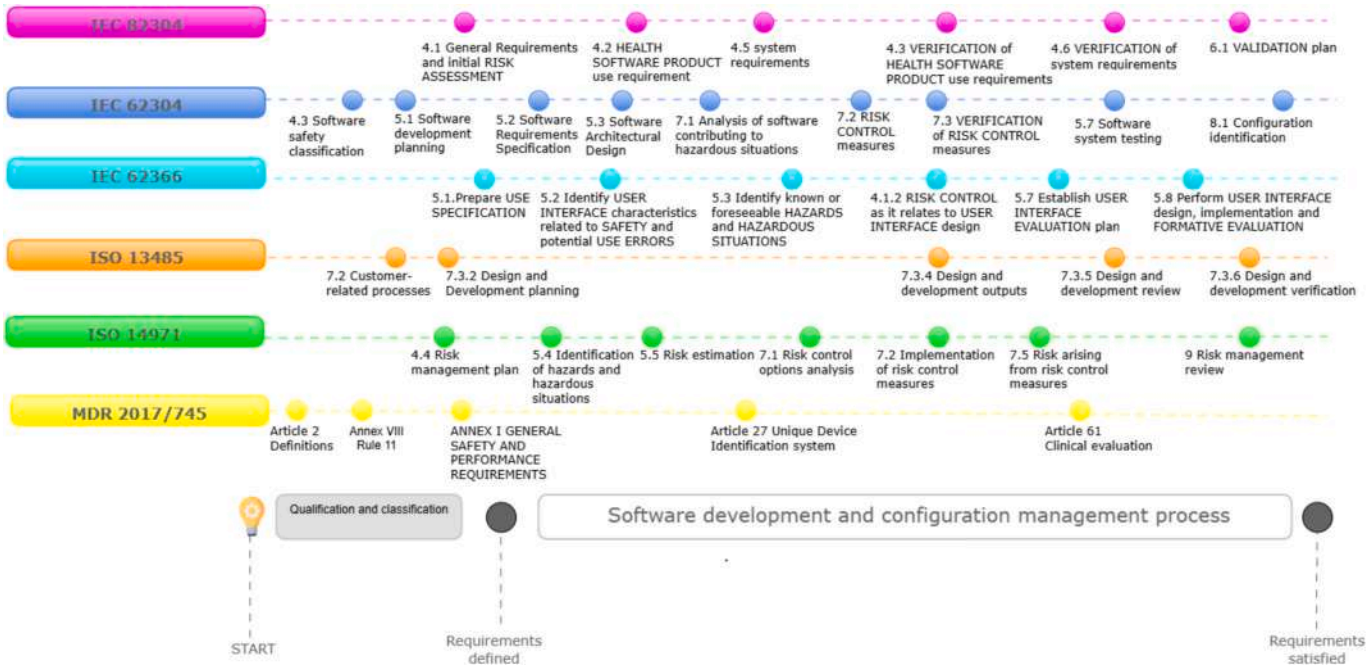

Fig. 3. Key interconnected normative points within the software medical product development.

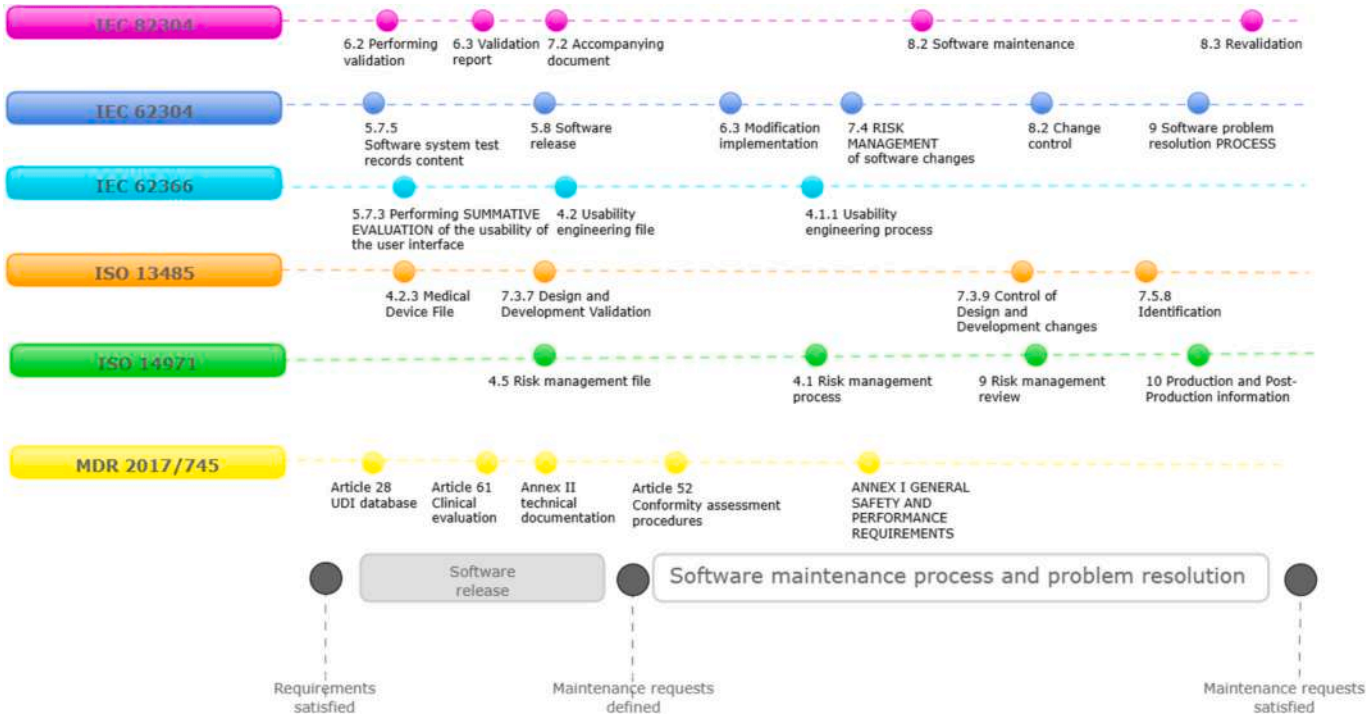

Fig. 4. Key interconnected normative points within the software medical product maintenance.

intended user profile, the intended usage environment, and the characteristics of safety and security. This activity is carried out thanks to the application of ISO 14971 to identify hazardous situations and harms. According to ISO 14971 and IEC 62304, each risk can be characterized by its severity and its probability, but in those cases where the estimation of probability is challenging (such as for example the probability of occurrence of a software bug), the estimation of the risk is based solely on its severity.

