

# Verification & Validation in Medi SPICE

M S Sivakumar, Valentine Casey, Fergal Mc Caffery, Gerry Coleman

Regulated Software Research Group,

Dundalk Institute of Technology & Lero, Ireland

{smadh09@studentmail, Val.Casey, Fergal.McCaffery, Gerry.Coleman}@dkit.ie

**Abstract.** Effective verification and validation are central to medical device software development and are essential for regulatory approval. Although guidance is available in multiple standards in the medical device software domain, it is difficult for the manufacturer to implement as there is no consolidated view of this information. Likewise, the standards and guidance documents do not consider process improvement initiatives. This paper assists in relation to both these aspects and introduces the development of processes for verification and validation in the medical device domain.

Keywords: Medical device standards, Medical device software verification and validation, Medical device software process assessment and improvement

## 1 Introduction

Verification and Validation (V&V) activities are important activities in the software development lifecycle and consume up to 50% of project development time [1], [2] and up to 50% of the total cost [3]. While both V&V play a key role in software development, there is a level of ambiguity in the use of both of these terms. This is evident from the difference in the definition of these terms in the literatures [4], [5] and [6].

When developing safety-critical software it is imperative to have software development practices which incorporate effective V&V activities. In this context V&V are addressed by numerous standards for both generic and safety-critical software development which include specific medical device standards.

The National Institute of Standards and Technology performed a study, indicating that software defects cost the U.S. economy in the region of \$59.5 billion a year [7]. The study also indicates that better testing could detect and

remove defects early in the development process and reduce the cost by more than a third [7]. However, there are challenges in the implementation of V&V in the context of general software development and these challenges are even greater in safety-critical domains. The requirements put forth by the regulatory bodies stress the need for supporting documentation and it can be challenging to satisfy these regulatory requirements and meet the pressures of the market at the same time.

## 2 Verification & Validation in Generic Software Development

Two important reference models which are widely used in the context of software process improvement are the Capability Maturity Model® Integrated (CMMI®) [8] and ISO/IEC 15504-5:2006 [9]. When considering software V&V we determined it was of value to consider both.

CMMI® recommends a lifecycle approach for V&V activities. It defines verification as “Confirmation that work products properly reflect the requirements specified for them”. In other words, verification ensures that “you built it right” and validation is “Confirmation that the product, as provided (or as it will be provided), will fulfill its intended use”. In this context, validation ensures that ‘you built the right thing’. The V&V processes are part of the engineering processes category and both are level 3 process areas in the staged model. The model also provides guidance in terms of examples of methods such as peer reviews; statement coverage testing and branch coverage testing that could be performed. The validation process area incrementally validates products against the customer’s needs. Validation may be performed in the operational environment or simulated operational environment. Coordination with the customer in relation to validation requirements is an important element of this process area. The scope of the validation process area includes validation of products, product components, selected intermediate work products, and processes. These validated elements may often require re-verification and re-validation.

In ISO/IEC 15504-5:2006 V&V are two distinct processes and are part of the supporting lifecycle process group. Both of these processes are based on the respective lifecycle processes in ISO/IEC 12207 AMD1 [10]. In ISO/IEC 15504-5, the purpose of the verification process is to confirm that each software work product and/or service of a process or project properly reflects the specified requirements. The tasks pertaining to verification include: development of a verification strategy, development of criteria for verification, performing the activity of verification, determination of actions based on verification results and making the results available to the stakeholders. Industry experience indicates that V&V activities typically consume about 30-50% of development budgets [11]. CMMI® and ISO/IEC15504-5 are not prescriptive when it comes to methods and tools to be used for V&V. Rather it

is left to the discretion of the user to select and apply methods. Though CMMI<sup>®</sup>, considers validation and verification, it is still rather modest in its focus on these areas compared to other elements of the development processes [12].

### 3 Verification & Validation for Safety-Critical Software Development

Software can be a critical element of complex, potentially dangerous products such as weapons systems, aerospace systems and medical devices. These are critical because failure can result in loss of life, significant environmental damage, and major financial loss [13]. It has also been found that there is a relationship between the increasing occurrence of system accidents and the increasing usage of software [14]. In these circumstances these products are required to meet a very high-level of reliability, security, and performance. Therefore, ensuring that such systems meet their predefined requirements and that they perform as expected is an essential and often challenging issue [15].

