Software Process Improvement & Roadmapping – A Roadmap for Implementing IEC 62304 in Organizations Developing and Maintaining Medical Device Software

Peter Rust, Derek Flood, Fergal McCaffery

Regulated Software Research Centre & Lero,
Dundalk Institute of Technology, Dundalk, Ireland
{peter.rust, derek.flood, fergal.mccaffery}@dkit.ie

Abstract. Organizations engaged in medical device software are required to demonstrate compliance with a set of medical device standards and regulations before the device can be marketed. One such standard *IEC 62304, Medical device software -- Software life cycle processes*, is a standard that defines the processes that are required to be executed in order to develop safe software. Demonstrating compliance with IEC 62304 can be problematic for organizations that are new to or have limited experience in the domain. The standard defines what processes must be carried out, but does not state how. This paper presents a research method for generating a roadmap that will guide organizations in the implementation of IEC 62304.

Keywords: Medical device software, medical device standards, regulatory compliance, software roadmap, Software Process Improvement, Software Process Improvement Roadmaps, IEC 62304

1 Introduction

Developing safe medical device software is critical, especially considering the number of recalls of medical devices and the number of deaths and serious injuries caused by failure of software in medical devices [1][2]. Alemzadeh et al.[2] describe how 33.3% of Class I (presenting a high risk of severe injury or death to patients) recalls between 2006 and 2011 were software related.

Authorities around the world, charged with the regulation of medical devices, have recognized the importance of standards adoption in the development and manufacture of medical devices. ISO 13485 [3], ISO 14971 [4] and IEC 62366 [5] form a suite of standards introduced to help improve the development of safe medical devices, including software.

Software is now also deemed to be a medical device in its own right [6]. IEC 62304 [7] identifies the processes that need to be carried out but do not say how the processes should be carried out. The existing Software Process Improvement (SPI) models, such as the Capability Maturity Model® Integration (CMMI®) [8] and

ISO/IEC 15504-5:2012 (SPICE) [9] are directed to the general software development domain and do not provide sufficient coverage to achieve medical device regulatory compliance [10]. MDevSPICE® (formally known as Medi SPICE) has been developed to fill this gap [10]. MDevSpice® is based on ISO/IEC 15504-5:2012 [9], IEC 62304:2006 [11] and ISO/IEC 12207:2008 [12] and has being developed in line with the requirements of ISO/IEC 15504-2:2003 [13] and contains a Process Reference Model (PRM) and Process Assessment Model (PAM). However, these models only identify the gaps in an organizations processes but not how to fill them. The aim of this project is to develop a set of tailored "How To" SPI roadmaps for medical device companies to both improve their software development practices and assist them to achieve regulatory compliance. To meet this aim, this paper describes the creation of a roadmap for the implementation of IEC 62304.

The remainder of this paper is structured in the following manner: Section 2 outlines the related work carried out with regard to the use of roadmapping in general, in the SPI field and in the medical device standards domain. Section 3 discusses the importance of the software development lifecycle within the medical device domain. Section 4 describes the research method used in developing roadmaps while section 5 details the generation of the IEC 62304 roadmap. Section 6 discusses the experience of generating the roadmap. Section 7 outlines the future work before the paper is concluded in section 8.

2 Related Work

The roadmapping process is established and proven in the technology domain and continues to be adopted in many other fields of endeavour. Phaal [14] lists over 2000 public domain roadmaps organized by topic including chemistry, construction, defence, energy, transport and many more. A number of large companies use roadmapping to develop their strategic planning going forward. NASA embraced roadmapping in 2005[15] arising out of a number of cost overruns in their development budgets.

Within the SPI domain, the number of published roadmaps is limited. McFeeley et al.,[16] have developed a high level process improvement roadmap and describe how their roadmap is intended to provide an organization with a guide to forming and carrying out an SPI program.