Subsequently the manufacturer can identify user requirements, requirements to reach intended use, and regulatory requirements (*4.2 HEALTH SOFTWARE PRODUCT use requirements, 5.1 Prepare use specification, ANNEX I GENERAL SAFETY AND PERFORMANCE REQUIREMENTS*) that are added to the required risk mitigation measures to create a full list of inputs to define software requirements (*5.2 Software Requirements Specification, 4.5 System requirements*). The initial definition of software requirements is necessary to define the software's architecture, its interactions with external hardware and software components and specify its functional and performance requirements (*5.3 Software ARCHITECTURE DESIGN*). The different activities around the software development need to be planned and implemented

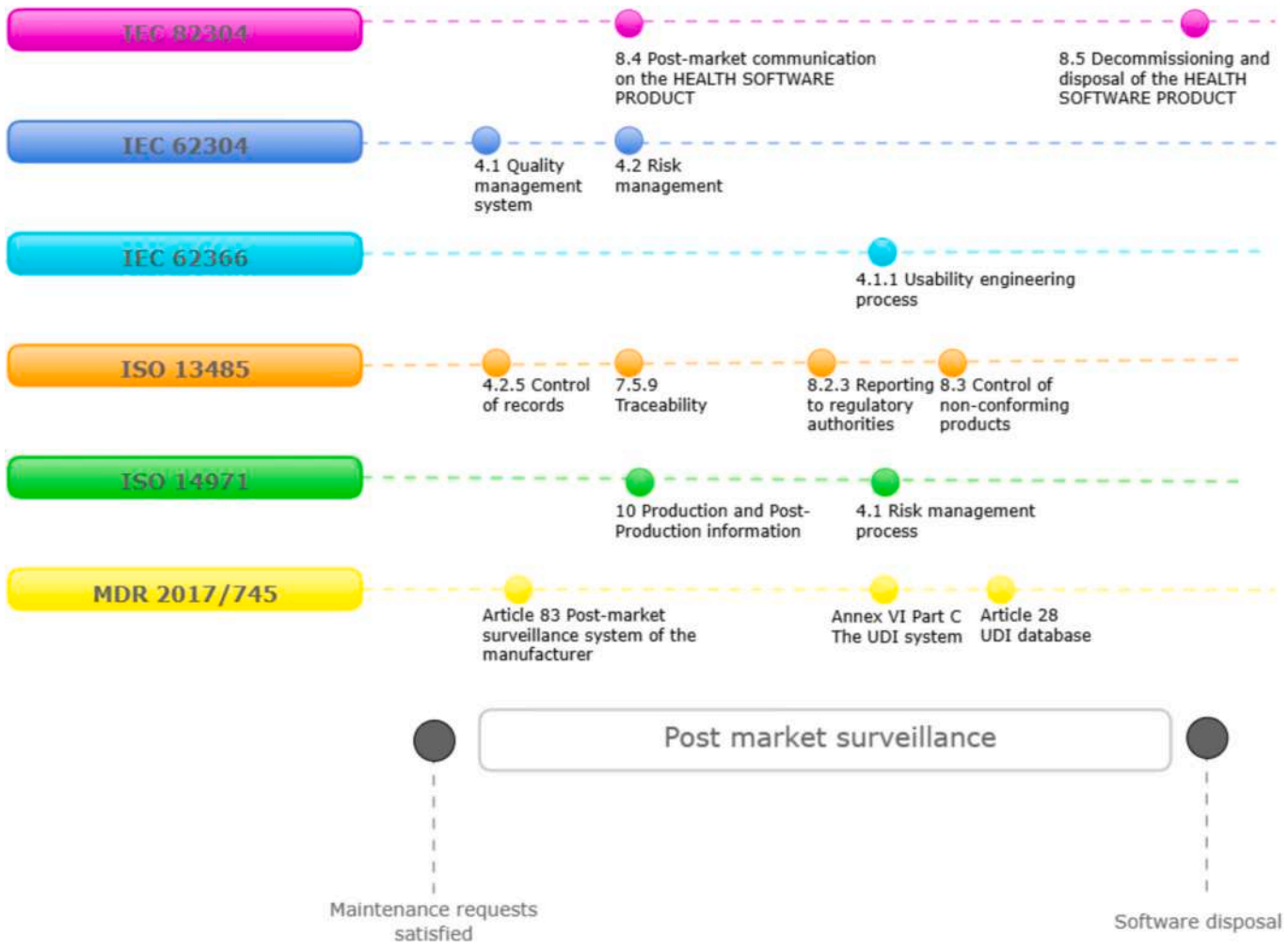

Fig. 5. Key interconnected normative points within the post-market phase.

considering risk and quality management, usability processes and verification and validation evidence (*7.2 customer-related processes, 5.1 Software development planning, 4.4 risk management plan, 7.3.2 Design and Development planning, 5.7 Establish USER INTERFACE EVALUATION plan, 6.1 Validation plan, Article 61 Clinical Evaluation*). Importantly, the planning has to ensure traceability among the different aspects linking e. g. system/software requirements and testing with risk control measures.

The risk analysis process (5.*4 Identification of hazards and hazardous situations, 5.5 Risk estimation*) is typically concurrent with development steps since the design input phase (*4.1 General requirements and initial RISK assessment, 7.1 Analysis of software contributing to hazardous situations*); importantly, a usability engineering process must manage safety related to user interface (*5.2 Identify USER INTERFACE characteristics related to SAFETY and potential USE ERRORS, 5.3 Identify known or foreseeable HAZARDS and HAZARDOUS SITUATIONS)*. After the identification of hazardous situations, the software risk control measures should be defined to reduce the probability of the software failing (*7.1 Risk control option analysis, 7.2 Implementation of risk control measures*) and this is typically part of development activities (*7.2 RISK CONTROL measures, 4.1.2 RISK CONTROL as it relates to USER INTERFACE design*). To assess the efficacy of control measures, a verification that software risk control measures work as intended should be done (*7.3 VERIFICATION of RISK CONTROL measures, 7.5 Risk arising from risk control measure, 9 Risk management review*) and this is typically part of functional testing as described in IEC 62304 (*5.7. Software system testing*). In general, a detailed testing process on the developed output (*7.3.4 Design and Development output)* needs to be implemented (*7.3.6 Design and Development verification*) and shall address both use and system requirements (*4.3 VERIFICATION of HEALTH SOFTWARE PRODUCT use requirements, 4.6 VERIFICATION of system requirements*). If the developers include software ITEMS that are developed by third parties (including operating systems) detailed policies for the selection of said SOUP shall be provided. To identify potential software use-related risks and make the necessary adjustments to mitigate them, a formative evaluation has to be conducted "with the intent to explore user-interface design strengths, weaknesses, and unanticipated use errors" (*5.8 Perform USER INTERFACE design, implementation and FORMATIVE EVALUATION*); this phase is then integrated at the end of the development with the summative evaluation aimed to "obtain objective evidence that the user interface can be used safely".

