Addressing the AI-enabled Medical Device Software Development Life Cycle: A Review and Integration of IEC 62304 and ISO 5338

Karla Aniela Cepeda-Zapata[1[0000-0003-0551-0897]], Zaffar Haider Janjua[1[0000-0002-6331-0140]], Róisín Loughran[1[0000-0002-0974-7106]], Tomás Ward[2[0000-0002-6173-6607]], and Fergal McCaffery[1[0000-0002-0839-8362]]

[1] Dundalk Institute of Technology, Dundalk, Ireland
{karla.cepeda, zaffar.haider, roisin.loughran, fergal.mccaffery}@dkit.ie
[2] Dublin City University, Dublin, Ireland
tomas.ward@dcu.ie

Abstract. International standards are essential for harmonising best practices to ensure the quality of products and services. IEC 62304 is a recognised reference standard for defining development life cycle (DLC) activities for medical device software (MDS). However, given the growing adoption of AI within this sector, it is essential to embed AI-specific requirements to guide the development of AI-enabled Medical Device Software (AIeD). As part of previous work, this paper reviews and maps IEC 62304:2006+A1:2015 and ISO/IEC 5338:2023, a process standard that provides generic life cycle processes for AI systems. These two standards are derived from ISO/IEC/IEEE 12207 and share elements that can be integrated to propose an initial DLC framework for AIeD. The integration involves joining process IDs and identifying relevant keywords from each activity/task. Fourteen processes from ISO 5338 are aligned across activities in the IEC 62304 development process, including two AI-specific processes: Data Engineering and Knowledge Acquisition. Additionally, +11 AI-specific tasks and +24 AI-specific guidelines are identified and aligned. Future work will incorporate additional relevant standards and regulatory considerations to further detail and support this framework. Lessons from this paper could stimulate future research to automate the integration of standards, thereby enabling easier compliance for MDS manufacturers.

Keywords: Medical Device Software, Artificial Intelligence systems, AI-enabled Medical Device Software, Development Process, Process Standards

1 Introduction

With the increasing adoption of Artificial Intelligence (AI) technology, it has become crucial to adopt appropriate software engineering best practices [1] to ensure trustworthiness, transparency, and traceability. In the Medical Device Software (MDS) industry, international standards play a crucial role in harmonising best practices to protect

and ensure that products and services are safe, reliable, and of high quality [2, 3]. Various standards provide best practices for software engineering processes. ISO 12207[1] defines the generic software system life cycle applicable to a wide range of software systems [4]. In the medical device sector, standards are not mandatory. However, recognised standards can be used to establish a presumption of compliance with relevant regulatory requirements [3, 5, 6]. One of these standards is IEC 62304[2] for the MDS development process, recognised by the FDA [7] and supporting the UK MDR 2002 [8], a fundamental process standard. More recently, ISO 5338[3] has been introduced to extend the traditional life cycle framework to account for the particularities of AI systems [9]. While IEC 62304 is effective for conventional MDS development [10], there is no framework to support the development life cycle (DLC) of AI-enabled Medical Device Software (AIeD). Furthermore, while ISO 5338 provides a generic life-cycle framework for AI systems, it also requires additional requirements for the MDS sector [9]. This gap highlights the need for a harmonised approach that supports manufacturers throughout the AIeD development process.

This paper continues the work presented in [11], which hypothesised that integrating IEC 62304 and ISO 5338 via ISO 12207 could facilitate the initial proposal of a DLC process for AIeD. The objective of this paper is to review and align the processes of IEC 62304 and ISO 5338 to define a preliminary DLC framework for AIeD, with a focus on data-driven AI models. While this study focuses on three standards, future work will expand the review to include other standards identified in [11]. The structure of the paper is as follows: Section 2 discusses the MDS development life cycle and relevant ISO standards; Section 3 presents life cycle processes of AI systems; Section 4 reviews the underlying structure of IEC 62304 and ISO 5338; Section 5 introduces the methodology implemented for extraction and integration of the standards; Section 6 presents processes aligned; Section 7 discusses the results obtain; and Section 8 concludes the paper including lessons learnt and future work.