Within the safety-critical software arena, different standards/certifications are available for different industries. These include the MIL-STD-498 [16] for military applications, DO-178B [23] for Aerospace, and Automotive SPICE and ISO 26262 [17] in the Automotive industry. IEC 60880 [18] describes the European standards for the certification of nuclear power generating software. IEC 61508 [19] describes a general-purpose hierarchy of safety-critical development methodologies that has been applied to a variety of domains ranging from medical instrumentation to electronic switching of passenger railways. Though these standards address V&V in sufficient detail, their role is not to address process improvement. In addition, there are some [20] who consider that a CMMI<sup>®</sup> V&V assessment inadequate when dealing with safety-critical software, and they propose a new framework for V&V assessment, focused on safety-criticality. This framework is defined through integrating safety standards with the V&V process areas of the CMMI<sup>®</sup> and the ISO 9001 standard [21].

The following are some of the attributes of safety standards: (1) Product versus process (2) Safety management agents (3) Risk assessment (4) Integrity levels (5) Design safety and (6) Assurance techniques [22]. Based on these criteria, we decided to use DO-178B and the Automotive SPICE as part of our research for developing V&V processes for the medical device software domain. Automotive SPICE has been derived from ISO/IEC 15504-5. This was of particular relevance as it was developed for a safety-critical domain to facilitate process assessment and improvement. Therefore, existing software process reference models need to be adapted and extended to meet the specific requirements of medical device software development which is safety-critical in nature.

## 4 Research Approach and Outcomes

The research undertaken involved a number of key stages:

1. The V&V processes were reviewed in detail and consideration was given to how they were addressed by generic software development standards and process improvement reference models, which included ISO/IEC 12207, ISO/IEC 15504-5 and CMMI®.
2. A literature review of V&V was performed in terms of safety-critical software development. This included a review of the V&V processes addressed by safety-critical software development standards such as DO-178B and Automotive SPICE [24].
3. A literature review and analysis was also performed in relation to medical device software V&V. This incorporated the Food and Drug Administration (FDA) guidelines for Software Validation (FDA GPSV) [25], the Medical Device Directive (MDD) 1993/42/EEC [26] and amendment 2007/47/EC [27], ISO/IEC 62304 [28], ISO/IEC 13485 [29] and ISO/IEC 14971 [30].
4. Based on this analysis we defined a set of processes for V&V for medical device software development. The processes were assigned a Process ID, Process Name, Process Purpose, Process Outcomes and a set of Specific Practices. Guidance in the implementation of these specific practices is provided through a set of sub-practices and notes. These processes were developed as part of the Medi SPICE [41].

### 4.1 Regulatory Nature of the Medical Device Domain

Studies in the medical device industry [31], [32], point to the fact that software is one of the most critical factors for cutting edge products and the role software plays is predicted to continue to increase [32]. It is also expected that, by 2015, the research and development investment in software in this area will increase from 25% of the overall budget in 2002, to 33% [32].

However, as the role of software in the medical device domain increases, so do the number of failures which arise due to software defects. An analysis of medical device recalls by the FDA in 1996 [33] found that software was increasingly responsible for product recalls. This continues to be the case and in the period the 1<sup>st</sup> January 2010 to 1<sup>st</sup> January 2011 the FDA recorded 80 medical device recalls and state software as the cause [46]. A German survey on medical device recalls indicated that software was the top cause for risks related to construction and design defects of medical device products. This analysis, from June 2006, showed that 21% of the medical device design failures were caused by software defects. This was an increasing trend, as the figures from November 2005 showed software was responsible for 17% of construction and design defects [34].