A unique identification of software and its components, including related versions, should be established to ensure device traceability on the market (Unique Device Identifier – UDI system) and improve its safety (*Article 27, 8.1 Configuration identification*).

Therefore, after verification, the software design and development are completed with the satisfaction of the initial requirements and review of the key activities (*7.3.5 Design and Development review, 9 Risk management review*).

3.1.2. PART B – Software maintenance

After the software verification has been completed and all design and development activities have been documented, a verified version of the software is released (*5.8 Software release*).

The software placement on the market involves a series of further

activities: software validation to confirm that the software meets the use intended during its design (*6.2 performing validation, 7.3.7 Design and Development validation, 6.3 Validation report*), summative evaluation of usability to get objective evidence that the medical device is safe to use (*5.7.3 SUMMATIVE EVALUATION of the usability of the user interface*), assessment in clinical practice to determine safety and efficacy in relation to the intended purposes (*Article 61 Clinical Evaluation*). Importantly, all the implemented activities must be reported and collected in a medical device file (*4.2.3 Medical Device File*) in order to demonstrate that all the various requirements have been met (*7.2 Accompanying documentation, 5.7.5 Software system test records content, 4.2 Usability engineering file, 4.5 Risk management file, Annex II technical documentation*) and documenting procedures for product identification (*7.5.8 Identification*) must be ensured. It is worth noting that the manufacturer shall undertake an assessment of the conformity of the device, in accordance with the applicable procedures (*Article 52, ANNEX I General Safety and Performance Requirements*); for class II and III devices, to place the software on the market, the intervention of a third-party notified body for the conformity evaluation will be necessary. Moreover, it is necessary to submit the information to the UDI database before the product is placed on the market in order to ensure its traceability (*Article 28*).

The testing phase is not concluded with the pre-release validation and verification stages but, especially for software installed in healthcare systems, Information Technology departments are responsible for evaluating the software in the environment in which is used. In particular, the final conformity assessment that concludes the administrative validation procedure, is made up by dedicated integration and functional tests, on the software itself, which come from internal operating procedure inspired to international standard such as the *ISO 12207 Systems and software engineering – Software life cycle processes*, from which the 62,304 has been derived in terms of approach and concepts. Of considerable importance is the subsequent phase of maintenance of the developed product (*8.2 Software maintenance)*. Policies to address the modification requests and change control of both ITEMS that are developed internally and SOUPS must be adopted (*6.3 Modification implementation, 7.4 Risk management of software changes, 8.2 Change control, 7.3.9 Control of Design and Development changes*) and need to ensure risk control and traceability of changes. This phase of the life cycle focuses on the management of modifications and changes and includes also bug fixes and problem resolution (*9 Software problem resolution process*) [3]. Software changes might also imply an update of the UDI vector depending on whether it is a major or minor release. Importantly, the regulatory impact of new versions, according to the concept of significant changes introduced in MDCG 2020–3 [23] that are related to the design or intended purpose of the device, requires a careful assessment on a case-by-case basis.

A re-validation might be needed according to the type of implemented changes (*8.3 Revalidation*). Within this phase risk management and usability processes need to be transversally and continuously considered as well (*4.1 Risk management process, 9 Risk management review, 10 Production and Post Production information, 4.1.1 Usability engineering process).*

3.1.3. PART C – software post-market

A set of post-market surveillance (PMS) activities needs to be ensured in order to monitor the quality, performance and safety of the software device once it is placed on the market (*Article 83*). Traceability of the device needs to be guaranteed (*7.5.9 Traceability. Annex VI Part C – The UDI system, Article 28 UDI database*) as well as management of non-conformities (*8.3 Control of non-conforming products*) and communication with the regulatory entities (*8.2.3 Reporting to regulatory authorities*). The availability of structured procedures facilitates this phase (*4.1 Quality management system, 4.2.5 Control of records*). The PMS system serves not only to meet regulatory requirements, but also to improve risk management and improve the quality of a software device by appropriately identifying any eventual problems in design and/or use that can be managed according to the maintenance process reported above. If post-market surveillance data highlights product issues that impact on safety and security, software changes will need to be managed to comply with the requirements of ISO 82304. During the post-market surveillance, security, vulnerability and changes in regulatory requirements that impact on the use of the software shall be evaluated, analyzed and communicated to the final user (*8.4 post-market communication on the HEALTH SOFTWARE PRODUCT*). Importantly, within this phase usability and risk management processes need to be continued *(4.2 Risk management, 4.1 Risk management process, 10 Production and Post Production information, 4.1.1 Usability engineering process.)*.

It is also crucial to communicate the availability of a new software version linked for example to the introduction of new features, to updates on the software identification or to updates of the accompanying documentation, as well as to the correction of errors or bugs (*8.4 post-market communication on the HEALTH SOFTWARE PRODUCT*).

The life cycle of the software ends with the decommissioning and disposal of the software and this part in technical terms is detailed and described by the 82,304 standards. This final stage shall be included, as appropriate, safeguarding personal and health-related data in connection with security and privacy (*8.5 Decommissioning and disposal of the HEALTH SOFTWARE PRODUCT*).

3.2. Other relevant practices

Besides the provided mapping, specific further inputs were identified according to e.g., innovation or security. Thus, a brief description of these other useful principles (Fig. 6) within the scope of medical device software are reported below.

3.2.1. MDCG guidelines

To ensure the correct application of the requirements of the MDR for a SaMD, a set of guidelines published by the Medical Device Coordination Group (MDCG) are available.