2 Medical Device Software Development Life Cycle

IEC 62304 is likely the most widely referenced standard in the medical industry sector [10]. It specifies the activities and tasks required for the development and maintenance of MDS, whether embedded in medical hardware or functioning as standalone software. Built upon ISO 12207:2008 and tailored for safety-critical MDS technology, it defines five top-level process groups: (1) *Development*, covering software development planning, requirements analysis, architectural design, detailed design, unit implementation and verification, integration and integration testing, system testing, and release; (2) *Maintenance*, including maintenance planning, problem and modification analysis, and change implementation; (3) *Risk management*, addressing the identification of software-related hazardous situations, implementation of risk control measures, and

[1] Two versions are considered: ISO/IEC/IEEE 12207:2008 and ISO/IEC/IEEE 12207:2017. The year is included when necessary to distinguish them.
[2] The version used in this paper is IEC 62304:2006+A1:2015.
[3] The version used in this paper is ISO/IEC 5338:2023.

verification of their effectiveness in alignment with ISO 14971; (4) *Configuration management*, ensuring identification, change control, and status accounting of software items; and (5) *Problem resolution*, covering problem reporting, investigation, resolution, verification, and trend analysis. It also mandates general requirements such as Quality Management System integration, safety classification (Classes A–C), and treatment of legacy software. IEC 62304 is life-cycle-process agnostic, supporting sequential, iterative, or agile approaches, while embedding software safety classification-driven rigour, complete traceability from requirements to verification, and continuous risk control across all process stages to meet regulatory and clinical safety expectations.

3 Artificial Intelligence Life Cycle Framework

ISO 5338 defines a comprehensive AI system life cycle process model, and its foundation is based on ISO/IEC/IEEE 15288:2023 and ISO 12207:2017 [9, 12, 13]. This standard incorporates the modification and addition of AI-specific aspects, such as Data Engineering and Continuous Validation, to extend the traditional software life cycle process [12]. ISO 5338 categorises processes into three types: *generic*, identified as traditional; *modified*, where elements are modified, added, or removed from traditional processes; and *AI-specific*, new processes developed for AI systems [12]. ISO 5338 does not prescribe a life cycle model, but it provides an interpretation within stages, as per ISO 22989:2023 [12, 13].

Additionally, processes are organized into four process groups: (1) Agreement process group, including Acquisition and Supply extended for AI data sourcing and model provision; (2) Organizational project-enabling process group, including Life-cycle model management, Infrastructure management, Portfolio management, Human resource management, Quality management, and Knowledge management with AI-specific requirements for dataset lineage and experimental reproducibility; (3) Technical management process group, such as Project planning, Project assessment and control, Decision management, Risk management, Configuration management, Information management, Measurement, and Quality assurance; and (4) Technical process group, including Business/mission analysis, Stakeholder needs definition, System requirements analysis, Architecture and design definition, System analysis, Knowledge acquisition (AI-specific process), AI data engineering (AI-specific process), Implementation (including AI-specific tasks), Integration, Verification, Transition, Validation, Continuous validation (AI-specific process), Operation, Maintenance, and Disposal.

AI-specific aspects are embedded alongside generic software engineering activities because AI software is often treated as professional software [13]. ISO 5338 establishes a life-cycle framework for AI-specific processes and activities that can also serve as a checklist for proper governance when building AI systems [13]. As this standard is based on established software practices, ISO 5338 could also be utilised to facilitate the integration of AI-specific aspects into the MDS industry.

4 Process Description of IEC 62304 and ISO 5338

IEC 62304 and ISO 5338 are standards that describe processes. A process can be defined as a set of interrelated activities in which inputs are transformed into outputs [4, 14]. In ISO 12207:2017, processes can be decomposed into process constructs, namely *process groups, processes, activities* and *tasks*. Process constructs might also have attributes, including a *name*, *purpose,* and *outcome*. Each process construct can be denoted with a *clause numbering* [4], which can be referred to as an identification number (ID). The super-type in ISO 12207:2017 is *Process Group*, where processes are clustered as follows: Agreement, Organisational Project-Enabling, Technical Management, and Technical. To represent process constructs in this section, UML diagrams and Class Notation are utilised. The process constructs in ISO 12207:2017, and their relationships are available in [4, 15].