Due to the safety-critical nature of medical devices, medical device companies who wish to sell their products must comply with the regulatory requirements of the respective countries where they plan to market them. Medical devices can only be sold in the US if they comply with the FDA regulations [35], whereby a quality system needs to be in place that complies with the FDA Regulations 21 CFR Part 820, Quality System Regulation (QSR) [36]. In order to sell devices the manufacturer not only has to prove safety and effectiveness, but also has to demonstrate that the design and development of the device including the software complies with the FDA regulations. The “Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices” document [37] details these requirements. Though the regulatory bodies, such as the FDA provide guidance documents, they do not dictate that a particular method must be used [38]. The quality system process itself is designed by the medical device manufactures and the quality system process needs to ensure that the manufacturer is designing and building a quality product. The difficult part is that the manufacturer has to provide evidence to the FDA inspectors that the correct processes have been followed [35].

In order to achieve standardization of expectations and for better guidance for implementation by manufacturers, the FDA Center for Devices and Radiological Health (CDRH) has published guidance documents which include risk-based activities to be performed during software validation [25], pre-market submission [37] and when using off-the-shelf software in a medical device [39]. Although the CDRH guidance documents provide information on which software activities should be performed, they do not enforce any specific method for performing these activities. The result is that the medical device manufacturers could fail to comply with the expected requirements.

Within the medical device industry a decision was initially made to recognize ISO/IEC 12207:1995 (a general software engineering lifecycle process standard) as being suitable for general medical device software development [10]. However, upon careful examination of ISO/IEC 12207, the Association for the Advancement of Medical Instrumentation (AAMI) software committee decided it was necessary to create a new standard specifically for medical device software development. The AAMI used ISO/IEC 12207:1995 as the foundation for their new standard “AAMI SW68, Medical device software – Software lifecycle processes” [38]. In 2006, a new standard AAMI/IEC 62304 [28] was released that was based on the AAMI SW68 standard. In order to sell medical devices within the Europe Union (EU) the CE mark is required. To achieve the CE mark compliance is required with the Medical Device Directive (MDD) (1993/42/EEC) and amendment MDD (2007/47/EC), In-Vitro Diagnostic Directive (IVDD) [44] and the Active Implantable Medical Device Directive (AIMDD) [45] depending on the type of medical device being submitted. As stated in the latest amendment to the MDD, Section 1 (g) of

MDD (2007/47/EC) "For devices which incorporate software or which are medical software in themselves, the software must be validated according to the state of the art taking into account the principles of development lifecycle, risk management, validation and verification". "State of the Art" is used here to mean what is generally accepted as good practice. Since this requirement was introduced, developers must now validate the software be it integrated or standalone, regardless of device class. IEC 62304 and its aligned standards are often seen as a good place to start when validating software. Whilst these standards are generally accepted and are harmonised under the MDD they do contain omissions which make them difficult to apply to standalone software as an active medical device. As we had observed from our research this is exemplified by IEC 62304 where there is no provision within the standard to validate the system elements of standalone software.

While there are numerous standards in the medical device domain they are oriented towards achieving regulatory compliance. As a result the focus of medical device software development is compliance rather than process improvement. To address this Medi SPICE is being developed. The objective of Medi SPICE is to provide a process assessment and improvement model which is domain specific to medical device software development and incorporates regulatory compliance. Medi SPICE will also enable the harmonization of different standards in the medical device software development domain, thus bringing best practices available in multiple standards into a single framework which will aid manufacturers in the implementation of their requirements as well as in their process improvement exercise. The results of a Medi SPICE assessment may be used to indicate the state of a medical device suppliers software practices in relation to the regulatory requirements of the industry, and identify areas for process improvement. The results of these assessments may also be used as a criterion for supplier selection. The authors believe that, with the publication of the Medi SPICE more specific guidance will be available for the basis of process design and assessment in the medical device industry [41].