In particular, the MDCG 2019–11 guideline "Guidance on Qualification and Classification of Software in Regulation (EU) 2017/745 – MDR and Regulation (EU) 2017/746 – IVDR" provides a useful tool to determine whether the software meets the definition of medical device and falls within the scope of the Regulation and supports in the correct classification of the medical device that is fundamental for determining the regulatory path to be taken. Interestingly, the recent technical document about "Characterization Considerations for Medical Device

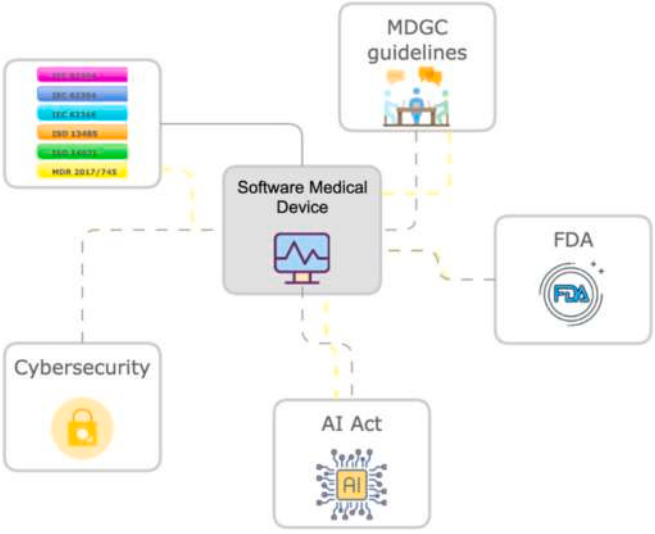

Fig. 6. Key inputs for SaMD lifecycle.

Software and Software-Specific Risk" provided by the IMDRF SaMD working group can further support the manufacturer [24].

In order to clearly and unambiguously identify a medical device software and facilitate its traceability, the MDCG 2018–5 guideline [25] offers a useful tool for the creation of the UDI (Unique Device Identifier) system.

The MDCG 2020–1 guideline [26], on the other hand, offers a useful framework for determining the level of clinical evidence required in the clinical evaluation of medical device software, to verify the absence of unacceptable clinical risks (safety), the ability to obtain the intended use declared by the manufacturer (performance) and the positive health impacts (clinical benefits).

Importantly, MDCG 2019–16 [27] "Guidance on Cybersecurity for medical devices" considers the relevant concepts about secure design, manufacture and surveillance, including an overview of cybersecurity risk management process and safety risk management relationship.

In case of software interacting with a medical device hardware to achieve its intended purpose, it is essential to consider the MDCG 2023–4 "Medical Device Software (MDSW) – Hardware combinations Guidance on MDSW intended to work in combination with hardware or hardware components"[28].

Finally, MDCG 2020–3 "Guidance on significant changes regarding the transitional provision under Article 120 of the MDR with regard to devices covered by certificates according to MDD or AIMDD" provides a useful overview of significant changes that require a careful assessment in terms of regulatory impact by the manufacturer [23].

3.2.2. FDA guidelines

The FDA (United States Food and Drug Administration) has published guidance documents specific to medical device software. The FDA uses the term SaMD (Software as a Medical Device) instead of medical device software. It defines SaMD as software intended to be used for one or more medical purposes that perform these purposes without being part of a hardware medical device. Contrary to the European regulatory framework, in the United States software intended to drive or influence the use of a medical device is not SaMD and it is not regulated.

The FDA uses the technical documents "Software as a Medical Device: Possible Framework for Risk Categorization and Corresponding Considerations" and "Software as a Medical Device (SaMD): Key Definitions" provided by the IMDRF to clarify what software is not in scope of its regulation [2930]; the "Quality Management System for Software as a Medical Device" document is also an important instrument for the application of medical device quality management system requirements to SaMD [31].

The FDA issued the "Software as a Medical Device (SAMD): Clinical Evaluation" guidance to describe the approach to clinical evaluation and to define principles for demonstrating safety, effectiveness, and performance [32].

3.2.3. Ai-based medical devices

When a medical device software meets the definition of an AI (Artificial Intelligence) based system, i.e., it is a software designed to operate with varying levels of autonomy and that can have the ability to adapt and that, for implicit or explicit objectives, deduces from the input it receives how to generate the output, it will have to comply with the EU Regulation 2024/1689, the AI Act [33]. If a software falls under both the AI Act's definition of AI and that of medical device software, it needs to be analyzed in terms of classification, defining whether it is a "high-risk" device. Specific obligations are provided for AI systems classified as high-risk; these include the implementation of a risk management system, transparency obligations toward users, and conformity assessment procedures for CE marking that may be handled through third-party bodies. A software manufactured in accordance with the MDR and qualified as a high-risk AI system according to the AI Act must meet the obligations under both regulations (the MDR and the AI Act).

In this scenario, it is necessary to coordinate the provisions set forth in the mentioned regulations in order to have a complementary and balanced set of norms to apply and to avoid contrast between them. Precisely to this end, the AI Act establishes in article 8 paragraph 2 that "in ensuring the compliance of high-risk AI systems (…) with the requirements set out in the AI Act, and in order to ensure consistency, avoid duplication and minimize additional burdens, providers shall have a choice of integrating, as appropriate, the necessary testing and reporting processes, information and documentation they provide with regard to their product into documentation and procedures that already exist and are required under the Union harmonisation legislation listed in Section A of Annex I." Guidelines, standards or technical documents like the ISO/IEC 42,001 "Information technology – Artificial intelligence – Management system"[34] or the recent "Good machine learning practice for medical device development: Guiding principles" are desirable and can facilitate a balanced approach in design and development of innovative solutions by the IMDRF Artificial Intelligence/Machine Learning-enabled Working Group [35].

More generally, it is paramount to underline that most of the provisions and obligations set forth by the AI Act are applicable in the case of high-risk AI system, and are devoted to establishing technical requirements for the design and manufacturing of the product, as well as duties for the economic operators involved in its entire life cycle that aim at ensuring a constant level of safety through time. Differently, the duties of the economic operators in case a non-high-risk AI system are fewer and mainly devoted to ensuring transparency of its use.

3.2.4. Cybersecurity

The MDR Regulation introduced new general security and performance requirements related to cybersecurity that are also applicable to medical device software, covering both pre-market and post-market aspects. Safety requirements must be identified and defined early in the product lifecycle. In addition to the MDCG guideline 2019–16, IEC 81001–5-1 makes it possible to respond to the requirements of the MDR.

The solutions adopted by the manufacturer for the design and manufacture of devices must comply with the principles of information security, considering the generally recognized state of the art. In addition, devices must be designed and manufactured to protect, as far as possible, from unauthorized access that could prevent the device from operating as intended. An appropriate security development process must be applied and the testing process needs to include evidence of the effective implementation of this type of requirements.

Assets, vulnerabilities, and security threats must be systematically analyzed and documented. It is also critical to assess the impact that loss of confidentiality, loss of integrity, or loss of availability could have on the security, effectiveness, or security of the data or system. Adequate cybersecurity information must be included in the documentation accompanying the software.