ISO 12207:2017 and ISO 5338 share similar process constructs, but other elements can also be included. One is the *type* of process, which is based on the addition of AI-specific elements [9]. In other words, there is a *modified* or *AI-specific* process which might contain *AI-specific activities* or *guidelines*. Additionally, ISO 5338 outlines processes within *AI Stages*, as portrayed in ISO 22989:2023. A stage is defined as the major life-cycle period associated with a system of interest, providing a high level of visibility and control over the processes [16]. Stages are not considered within ISO 12207:2017[4]. However, for this paper, an AI Stage is included as a process construct. AI-specific guidelines are also included as a process construct. As in ISO 12207:2017, clause numbering is also used as a process identifier. Another particularity of ISO 5338 is the unstructured arrangement of tasks and guidelines, which lack IDs and distinct names. Additionally, activities in ISO 5338 are defined only for modified and AI-specific processes, whereas for generic ones, activities must be reviewed in accordance with ISO 12207:2017. AI-specific tasks in ISO 5338 also lack notes. The super-type in ISO 5338 is *Process Group*, similar to ISO 12207:2017. The process constructs of ISO 5338 are shown in Fig. 1.

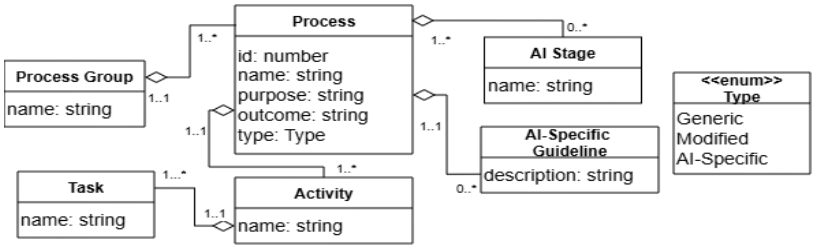

Fig. 1. Diagram representing the process constructs of ISO 5338, based on [15].

When comparing IEC 62304 process constructs against ISO 12207:2017 and ISO 5338, these standards share a similar underlying structure, though with some differences. IEC 62304 does not have Process Groups; the highest level is Processes.

[4] There are references on how to implement stages for software life cycle processes in ISO/IEC/IEEE 24748-1:2024 and ISO/IEC/IEEE 24748-3:2020.

Moreover, processes in IEC 62304 lack a defined purpose and outcome, unlike those in ISO 12207 and ISO 5338. It is also observed that Tasks can be subdivided into subtasks, referred to in this paper as *provisions*, which are considered the lowest process construct in IEC 62304. The process constructs of IEC 62304 are illustrated in Fig. 2, where clause numbering can also serve as an identifier for each process, activity, and task. Additionally, Safety Classification is included as a process construct within IEC 62304. This is primarily to highlight one of its differences from ISO 12207. Safety Classification identifies activities required based on the safety needed [17].

IEC 62304 provides a relationship table to ISO 12207:2008 in its Annexe C.6. This table can be used to align IEC 62304 to ISO 12207 by mapping IDs, providing an initial alignment to complement MDS development with AI-specific activities.

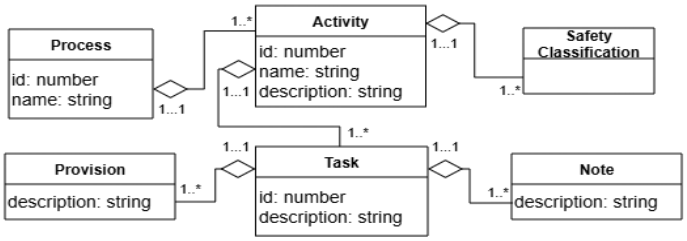

Fig. 2. Diagram representing process constructs of IEC 62304, based on [15].