#### 4.2 Verification & Validation in Medi SPICE

Based on our research which comprised of an extensive literature review and comparative analysis of standards in the medical device and other safety-critical domains, we arrived at the following findings, which were incorporated into the definition of the processes related to V&V in Medi SPICE:

- a. From the literature review and comparison across other standards and models, it became clear that the terms V&V are frequently used interchangeably. The FDA guidelines distinguish between verification and validation. Though the FDA is clear on the definition part, sections 4, 5 and 6 of the FDA GPSV, which deals with operational activities, still use the term validation only and no reference is made

to verification. Guidance on differentiating between V&V activities with respect to the different engineering activities/work products should be in place. The amendment 2007/47/EC to the MDD stresses the importance validation plays and the need for state of the art validation and verification

- b. Verification is not addressed as a separate process in IEC 62304 and verification practices are integrated into other engineering processes. Validation is considered a systems level process and outside the scope of IEC 62304 even when the system consists entirely of software.
- c. Automotive SPICE has V&V criteria and V&V records as outputs in its processes. The ISO/IEC 15504-5 does not go to this level of detail.
- d. The IEEE Standards for Software V&V state that classical Independent Verification & Validation (IV&V) is generally required for the development of software systems deemed “critical”, i.e., those which can result in loss of life, mission or significant social or financial loss [42]. Independence is an important factor addressed by DO-178B. The degree of independence is also addressed in ISO/IEC 15504-5 and Automotive SPICE. The FDA GPSV Sec 4.9 does address independence and leaves it to the discretion of device manufacturers as to how this is to be achieved. Independence is not addressed as part of IEC 62304 and it assumes that it is taken care of by/through ISO 13485. Therefore, in Medi SPICE we placed a particular focus on the clarity of independence in the verification and validation processes.
- e. Software developed for medical devices concerns itself with obtaining regulatory approval as opposed to improving processes to achieve more efficient software development [40]. Models like CMMI<sup>®</sup> and ISO/IEC 15504-5 have separate process areas for verification and validation. A separate process area for critical processes like V&V enable organizations to understand their strengths and weaknesses in a detailed manner and can provide help when embarking on process improvement initiatives.

From our analysis of the relevant literature regarding V&V and through the comparative analysis of process improvement models and standards, our goal was to determine best practice in this area and to facilitate process improvement. Our objective was also to satisfy the requirements of the relevant medical device standards which include the FDA GPSV, MDD, ISO/IEC 13485, IEC 62304, IEC TR 80002-1 [43] and ISO14971. Having established these elements, it was imperative we incorporate them into Medi SPICE. To achieve this we developed the following Medi SPICE processes with a particular emphasis on verification and validation:

1. Software Construction
2. Software Integration
3. Software Testing
4. Verification

## 5. Validation

As discussed in Section 4, our objective was to incorporate the relevant standards and the most effective elements of process improvement models into a common framework specifically designed for the medical device software domain, Medi SPICE.

Unlike ISO/IEC 15504-5, where there is no requirement for the classification of outcomes and processes based on safety, this was an important element which had to be included in Medi SPICE. We therefore utilized the classification schema provided by IEC 62304, which is used to associate the outcomes and specific practices with the safety level of the software for which the practices are applicable. While we based the practices on the ISO/IEC15504-5, our analysis of standards in similar safety-critical domains highlighted that it would be beneficial to use Automotive SPICE as our reference, as it has been developed to meet the specific requirements of safety-critical software development. As a result of the findings from our research, we have included V&V as separate process areas in Medi SPICE. The validation process includes many of the recommendations that were produced as part of this research.

Risk management is an integral part of medical device software development. In this context the relevant standards for medical device development are ISO 14971 and IEC 62304. IEC/TR 80002-1 provides specific guidance as to how these two standards can be combined to address risk with regard to medical device software development. The requirements of V&V as required by these standards have been addressed in the five processes listed above.

### 4.3 Summary of Verification & Validation related Processes in Medi SPICE

For the purpose of this paper, we use the Software Testing (ENG.8) process in Medi SPICE as an example. This process relates to the IEC 62304 Software System Testing activity which has five tasks. As an outcome of our analysis, specific practices (1 to 10) were defined for the Software Testing Process. The practices and how they map to relevant medical device standards are illustrated in the Table 1:



Table 1: Mapping ISO/IEC 62304 Tasks to Medi SPICE practices

| IEC 62304 Reference | Sub Task / Clause                         | Medi SPICE Reference | Medi SPICE Reference  |
|---------------------|---|----------------------|---|
| 5.7.1               | Establish tests for software requirements | ENG.8.SP1            | Develop software test strategy                                  |
|                     |   | ENG.8.SP1.1          | Establish a set of tests  |
|                     |   | ENG.8.SP2            | Develop test specification for software test                    |
|                     |   | ENG.8.SP4            | Test the integrated software                                    |
| 5.7.2               | Use software problem resolution process   | ENG.8.SP5            | Record the anomalies  |
| 5.7.3               | Retest after changes                      | ENG.8.SP9            | Develop regression test strategy and perform regression testing |
| 5.7.4               | Verify software system testing            | ENG.8.SP3            | Verify test specification for software test                     |
|                     |   | ENG.8.SP7            | Verify software testing   |
| 5.7.5               | Software system test record contents      | ENG.8.SP6            | Record results of software test                                 |
|                     |   | ENG.8.SP8            | Ensure consistency and bilateral traceability                   |
|                     |   | ENG.8.SP10           | Conduct risk management activities                              |

Against the five tasks that the IEC 62304 provides, the Medi SPICE Software Testing Process has nine specific practices and one sub practice. In line with the good practice of ensuring traceability at each engineering activity level as observed in ISO/IEC 15504-5, Medi SPICE also focuses on using traceability in each engineering activity as it is very important from a verification perspective. In addition, to the specific practices a single sub-practice, additional implementation guidance is provided through 10 notes in the Software Testing Process. It may be noted from Table 1 that a specific practice – Conduct risk control activities has been added as part of the model. We are thus providing guidance through Medi SPICE on risk management activities in line the ISO 14971, which requires verification of the implementation of risk control, as well as verification of the reduction of risk through adopting risk control mechanisms.

Table 2, outlines how we have addressed some of the typical software testing tasks, with reference to the FDA GPSV guidance document.

Table 2: Mapping FDA Typical tasks to Medi SPICE

| <b>FDA Typical Tasks</b>            | <b>Medi SPICE Reference</b>   |
|-------------------------------------|---|
| Test Planning                       | Software Construction   |
| Functional test case identification | Software Construction   |
| Traceability analysis               | Software Construction,<br>Software Integration,<br>Software Testing |
| Unit (module) test execution        | Software Construction   |
| Integration test execution          | Software Integration  |
| Functional test execution           | Software Integration  |
| System test execution               | Software Testing  |
| Error evaluation/resolution         | Software Testing  |
| Final Test Report                   | Software Testing  |

The requirements of FDA GPSV are directly addressed by Medi SPICE as can be observed from the mapping. Further, it needs to be noted that the task of Acceptance test execution is not addressed by Medi SPICE as part of software engineering processes. This is in line with the Automotive SPICE as well as ISO/IEC 15504-5, where acceptance testing is part of the acquisition group of processes.

## 5 Conclusions and Future Work

Further to the definition of a set of process areas and the associated practices related to V&V, the processes should be piloted in organizations within the medical device software development industry. Based on the results observed, the processes should be evaluated and continuously improved based upon feedback from the medical device software development industry. Additionally, as V&V absorbs a significant amount of project time, further research will be performed out on practices, which could bring in reduction in cycle time for V&V activities but without compromising quality and safety features of the products being developed.

Globalization of software development has led to distributed teams working on the same product in different locations. Understanding the challenges in globally distributed V&V in the context of medical device software development and what additional practices could aid practitioners in such

cases. These practices could then become notes or sub-practices in further versions of Medi SPICE.

As medical device manufacturers outsource their medical device software development, it would be worthwhile to examine: (a) what could be drivers in outsourcing V&V activities to a third party. (b) if outsourcing of medical device software development is performed, will V&V still be done internally? (c) What risks should be considered and practices should be included in a reference model for medical device software development from a V&V perspective for 3rd party software (COTS) or where certain activities are outsourced?

## 6 Acknowledgments

This research is supported by the Science Foundation Ireland (SFI) Stokes Lectureship Programme, grant number 07/SK/I1299, the SFI Principal Investigator Programme, grant number 08/IN.1/I2030 (the funding of this project was awarded by Science Foundation Ireland under a co-funding initiative by the Irish Government and European Regional Development Fund), and supported in part by Lero - the Irish Software Engineering Research Centre (<http://www.lero.ie>) grant 03/CE2/I303\_1.