3.2.5. General data protection Regulation (GDPR)

It is of paramount importance to underline that in every step of the life cycle of a software of the kind here discussed any economic operator involved shall comply with the provisions set forth by Regulation 2016/679 GDPR[36] on the processing of personal data, according to the specific activity performed. Indeed, a software needs data during both its development and its functioning, and this data shall be processed according to the norms established by the GDPR.

Therefore, in order to ensure adequate compliance, it is always advisable to follow the mentioned Regulation from the very beginning of the software's life cycle. Coherently with this, but differently from the MDR, the AI Act includes norms specifically devoted to establishing requirements to be complied with when building the training data set (art. 10).

3.2.6. Regulation (EU) 2017/746

Finally, the focus of this work is SaMD lifecycle within the MDR 2017/745 framework, but it is worth mentioning that, in conjunction

with it, the In Vitro Diagnostic Medical Device Regulation (IVDR 2017/746), a regulation governing the placing on the market of in vitro diagnostic medical devices (IVDs), has entered into force. IVDR specifically addresses tools, "whether used alone or in combination, intended by the manufacturer to be used in vitro for the examination of specimens, including blood and tissue donations, derived from the human body" to provide information about a person's health or predisposition to disease. Similarly to what applies for MDR, software can be qualified as In Vitro Diagnostic Medical Device Software and classified according to the related risks [10].

4. Discussions

The normative landscape of SaMD is complex and fragmented. To our knowledge for the first time within this work, an integrated picture of key normative aspects related to the software life cycle was provided. The analysis was based on EU MDR and on standards that can be considered relevant to the sector, namely IEC 62304, IEC 82304, ISO 14971, IEC 62366, ISO 13485. The result is a complementary and interconnected set of normative conditions mapped on the product lifecycle main blocks, a unique and original supporting tool for effective design, development and maintenance of medical device software. This tool was enriched with an overview of further references, including guidelines related to the medical device software and cybersecurity/privacy regulation that might facilitate manufacturers, developers and innovators in providing evidence of compliance with requirements; the multiple regulatory aspects to be taken into account for AI based technology were also reported.

The adoption of this multifaceted framework provides a structured foundation that can support stakeholders on different levels of the SaMD development life cycle. Specifically:

i) Early-stage strategic alignment. The availability of a clear vision on the normative context since the design phase of an innovative device, can impact on a more effective planning. This framework enables developers and project managers to anticipate compliance requirements, allocate resources more efficiently, and align development strategies with long-term regulatory expectations. This proactive approach reduces the likelihood of costly re-design or regulatory setbacks later in the development process.

ii) *Support for new developers*. The provided tool that can facilitate the improvement of learning curve for software developers entering the medical device field dealing for the first time with a fragmented and complex scenario. The framework acts as a didactic and operational guide for software developers who may be unfamiliar with the intricacies of medical device regulation. Given the fragmented nature of the regulatory landscape—spanning international standards, risk management protocols, cybersecurity requirements, and quality assurance procedures—this resource can significantly ease the onboarding process and support faster adaptation to compliance practices.

iii) *Integrated development and compliance workflows*. This picture provides the possibility to implement project and production approaches integrating, as appropriate, requirements, testing and reporting processes. The approach encourages the embedding of regulatory consideration into project management and software engineering processes. This integration facilitates the concurrent development of requirements definition, verification and validation procedures, risk assessment, and documentation practices, thus supporting both regulatory readiness and agile software development approaches.

iv) *Enhanced stakeholder collaboration.* This work supports the possibility to facilitate communication and cooperation among the stakeholders through a consistent picture and common understanding across this area. By providing a coherent and shared representation of regulatory elements, interdisciplinary communication e.g. among developers, quality assurance teams, regulatory affairs professionals, and clinical stakeholders can be improved. This shared understanding promotes better coordination, minimizes misinterpretation of compliance needs, and helps align technical decisions with regulatory and clinical expectations throughout the product lifecycle.

As reported in the above-mentioned list, there are many advantages using this framework, segmented into separated stage of the lifecycle of a software medical device. Moreover, this mapping work could be translated into practical prescriptions for developers, suggesting a methodical approach for facing with the regulatory compliance aspect from the early beginning of an innovative idea to the maintenance of a placed-on-the-market device, satisfying customers' needs and mandatory requirements. A practical tip for developers is to translate the mentioned key points, graphically reported in figures and synthetically represented in the table in Annex A, into a check list to be used to demonstrate the evidence of conformity presumption to regulatory requirements, during internal audit or inspections by the Notified Body; this working methodology will allow manufacturers to provide, for each requirement, the proof of a well-structured approach to regulatory compliance in a complex and prescriptive environment.

A structured roadmap and process considering these inputs and their interconnection potentially allow manufacturers to navigate the regulatory landscape more effectively than considering separately the normative requirements, with a smoother interaction with the key actors [37], facilitating access to and adoption of safe and effective innovation and improving traceability on a three-fold level including design and testing, change control and post-market surveillance. This is particularly relevant for disruptive technologies like those based on AI, whose translation in clinical practice is extremely complex with highrisk of remaining limited. AI-based medical devices should be primarily considered SaMDs because their core functionality is driven by software algorithms. This approach ensures that this type of product is subjected to the same regulatory, risk and quality control that is essential for patient/user safety and effectiveness of the use. Accordingly, the need of a combined strategy, including different regulations, existing standards and continuously involving the key stakeholders is reported e.g., in the recent FUTURE-AI consensus paper [38] based on six guiding principles—fairness, universality, traceability, usability, robustness, and explainability aimed to overcome barriers about ethical, technical and clinical risks associated with healthcare AI.

It is worth noting that the effective adoption of an integrated normative framework pushes innovation also in terms of operative approaches that can be incorporated into procedures in order to facilitate e.g change control [3] or collection of evidence in terms of technical and clinical performance[39–41], resulting in a general advancement for the involved community. In fact, the provided picture clearly show the importance of – and can facilitate the integration of – feedback and learning processes within the life-cycle of a software. These aspects represent a crucial component in the development and maintenance of a medical device and can be supported by specific software tools designed to collect user input, monitor system performance, and enable rapid iteration. The implementation of feedback-driven development approaches enforces risk analysis and change management control ensuring patient safety and devices' performance, with a final positive effect on the clinical outcomes. The implication of this approach can result in different levels of impact depending on the device MDR risk classification. For example, for a Class III software with a direct role in diagnosis and treatment decisions, feedback and learning processes will probably require a stringent management, including that also apparently minor updates might need to go through updated regulatory steps. In general, risk management is not a one-time activity, but rather an evolving process informed by continuous feedback from verification activities, validation outcomes, real-world performance, and incident reporting. Segmenting the software into separated modules is a possible strategy for a better management of associated risks, that strongly impact on the device classification. Anyway, the framework suggested in this paper provides a general approach, able to cover the wide category of medical device software products. It is worth noting that a well-structured approach supports the demonstration of conformity to