5 Methodology

5.1 Introduction

A mixed-method approach is used to integrate the MDS development and the AI system life cycles. This involves leveraging the underlying structure of standards and existing relationships among them. As IEC 62304 and ISO 5338 are derived from ISO 12207:2008 and ISO 12207:2017, respectively, ISO/IEC/IEEE 12207-2[5] (ISO 12207-2) is utilised. The three major steps to integrate IEC 62304 and ISO 5338 through ISO 12207-2 are as follows:

1. Extract relevant information from each standard and save this information in an MS Excel file:
   a. Extract the DLC process content from IEC 62304,
   b. Extract processes, activities, and tasks from ISO 5338,
   c. Extract the relationship table to ISO 12207:2008 from IEC 62304,
   d. Extract relationship tables from ISO 12207-2; and,
   e. Manually create a relationship table between ISO 5338 and ISO 12207:2017.
2. Integrate information by aligning processes using IDs, particularly dependent on the existing relationship provided in IEC 62304 (in Annex C.6) to ISO 12207:2008. This integration is illustrated in Fig. 3.

[5] This standard provides a relationship table between ISO 12207:2008 and ISO 12207:2017, which is expected to facilitate the integration process for this paper.

3. Under processes aligned, keywords are identified for each task to enhance alignment. This is a relevant step for new AI-specific tasks that cannot be aligned with IEC 62304.

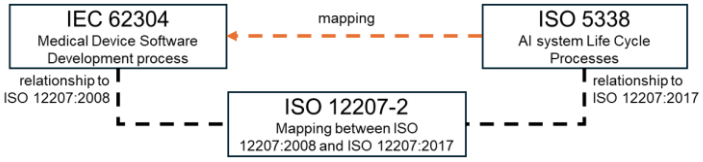

Fig. 3. Extraction example of data within standards using regex functions in Python.

Access to the standards is provided through our Institution, and copyright agreements are followed in accordance with the ISO guidelines [18]. The outcome of this paper is not intended to alter the content of IEC 62304, but rather to utilise existing work, i.e., IEC 62304 and ISO 5338, to propose a new framework, which is expected to evolve following a further review of standards, as identified in [11].

5.2 Content Extraction Process

The extraction of standards' content is performed to facilitate the mapping. This process involves extracting content from IEC 62304 and ISO 5338, which are referred to as *anchor* tables, and extracting information used to join them, referred to as *bridge* tables.

Anchor Tables. Process constructs are not presented in tabular format, which could have facilitated alignment of process IDs. **Error! Reference source not found.**Fig. 4 illustrates the method for extracting information from IEC 62304 and ISO 5338 using Python and regular expressions (regex). The extraction process begins by manually dividing the standard into the table of contents (ToC) and the body. The ToC serves as a "signpost" to identify and extract relevant information within the body. Regex is used to match process IDs from the ToC and transform them into a tabular format. Then, this is used to identify activities and tasks from the body by matching section names to data extracted from the ToC. The information extracted from this section is stored in MS Excel sheets, which serve as an *anchor* table.

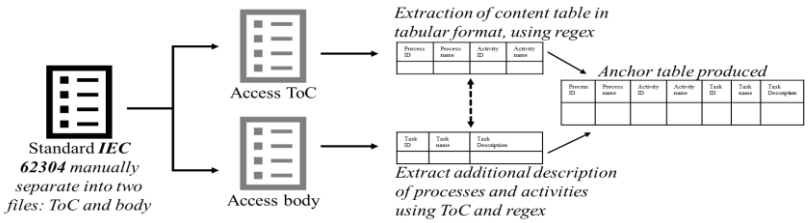

Fig. 4. Extraction example of data within standards using regex functions in Python.

Tasks and guidelines in ISO 5338 are organised in an unstructured approach, unlike IEC 62304. A large language model is implemented [19] to facilitate the identification

and extraction of tasks and guidelines in ISO 5338. Prompt is designed based on [19, 20]. This is locally implemented to promote the protection of copyrighted content [21]. The results are manually validated against the original source, stored in an MS Excel file, and integrated with the respective processes outlined in ISO 5338.

Bridge Tables. The ISO 12207-2 standard and Annex C.6 of IEC 62304 [17] are key to joining process IDs from IEC 62304 and ISO 5338. The latter provides a mapping from IEC 62304 to ISO 12207:2008, whereas the former provides a mapping for both ISO 12207:2008 and ISO 12207:2017. This information is presented in tables extracted with the Python library *pdfplumber*. Additional cleaning of the text is required to remove irrelevant Unicode characters, including line tabulation and newlines. Although ISO 5338 and ISO 12207:2017 have a similar structure, no table, as in IEC 62304, establishes a direct relationship between them. A table is manually created to explicitly define the relationship between ISO 5338 and ISO 12207:2017 by identifying process names and IDs. The information extracted in this section is referred to as *bridge* tables, which serve to connect the *anchor* tables.