## 7 References

1. E. Bernard, B. Legeard, X. Luck and F. Peureux, "Generation of test sequences from formal specifications: GSM 11-11 standard case study," *Software: Practice and Experience*, vol. 34, no. 10, 2004, pp. 915-948
2. S. Rayadurgam and M. P. E. Heimdahl, "Test-sequence generation from formal requirement models," *Proc. High Assurance Systems Engineering, 2001. Sixth IEEE International Symposium on*, 2001, pp. 23-31
3. G. Myers, C. Sandler, T. Badgett and T. Thomas, *The Art of Software Testing*, Second Edition, Wiley, 2004
4. Wallace, D.R.; Fujii, R.U.; , "Software verification and validation: an overview," *Software, IEEE* , vol.6, no.3, pp.10-17, May 1989
5. Berling, T.; Thelin, T.; , "An industrial case study of the verification and validation activities," *Software Metrics Symposium, 2003. Proceedings. Ninth International* , vol., no., pp. 226- 238, 3-5 Sept. 2003
6. Arthur, J.D.; Groner, M.K.; Hayhurst, K.J.; Holloway, C.M.; , "Evaluating the effectiveness of independent verification and validation," *Computer* , vol.32, no.10, pp.79-83, Oct 1999
7. Gregory Tassy, May 2002, *The Economic Impacts of Inadequate Infrastructure for Software Testing*, National Institute of Standards and Technology (NIST), report 02-3
8. CMMI Product Team (2006) *Capability Maturity Model® Integration for Development Version 1.2*. Software Engineering Institute
9. ISO/IEC 15504-5:2006 (2006) *Information technology — Process Assessment — Part 5: An Exemplar Process Assessment Model*. ISO, Geneva, Switzerland

10. ISO/IEC 12207:1995/Amd.1 (2002) Information Technology — Software life Cycle Processes Amendment 1. ISO, Geneva, Switzerland
11. E. Kit, 1995, Software Testing in the Real World, Addison-Wesley
12. page 1, Jacobs, J.C.; Trienekens, J.J.M.; , "Improving verification and validation in hardware/software environments. Introduction to the workshop," Software Technology and Engineering Practice, 2002. STEP 2002. Proceedings. 10th International Workshop on , vol., no., pp. 121- 122, 6-8 Oct. 2002
13. Why Software Requirements Traceability Remains a Challenge - Cross Talk Jul/Aug 2009 Issue, <http://www.stsc.hill.af.mil/crosstalk/2009/07/0907Kannenbergsaiedian.html>
14. Paper on System Safety in Computer-Controlled Automotive Systems - Nancy G. Leveson, Massachusetts Institute of Technology, <http://sunnyday.mit.edu/papers/sae.pdf>
15. page 1, Alawneh, L.; Debbabi, M.; Hassaine, F.; Jarraya, Y.; Soeanu, A.; , "A unified approach for verification and validation of systems and software engineering models," Engineering of Computer Based Systems, 2006. ECBS 2006. 13th Annual IEEE International Symposium and Workshop on , vol., no., pp.10 pp.-418, 27-30 March 2006
16. MIL-STD-498 (5th December 1994) Military Standard - Software Development And Documentation. US Department of Defense, USA
17. ISO/DIS 26262 Road vehicles - Functional safety. ISO, Geneva, Switzerland
18. IEC 60880:2006 (2006) Nuclear power plants - Instrumentation and control systems important to safety - Software aspects for computer-based systems performing category A functions. IEC, Geneva, Switzerland
19. IEC/TR 61508:2005 (2005) Functional safety of electrical/electronic/programmable electronic safety related systems. BSI, London
20. A. Y. Kyung, P. Seung-Hun, B. Doo-Hwan, C. Hoon-Seon and J. Jae-Cheon, "A Framework for the V&V Capability Assessment Focused on the Safety-Criticality," Proc. Software Technology and Engineering Practice, 2005. 13th IEEE International Workshop on, 2005, pp. 17-24
21. "ISO 9001:2000 - Quality management systems - Requirements," 2000
22. A. Eastaughffe, A. Cant, M.A. Ozols, "A Framework for Assessing Standards for Safety-critical Computer-Based Systems," ISESS, pp.33, Fourth IEEE International Symposium and Forum on Software Engineering Standards, 1999
23. RTCA, RTCA DO-178B, Software Considerations in Airborne Systems and Equipment Certification, 1992
24. Automotive SPICE Process Assessment (21 August 2005). SIG
25. US FDA Center for Devices and Radiological Health (2002) General Principles of Software Validation; Final Guidance for Industry and FDA Staff. CDRH, Rockville
26. European Council (1993) Council Directive 93/42/EEC Concerning Medical Devices. Official Journal of The European Communities, Luxembourg
27. European Council (2007) Council Directive 2007/47/EC (Amendment). Official Journal of The European Union, Luxembourg
28. ANSI/AAMI/IEC 62304:2006 (2006) Medical device software—Software life cycle processes. AAMI, Arlington
29. ISO 13485:2003 (2003) Medical devices — Quality management systems — Requirements for regulatory purposes. Second edn. ISO, Geneva, Switzerland