Regulations and Standards and may lay the foundations for evolving features or changes of the use context that may bring the device in un upper risk class. As an example, consider a product designed to monitor physiological processes that is classified as Class IIa: if changes in the monitored parameters could pose a danger to the patient, the product would then be classified as Class IIb. It is advisable, in the digital healthcare revolution we are experiencing, to arrange, from the beginning, a medical device file and all the related documentation, which will be able to enclose, in a second moment, elements that are not initially considered; only a wide and forward-looking approach will avoid onerous changes, reducing the associated effort and resource need. Another relevant aspect to be considered is the possibility of cross-product learning, since issues. collected with one product can provide inputs for design and risk mitigation strategies in the development of new or updated devices. A solid organization's quality management system based on a clear vision of the specific regulatory references is central for this cross-product feedback mechanism. Post-market surveillance data can lead to re-evaluation of risk acceptability, prompting updates to both the risk control measures and documentation, but feedback loops can be derived also from earlier phases like usability evaluations, clinical studies, and early verification results: within this context the output of this work provides an opportunity to appropriately manage the non-linear, iterative nature of the software lifecycle as governed by regulatory standards. In addition, it is relevant to consider that feedback loops and agile methodologies have to co-exist and be effectively integrated with the rigorous documentation control, traceability, and validation requirements imposed by medical device standards. This integration is crucial to develop not only a high-quality technical implementation but also its alignment with the organizational workflows and responsibilities of all the stakeholders involved, in order to make innovation concretely and timely available for the end-users.

5. Conclusions

In the era of Agile in healthcare, regulation and innovation concepts and timings need to be more aligned, enabling harmonized communication and collaboration between team members while responding promptly, through transparency and accountability, to emerging and urgent clinical needs. The picture within this work opens the door to a smoother management of the SaMD lifecycle, including key aspects, such as risk management and implementation of usability processes, and their inclusion in innovation's design, and ensuring the priority of regulatory science, namely safety for the end user.

CRediT authorship contribution statement

Martina Francesconi: Writing – review & editing, Writing – original draft, Methodology, Conceptualization. **Miriam Cangi:** Writing – review & editing, Writing – original draft, Methodology. **Silvia Tamarri:** Writing – review & editing, Methodology. **Noemi Conditi:** Writing – review & editing, Methodology. **Chiara Menicucci:** Writing – review & editing, Methodology. **Alice Ravizza:** Writing – review & editing, Methodology. **Luisa Cattaneo:** Writing – review & editing, Methodology. **Elisabetta Bianchini:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: [Elisabetta Bianchini is co-founder of QUIPU s.r.l., Pisa, Italy a spin-off company of the Italian National Research Council and the University of Pisa developing SaMDs.]

Acknowledgements

The idea behind this paper was presented and awarded within the Health Technology Challenge 2021 of AIIC (Italian Association of Clinical Engineers).

Silvia Tamarri is the General Director of THEMA s.r.l, Imola, a company that provides strategic-regulatory consulting services to companies operating in the medical device, in vitro diagnostic medical device, and related regulated fields.

Miriam Cangi is a Project Specialist of THEMA s.r.l., Imola, responsible for regulatory affairs activities on medical devices, assisting in regulatory submission and assure compliance with applicable medical device regulations per jurisdiction, guidance and standards.

Appendix

Table AI. Summary table outlining the specific normative references associated with each lifecycle phase.

	MDR 2017/745	ISO 14,971	ISO 13,485	IEC 62,366	IEC 62,304	IEC 82,304
Qualification and Classification	■ Article 2 Definitions ■ Annex VIII Rule 11	■ 4.4 Risk management plan	■ 7.2 Customer-related processes ■ 7.3.2 Design and Development planning	■ 5.1. Prepare USE SPECIFICATION	■ 4.3 Software safety classification ■ 5.1 Software development planning	■ 4.1 General Requirements and initial RISK ASSESSMENT
Software development and configuration management process	■ Article 27 Unique Device Identification system ■ Article 61 Clinical evaluation ■ ANNEX I GENERAL SAFETY AND PERFORMANCE REQUIREMENTS	■ 5.4 Identification of hazards and hazardous situations ■ 5.5 Risk estimation ■ 7.1 Risk control options analysis ■ 7.2 Implementation of risk control measures	■ 7.3.4 Design and development outputs ■ 7.3.5 Design and development review ■ 7.3.6 Design and development verification	■ 5.2 Identify USER INTERFACE characteristics related to SAFETY and potential USE ERRORS ■ 5.3 Identify known or foreseeable HAZARDS and	■ 5.2 Software Requirements Specification ■ 5.3 Software Architectural Design ■ 7.1 Analysis of software contributing to hazardous situations	■ 4.2 HEALTH SOFTWARE PRODUCT use requirement ■ 4.5 system requirements ■ 4.3 VERIFICATION of HEALTH SOFTWARE PRODUCT use requirements

(*continued on next page*)

(*continued*)

	MDR 2017/745	ISO 14,971	ISO 13,485	IEC 62,366	IEC 62,304	IEC 82,304
		■ 7.5 Risk arising from risk control measures ■ 9 Risk management review		HAZARDOUS SITUATIONS ■ 4.1.2 RISK CONTROL as it relates to USER INTERFACE design ■ 5.7 Establish USER INTERFACE EVALUATION plan ■ 5.8 Perform USER INTERFACE design, implementation and FORMATIVE EVALUATION	■ 7.2 RISK CONTROL measures ■ 7.3 VERIFICATION of RISK CONTROL measures ■ 5.7 Software system testing ■ 8.1 Configuration identification	■ 4.6 VERIFICATION of system requirements ■ 6.1 VALIDATION plan
Software release	■ Article 28 UDI database ■ Article 61 Clinical evaluation ■ Annex II technical documentation	■ 4.5 Risk management file	■ 4.2.3 Medical Device File ■ 7.3.7 Design and Development Validation	■ 5.7.3 Performing SUMMATIVE EVALUATION of the usability of the user interface ■ 4.2 Usability engineering file	■ 5.7.5 Software system test records content ■ 5.8 Software release	■ 6.2 Performing validation ■ 6.3 Validation report ■ 7.2 Accompanying document
Software maintenance process and problem resolution	■ Article 52 Conformity assessment procedures ■ ANNEX I GENERAL SAFETY AND PERFORMANCE REQUIREMENTS	■ 4.1 Risk management process ■ 9 Risk management review ■ 10 Production and Post-Production information	■ 7.3.9 Control of Design and Development changes ■ 7.5.8 Identification	■ 4.1.1 Usability engineering process	■ 6.3 Modification implementation ■ 7.4 RISK MANAGEMENT of software changes ■ 8.2 Change control ■ 9 Software problem resolution PROCESS	■ 8.2 Software maintenance ■ 8.3 Revalidation
Post market surveillance	■ Article 83 Post-market surveillance system of the manufacturer ■ Annex VI Part C The UDI system ■ Article 28 UDI database	■ 10 Production and Post-Production information ■ 4.1 Risk management process	■ 4.2.5 Control of records ■ 7.5.9 Traceability ■ 8.2.3 Reporting to regulatory authorities ■ 8.3 Control of non-conforming products	■ 4.1.1 Usability engineering process	■ 4.1 Quality management system ■ 4.2 Risk management	■ 8.4 Post-market communication on the HEALTH SOFTWARE PRODUCT
Software disposal						■ 8.5 Decommissioning and disposal of the HEALTH SOFTWARE PRODUCT