5.3 Integration Process and Semantic Alignment

To initiate the mapping between IEC 62304 and ISO 5338, anchor and bridge tables are inspected to identify which process constructs can be linked using their respective IDs. The anchor tables are joined using the bridge tables first at the activity-process level, illustrated in Fig. 5, as it was found to be easier to achieve through IEC 62304 development activities aligned with ISO 12207 processes, as per Annexe C.6 of IEC 62304. After the alignment, the processes are also clustered into AI stages to see the distribution of ISO 5338 processes across IEC 62304.

It is challenging to integrate IEC 62304 and ISO 5338 at the task-activity level. This is primarily due to the unstructured presentation of tasks and guidelines in ISO 5338, as well as the introduction of new AI-specific processes. Therefore, to review and align the semantics of each AI-specific element, keywords are manually identified and extracted for completeness after alignment through process IDs. This manual method is applied to processes aligned with bridge tables, yielding a narrow, concise set to review manually.

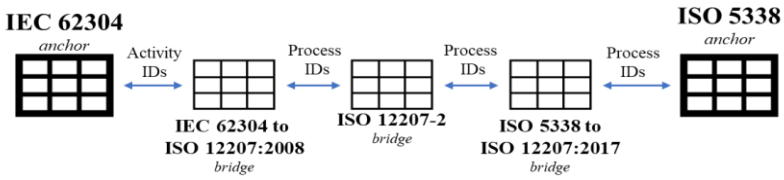

Fig. 5. Tables and the mapping process of anchor tables via bridge tables.

6 Results

The alignment of activity-process IDs from IEC 62304 to ISO 5338 is shown in Fig. 6. It has been identified that aligned ISO 5338 processes are not exclusive to a single IEC

62304 activity; some overlap across multiple IEC 62304 development activities. For instance, the Implementation process in ISO 5338 is located in both Software Development Planning and Software Unit Implementation activities of IEC 62304. In total, 15 unique ISO 5338 processes are aligned across the IEC 62304 development activities. Though 18 processes (see None in Fig.8) are not aligned to IEC 62304 due to two reasons: (1) the nature of ISO 5338 as an AI-specific standard, which includes AI-specific processes not connected to IEC 62304; and (2) not all processes in the ISO 12207 are part of IEC 62304. These AI-specific processes (1), which are not aligned with IEC 62304, are Data engineering, Knowledge acquisition, and Continuous validation. Therefore, it is not surprising that some processes are omitted once the anchor and bridge tables are joined.

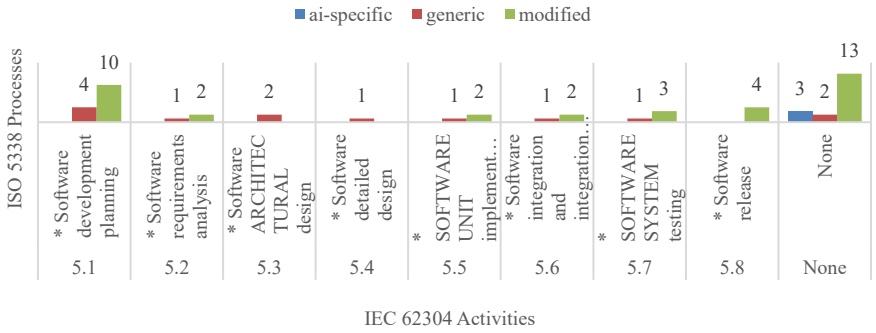

Fig. 6. Number of processes in the ISO 5338 mapped to the IEC 62304. Processes are categorised as AI-specific, generic, and modified, as defined in ISO 5338.

The aligned processes are broken down into AI stages in accordance with ISO 5338. Because this life cycle model focuses on the technical process group, some processes are shown in Fig. 6 are excluded. The technical processes and respective AI stages are shown in Fig. 7. Three AI stages are not aligned in any of the IEC 62304 development activities: Retirement, Continuous validation, and Deployment. It is also worth noting that the AI stage, *Design and Development*, is present across all IEC 62304 development activities.