30. ISO 14971:2007 (2007) Medical Devices — Application of risk management to medical devices. Second edn. ISO, Geneva
31. Future Trends in Medical Device Innovation. Advanced Medical Technology Association (AdvaMed), 2004, <http://www.advamed.org/MemberPortal/About/NewsRoom/MediaKits/futuretrendsinmedicaldeviceinnovaton.htm>
32. ITEA 2 Blue Book. September 2005, [http://www.itea2.org/itea\\_2\\_blue\\_book](http://www.itea2.org/itea_2_blue_book)
33. Dolores R. Wallace and D. Richard Kuhn: Failure Modes in Medical Device Software: An analysis of 15 years of recall data. International Journal of Reliability, Quality and Safety
34. Feldmann, R.L.; Shull, F.; Denger, C.; Host, M.; Lindholm, C.; , "A Survey of Software Engineering Techniques in Medical Device Development," High Confidence Medical Devices, Software, and Systems and Medical Device Plug-and-Play Interoperability, 2007. HCMDSS-MDPnP. Joint Workshop on , vol., no., pp.46-54, 25-27 June 2007
35. Weiguo Lin; Xiaomin Fan; , "Software Development Practice for FDA-Compliant Medical Devices," Computational Sciences and Optimization, 2009. CSO 2009. International Joint Conference on , vol.2, no., pp.388-390, 24-26 April 2009
36. US FDA/CDRH, 21CFR820, Quality System Regulation, 2007
37. US FDA/CDRH, Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices, 2005
38. Burton J, Mc Caffery F, Richardson I A risk management capability model for use in medical device companies. In: International Workshop on Software quality (WoSQ '06), Shanghai, China, May 2006. ACM, pp 3 - 8
39. US FDA Center for Devices and Radiological Health (1999) Off-The-Shelf Software Use in Medical Devices; Guidance for Industry, medical device Reviewers and Compliance. CDRH, Rockville
40. Eagles, S. Murray, J. Medical Device Software Standards: Vision and Status, Medical Device and Diagnostic Industry, May 2001, <http://www.mddionline.com/archive/all/1969?page=475>
41. Medi SPICE: "An Update", Fergal Mc Caffery, Alec Dorling, Valentine Casey, the 10th International Spice Conference (SPICE 2010), 18th to 20th May 2010, Pisa, Italy
42. IEEE 1998. IEEE Standards for Software Verification and Validation, Page 58
43. IEC/TR 80002-1:2009 (2009) Medical device software Part 1: Guidance on the application of ISO 14971 to medical device software. BSI, London
44. European Council (1998) Council Directive 98/79/EC On in vitro diagnostic medical devices, Official Journal of The European Communities, Luxembourg
45. European Council (1990) Council Directive 90/385/EEC On the approximation of the laws of the Members States relating to active implantable medical devices, Official Journal of The European Communities, Luxembourg
46. Medical & Radiation Emitting Device Recalls (2011) FDA, <http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfres/res.cfm>. Accessed 10/02/2011