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jbi.2025.104856.

References

[1] J. Yu, J. Zhang, S. Sengoku, Innovation Process and Industrial System of US Food and Drug Administration-Approved Software as a Medical Device: Review and Content Analysis, J. Med. Internet Res. 25 (2023), https://doi.org/10.2196/47505.

[2] S. Meister, W. Deiters, S. Becker, Digital health and digital biomarkers – Enabling value chains on health data, Curr. Dir. Biomed. Eng. 2 (2016) 577–581, https://doi.org/10.1515/CDBME-2016-0128/HTML.

[3] M.R. Martina, E. Bianchini, S. Sinceri, M. Francesconi, V. Gemignani, Software medical device maintenance: DevOps based approach for problem and modification management, J. Softw. Evol. Process. (2023), https://doi.org/10.1002/SMR.2570.

[4] A. Ss, The Essential Principles of Safety and Effectiveness for Medical Devices and the Role of Standards, Med. Devices (auckl) 13 (2020) 49–55, https://doi.org/10.2147/MDER.S235467.

[5] K. Zhou, G. Gattinger, The Evolving Regulatory Paradigm of AI in MedTech: A Review of Perspectives and Where We Are Today, Ther. Innov. Regul. Sci. 58 (2024) 456–464, https://doi.org/10.1007/s43441-024-00628-3.

[6] U.J. Muehlematter, P. Daniore, K.N. Vokinger, Approval of artificial intelligence and machine learning-based medical devices in the USA and Europe (2015–20): a