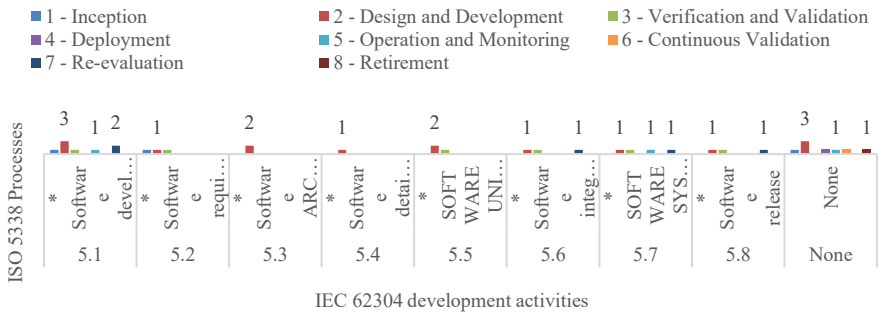

Fig. 7. ISO 5338 processes by AI Stage mapped to IEC 62304 Development Activities.

AI-specific processes in ISO 5338 could not be directly aligned with IEC 62304 development activities using process IDs. This outcome is not surprising, as these processes are neither present in nor related to the content detailed in IEC 62304's Annexe C.6 documentation. As the scope of this work is on the development activities of IEC 62304, the AI stage Design and Development is considered to identify relevant new AI-specific processes for AIeD. In this case, *Data Engineering* and *Knowledge Acquisition* are included as these processes are part of the AI stage Design and Development. By identifying processes and their respective stages, each task and guideline in ISO 5338 is analysed, and keywords are identified to evaluate and refine the aligned processes. The number of tasks and guidelines aligned to each development activity from IEC 62304 is shown in Table 1. The last column of this table indicates whether each MDS activity is adapted, generic, or new for AIeD development.

Table 1. Number of rows extracted from the Content Extraction Process.

IEC 62304 Dev. Act.	ISO 5338 processes	+AI-specific Tasks & Guidelines	Type of Dev. Act. AIeD
5.1 SW Dev. Planning	6.3.1 Project Planning (M) 6.3.2 Project Assessment and Control (M) 6.3.6 Information Management (M) 6.3.5 Configuration Management (M) 6.3.8 Quality Assurance (M) 6.3.8 Operation (M) 6.4.11 Verification (M) 6.4.10 Integration (G)	+ nine guidelines	Adapted
5.2 SW Requirements Analysis	6.4.3 System requirement Definition (M)	+ four guidelines	Adapted
5.3 SW Architectural Design	6.4.4 System Architecture Definition (G)	No task/guideline identified	Generic
5.4 SW Detailed Design	6.4.5 Design Definition (G)	No task/guideline identified	Generic
5.5 SW Unit Implementation	6.4.9 Implementation (M) 6.4.7 Knowledge Acquisition (AI) 6.4.11 Verification (M)	+ four tasks + five guidelines	Adapted
5.6 SW Integration and Integration Testing	6.4.11 Verification (M) 6.4.10 Integration (G)	+ one guideline	Adapted
5.7 SW System Testing	6.4.11 Verification (M)	+ one guideline	Adapted
5.8 SW Release	6.4.11 Verification (M) 6.3.5 Configuration Management (M)	+ two guidelines	Adapted
Data Engineering	6.4.8 Data Engineering (AI) 6.3.8 Operation (M) 6.3.8 Quality Assurance (M) 6.4.7 Knowledge Acquisition (AI)	+ seven activities + two guidelines	New

Dev. = Development, Act. = Activity, += additions, (M) = Modified, (G) = Generic, (AI) = AI-specific, = Inception, = Design and Development, = Verification & Validation, = Operation and Monitoring, = Outside AI Stages

7 Discussion

The integration of IEC 62304 and ISO 5338 through ISO 12207 provides a preliminary version of a DLC framework for AIeD, addressing the current gap between MDS development and AI-specific life cycle processes. Based on the results from Section 6, the proposed development process to complement IEC 62304 with AI-specific activities, tasks, and guidelines is illustrated in the Fig. 8. This version proposes adapting the following activities: Software Development Planning, Software Requirements Analysis, Software Unit Implementation, Integration & Integration Testing, Software System Testing, and Release, to consider AI-specific aspects. *Unit Implementation* is a relevant modified activity that includes AI-specific tasks, such as algorithm selection, model training, and model evaluation.

Data Engineering is integrated into the IEC 62304 development process as a new activity, encompassing tasks such as data collection and data preparation. As the focus of this work is on data-driven AI models, *Knowledge Acquisition* is kept at a task level to support Data Engineering and Implementation activities. Two activities remain generic for the development of AIeD, namely Software Architectural Design and Detailed Design, as no tasks/guidelines are aligned to these development activities in IEC 62304. Through this integration, it is also observed that Risk Management and Software Configuration Management should be tailored to AI-specific requirements. In the context of Risk Management, risks associated with AI systems, in line with ISO 14971 and ISO/IEC 23894, should be considered for AI systems [9].

The AIeD activities are illustrated in Fig. 8. Data Engineering is added close to the Implementation activity, as the data is needed to train and test an AI model. Therefore, Data Engineering should occur before Implementation. However, there is no suggestion of following a waterfall methodology for AIeD, which would be impractical given the iterative, experimental nature of building AI models.

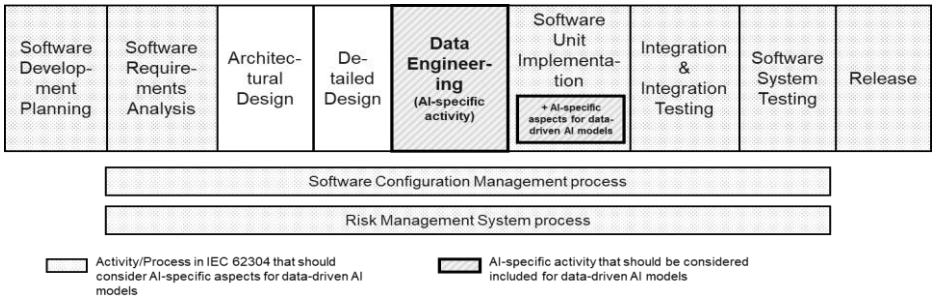

Fig. 8. First proposed for the Development life cycle process for AIeD.

8 Conclusions

This paper reviewed and integrated ISO 5338 processes into the IEC 62304 development process, allowing for the proposal of a preliminary DLC framework for AIeD. This bridging exercise enabled the identification of equivalent and complementary

processes and activities to those outlined in IEC 62304. In future work, this framework will be expanded to incorporate additional content from standards identified[6] in [11] and further refined through a review of key regulatory requirements to elaborate on its implementation in real-world settings. The AIeD framework is expected to support and guide MDS manufacturers throughout the development process, enabling cross-domain traceability, linking AI processes and tasks within the MDS DLC. This ensures that AI-specific activities are embedded within the same regulatory evidence chain required for clinical safety. Moreover, it could improve efficiency by reducing duplicated effort through shared work items and artefacts, such as integrating risk registers and unified traceability matrices, thereby lowering costs and accelerating time-to-market. Additionally, it could strengthen regulatory preparedness by providing a foundation for a future harmonised life-cycle process that meets MDS compliance requirements.

Three lessons learned from this study can highlight methodological and practical insights. First, process mapping must go beyond predefined relationships and clause numbering. While clause numbering, as a process identifier in standards, offers a consistent and convenient traceability mechanism, effective integration requires identifying shared meaning, e.g., by inspecting semantics. For example, the Data Engineering in ISO 5338 is explicitly mapped to the MDS development process to ensure data quality before model training. Other examples include Continuous Validation with an explicit mapping to risk management processes for ongoing clinical safety assurance. Second, the integration exercise demonstrated the value of process harmonisation. Because standards use different terminology and levels of abstraction, extracted information had to be cleaned and normalised before meaningful mapping was possible. This also illustrates the importance of developing standardised ontologies or knowledge bases for cross-standard comparison. Third, manufacturers benefit from a lifecycle-model agnostic approach. Both ISO 62304 and ISO 5338 allow flexibility in life cycle models (e.g., agile, iterative, V-model), and integration through ISO 12207 preserves this flexibility, enabling adoption across organisations with diverse development cultures. We believe that lessons learned from this paper could stimulate future research efforts on automating the integration of processes, thereby enabling easy compliance for medical device manufacturers.