comparative analysis, Lancet Digit. Heal. 3 (2021) e195–e203, https://doi.org/10.1016/S2589-7500(20)30292-2.
[7] EUR-Lex - 02017R0745-20200424 - EN - EUR-Lex, (n.d.). https://eur-lex.europa.eu/legal-content/IT/TXT/?uri=CELEX%3A02017R0745-20200424 (accessed January 10, 2025).
[8] E. Bianchini, C.C. Mayer, Medical Device Regulation: Should We Care About It? Artery Res. 28 (2022) 55–60, https://doi.org/10.1007/s44200-022-00014-0.
[9] M. Bretthauer, S. Gerke, C. Hassan, O.F. Ahmad, Y. Mori, The New European Medical Device Regulation: Balancing Innovation and Patient Safety, Ann. Intern. Med. 176 (2023) 844–848, https://doi.org/10.7326/M23-0454.
[10] Guidance on Qualification and Classification of Software in Regulation (EU) 2017/745 – MDR and Regulation (EU) 2017/746 – IVDR - DocsRoom - European Commission, (2019). https://ec.europa.eu/docsroom/documents/37581 (accessed December 10, 2021).
[11] T. Melvin, M. Torre, New medical device regulations: the regulator's view, EFORT Open Rev. 4 (2019) 351–356, https://doi.org/10.1302/2058-5241.4.180061.
[12] T. Melvin, The European Medical Device Regulation-What Biomedical Engineers Need to Know, IEEE J. Transl. Eng. Heal. Med. 10 (2022), https://doi.org/10.1109/JTEHM.2022.3194415.
[13] F. Garzotto, R.I. Comoretto, L. Dorigo, D. Gregori, A. Zotti, G. Meneghesso, G. Gerosa, M. Bonin, Preparing healthcare, academic institutions, and notified bodies for their involvement in the innovation of medical devices under the new European regulation, Expert Rev. Med. Devices. 19 (2022) 613–621, https://doi.org/10.1080/17434440.2022.2118046.
[14] IEC 62304:2006 - Medical device software — Software life cycle processes, (n.d.). https://www.iso.org/standard/38421.html#amendment (accessed January 10, 2025).
[15] IEC 82304-1:2016 - Health software — Part 1: General requirements for product safety, (n.d.). https://www.iso.org/standard/59543.html (accessed January 10, 2025).
[16] MDCG 2019-11 Guidance on Qualification and Classification of Software in Regulation (EU) 2017/745 – MDR and Regulation (EU) 2017/746 – IVDR , (n.d.). https://ec.europa.eu/docsroom/documents/37581 (accessed January 10, 2025).
[17] A. Ravizza, F. Sternini, A. Giannini, F. Molinari, Methods for Preclinical Validation of Software as a Medical Device, Heal. 2020 - 13th Int. Conf. Heal. Informatics, Proceedings; Part 13th Int. Jt. Conf. Biomed. Eng. Syst. Technol. BIOSTEC 2020. (2023) 648–655. Doi: 10.5220/0009155406480655.
[18] ISO 14971:2019 - Medical devices — Application of risk management to medical devices, (n.d.). https://www.iso.org/standard/72704.html (accessed January 10, 2025).
[19] IEC 62366-1:2015 - Medical devices — Part 1: Application of usability engineering to medical devices, (n.d.). https://www.iso.org/standard/63179.html (accessed January 10, 2025).
[20] ISO 13485:2016 - Medical devices — Quality management systems — Requirements for regulatory purposes, (n.d.). https://www.iso.org/standard/59752.html (accessed January 10, 2025).
[21] Iso, tr,, accessed February 8, - Medical Devices — Guidance on the Application of ISO 14971, (n.d.). h 24971 (2025) 2020, https://www.iso.org/standard/74437.html.
[22] Iec, tr,, accessed February 8, - Medical Device Software — Part 1: Guidance on the Application of ISO 14971 to Medical Device Software, (n.d.). h 80002–1 (2025) 2009, https://www.iso.org/standard/54146.html.
[23] UPDATE - MDCG 2020-3 Rev.1 - Guidance on significant changes regarding the transitional provision under Article 120 of the MDR - May 2023 - European Commission, (n.d.). https://health.ec.europa.eu/latest-updates/update-mdcg-2020-3-rev1-guidance-significant-changes-regarding-transitional-provision-under-article-2023-05-12_en (accessed February 9, 2025).
[24] IMDRF, Characterization Considerations for Medical Device Software and Software-Specific Risk, (n.d.).
[25] MDCG 2018-5 UDI Assignment to Medical Device Software , (n.d.). https://ec.europa.eu/docsroom/documents/31926 (accessed January 10, 2025).
[26] DocsRMDCG 2020-1 Guidance on Clinical Evaluation (MDR)/ Performance Evaluation (IVDR) of Medical Device Software , (n.d.). https://ec.europa.eu/docsroom/documents/40323 (accessed January 10, 2025).
[27] MDCG 2019-16 - Guidance on Cybersecurity for medical devices, (n.d.). https://ec.europa.eu/docsroom/documents/41863.
[28] MDCG 2023-4 - Medical Device Software (MDSW) – Hardware combinations Guidance on MDSW intended to work in combination with hardware or hardware components - European Commission, (n.d.). https://health.ec.europa.eu/latest-updates/mdcg-2023-4-medical-device-software-mdsw-hardware-combinations-guidance-mdsw-intended-work-2023-10-18_en (accessed January 10, 2025).
[29] Software as a Medical Device: Possible Framework for Risk Categorization and Corresponding Considerations | International Medical Device Regulators Forum, (n.d.). https://www.imdrf.org/documents/software-medical-device-possible-framework-risk-categorization-and-corresponding-considerations (accessed January 10, 2025).
[30] Software as a Medical Device (SaMD): Key Definitions | International Medical Device Regulators Forum, (n.d.). https://www.imdrf.org/documents/software-medical-device-samd-key-definitions (accessed January 10, 2025).
[31] Software as a Medical Device (SaMD): Application of Quality Management System | International Medical Device Regulators Forum, (n.d.). https://www.imdrf.org/documents/software-medical-device-samd-application-quality-management-system (accessed January 10, 2025).
[32] Software as a Medical Device (SAMD): Clinical Evaluation | FDA, (n.d.). https://www.fda.gov/regulatory-information/search-fda-guidance-documents/software-medical-device-samd-clinical-evaluation (accessed January 10, 2025).
[33] REGULATION (EU) 2024/1689 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 13 June 2024 laying down harmonised rules on artificial intelligence., (n.d.). https://eur-lex.europa.eu/eli/reg/2024/1689/oj/eng (accessed January 10, 2025).
[34] ISO/IEC 42001:2023 Information technology — Artificial intelligence — Management system, (n.d.). https://www.iso.org/standard/81230.html.
[35] IMDRF, Good machine learning practice for medical device development: Guiding principles, (n.d.).
[36] General Data Protection Regulation (GDPR) – Legal Text, (n.d.). https://gdpr-info.eu/ (accessed February 9, 2025).
[37] K. Woudstra, R. Reuzel, M. Rovers, M. Tummers, An Overview of Stakeholders, Methods, Topics, and Challenges in Participatory Approaches Used in the Development of Medical Devices: A Scoping Review, Int. J. Heal. Policy Manag. 12 (2023) 1–8, https://doi.org/10.34172/IJHPM.2022.6839.
[38] K. Lekadir, A.F. Frangi, A.R. Porras, B. Glocker, C. Cintas, C.P. Langlotz, E. Weicken, F.W. Asselbergs, F. Prior, G.S. Collins, G. Kaissis, G. Tsakou, I. Buvat, J. Kalpathy-Cramer, J. Mongan, J.A. Schnabel, K. Kushibar, K. Riklund, K. Marias, L.M. Amugongo, L.A. Fromont, L. Maier-Hein, L. Cerdá-Alberich, L. Martí-Bonmatí, M.J. Cardoso, M. Bobowicz, M. Shabani, M. Tsiknakis, M.A. Zuluaga, M.-C. Fritzsche, M. Camacho, M.G. Linguraru, M. Wenzel, M. De Bruijne, M. G. Tolsgaard, M. Goisauf, M.C. Abadía, N. Papanikolaou, N. Lazrak, O. Pujol, R. Osuala, S. Napel, S. Colantonio, S. Joshi, S. Klein, S. Aussó, W.A. Rogers, Z. Salahuddin, M.P.A. Starmans, FUTURE-AI: international consensus guideline for trustworthy and deployable artificial intelligence in healthcare, BMJ- 388 (2025) e081554, https://doi.org/10.1136/BMJ-2024-081554.
[39] E. Bianchini, M. Francesconi, M. Testa, M. Tanase, V. Gemignani, Unique device identification and traceability for medical software: A major challenge for manufacturers in an ever-evolving marketplace, J. Biomed. Inform. 93 (2019), https://doi.org/10.1016/j.jbi.2019.103150.
[40] F.E. Rademakers, E. Biasin, N. Bruining, E.G. Caiani, R.H. Davies, S.H. Gilbert, E. Kamenjasevic, G. McGauran, G. O'Connor, J.-B. Rouffet, B. Vasey, A.G. Fraser, CORE-MD clinical risk score for regulatory evaluation of artificial intelligence-based medical device software, NPJ Digit. Med. 8 (2025) 90, https://doi.org/10.1038/S41746-025-01459-8.
[41] E.G. Caiani, H. Kemps, P. Hoogendoorn, R. Asteggiano, A. Böhm, B. Borregaard, G. Boriani, H.-P. Brunner La Rocca, R. Casado-Arroyo, S. Castelletti, R. M. Christodorescu, M.R. Cowie, P. Dendale, F. Dunn, A.G. Fraser, D.A. Lane, E. T. Locati, K. Małaczyńska-Rajpold, C.O. Merşa, L. Neubeck, G. Parati, C. Plummer, G. Rosano, M. Scherrenberg, A. Smirthwaite, P. Szymanski, Standardized assessment of evidence supporting the adoption of mobile health solutions: A Clinical Consensus Statement of the ESC Regulatory Affairs Committee: Developed in collaboration with the European Heart Rhythm Association (EHRA), the Association of Cardiovascular Nursing & Allied Professions (ACNAP) of the ESC, the Heart Failure Association (HFA) of the ESC, the ESC Young Community, the ESC Working Group on e-Cardiology, the ESC Council for Cardiology Practice, the ESC Council of Cardio-, Eur. Hear. Journal. Digit. Heal. 5 (2024), https://doi.org/10.1093/EHJDH/ZTAE042.