Acknowledgments. This work is supported by the HEA's Technological University Transformation Fund, Biodesign Europe, Dundalk Institute of Technology, DkIT Connect DAC, and partially supported by Research Ireland (Grant number 21/FFP-A/9255). This work is also partially supported by Recon4IMD, co-funded by the European Union's Horizon Europe Framework Programme (101080997), the Swiss State Secretariat for Education, Research and Innovation (23.00232), and by United Kingdom Research and Innovation (10083717 & 10080153).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

[6] Horizontal AI standard documents include: ISO 8183, ISO 22989, ISO 23053, ISO 5259 series, and ISO 24027. AI and Healthcare-related standards include: ANSI/CAT 2107, ANSI/CAT 2090, and IEEE 2801-2022. For more information, see previous work [11].

References

1. Menzies, T.: The Five Laws of SE for AI. IEEE Softw. 37, 81–85 (2020). https://doi.org/10.1109/MS.2019.2954841.
2. British Standards Institution (BSI): What is a standard?
3. FDA: Standards at FDA, https://www.fda.gov/regulatory-information/fda-standards-program/standards-fda, last accessed 2025/04/02.
4. BSI: ISO/IEC/IEEE 12207:2017 Systems and software engineering — Software life cycle processes. (2017).
5. European Commission: Single Market and Standards - Harmonised Standards, https://single-market-economy.ec.europa.eu/single-market/european-standards/harmonised-standards_en, last accessed 2023/08/22.
6. European Commission: Harmonised Standards, https://single-market-economy.ec.europa.eu/single-market/european-standards/harmonised-standards_en#:~:text=A%20harmonised%20standard%20is%20a,to%20one%20of%20these%20organisations., last accessed 2023/06/23.
7. U.S. FDA: Recognized Consensus Standards: Medical Devices - IEC 62304 Edition 1.1 2015-06. (2019).
8. Gov.UK: Designated standards: medical devices, https://www.gov.uk/government/publications/designated-standards-medical-devices, last accessed 2025/10/02.
9. BSI: ISO/IEC 5338:2023 Information technology — Artificial intelligence — AI system life cycle processes. (2023).
10. COCIR: Artificial Intelligence in EU Medical Device Legislation. (2020).
11. Cepeda-Zapata, K.A., Loughran, R., Ward, T., McCaffery, F.: A Review of AI Life Cycle-Related Standards to Address AI-Enabled Medical Device Development. Systems, Software and Services Process Improvement. (2026).
12. Cheng, Y.: Introduction of AI system life cycle processes. (2023).
13. Rob van der Veer: ISO/IEC 5338: Get to know the global standard on AI systems, https://www.softwareimprovementgroup.com/iso-5338-get-to-know-the-global-standard-on-ai-systems/, last accessed 2025/09/22.
14. International Organization for Standards: ISO/IEC/IEEE 24765:2017 Information Technology — Vocabulary — Fundamental terms. (1993).
15. Henderson-Sellers, B., Gonzalez-Perez, C., McBride, T., Low, G.: An ontology for ISO software engineering standards: 1) Creating the infrastructure. Comput Stand Interfaces. 36, 563–576 (2014). https://doi.org/10.1016/j.csi.2013.11.001.
16. BSI: BS ISO/IEC/IEEE 15288:2023 Systems and software engineering — System life cycle processes. (2023).
17. BSI: IEC 62304:2006+A1:2015 Medical device software — Software life-cycle processe. (2015).
18. IEC, ISO: How to best use IEC and ISO Standards - A user guide on licensing options and respecting copyright. (2019).
19. Fulford, I., Ng, A.: ChatGPT Prompt Engineering for Developers, https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/, last accessed 2025/09/26.
20. JohannesJolkkonen: LLM Powered Consultancy Graph Generation, https://github.com/JohannesJolkkonen/funktio-ai-samples/blob/main/knowledge-graph-demo/notebook.ipynb, last accessed 2025/10/02.
21. European Commission: Living guidelines on the responsible use of generative AI in research. , Brussels (2025).