Höss et al.,[17] launched a pilot project to acquire skills in implementing IEC 62304 in a hospital-based environment (in-house manufacture). They concluded that the pilot project carried out at their facility clearly demonstrated that the interpretation and implementation of IEC 62304 is not feasible without appropriately qualified staff. They recognized that it could be carried out by a small team with limited resources although the initial effort is significant and a learning curve must be overcome.

It can be seen that applying the roadmapping process to IEC 62304 and generating a roadmap that will aid medical device software development organizations in the implementation of IEC 62304 is a necessary and justified step.

Flood et al. [18][19] have already applied the roadmapping process to ISO 14971 and IEC 62366 and these roadmaps have been validated with industry experts. A roadmap has also been developed for traceability in the medical device domain leaving the development of an IEC 62304 roadmap as the last piece of the puzzle.

3 Software Development Lifecycle in the Medical Device Domain

Safe medical device software requires risk management, quality management and good software engineering [20]. IEC 62304 does not prescribe a specific lifecycle model, but rather the standard provides a framework of life cycle processes with activities and tasks that are necessary for the safe design and maintenance of medical device software. IEC 62304 is not a standalone standard and the manufacturer of a medical device is responsible for ensuring compliance with the other relevant standards. Irrespective of the lifecycle model chosen, the processes defined in the standard must form part of the model and be implemented during the development of the medical device software. One method organizations have of doing this is through mapping the standard to their particular life cycle model. The IEC 62304 implementation roadmap will remove this step in the software development process as the requirements of IEC 62304 are already mapped to the defined processes, identified as Activities and any gaps that exist in the organizations processes will be detected.

4 Research Method

The aim of the paper is to describe the roadmapping process undertaken to develop an SPI roadmap for IEC 62304. The method chosen has already been used successfully in developing roadmaps for ISO 13485, ISO 14971 and IEC 62366[18][21].

4.1 Overview

The definition of a Roadmap for the purposes of applying the roadmapping process to this and the other standards in the domain is "*A series of Milestones, comprised of Goals that will guide an organization through the use of specific Activities towards compliance with regulatory standards*"[18].

After evaluation of the IEC 62304 standard it was found that the existing terminology used in the roadmap definition was inappropriate. The use of milestone, goal and activity conflicted with their use in IEC 62304. Therefore the definition of a roadmap in this context has been redefined. The definition now reads "*A series of Activities, comprised of Tasks that will guide an organization through the use of specific "How To's" towards compliance with regulatory standards"*. All further references in this paper will use this new terminology.

4.2 Roadmap Development Method

To generate the roadmap for IEC 62304 the roadmap development method described by Flood et al [19] has been applied. This method, described below, has been revised in light of the changes to the definition of a roadmap.

1. Identify requirements of the standard and rephrase them as Tasks;
2. Group the Tasks into logical Activities;
3. Order the Activities into a sequence by which they can be introduced into an organization in a rational manner;
4. Validate the generated roadmap;
5. Identify the "How To's" that can meet the identified Tasks;
6. Validate the "How To's" in a host organization.

5 Roadmapping and Roadmaps

5.1 Roadmap Generation

In step 1 as described above the standard was decomposed into its elementary requirements and a total of 172 elementary requirements were identified. The requirements were then transformed into Tasks by the application of an action verb.

Taking as an example of the transformation process requirement 5.3.5 which states that "*the manufacturer shall identify the segregation between software items that is essential to risk control, and state how to ensure that the segregation is effective*". This was transformed into a Task defined as "***Identify the segregation between software items that is essential to risk control and state the measures taken that ensure the segregation is effective.***"

In step 2 when the transformation of all the requirements was complete, the Tasks were analysed for particular keywords that would aid their grouping into logical Activities. The above Task was assigned the keyword "Software Detailed Design". A total of five Tasks were grouped according to this keyword and an Activity created titled "Software Detailed Design". This process continued until all Tasks were grouped resulting in sixteen Activities. These are detailed in Table 1.

Table 1: Number of Tasks per Activity

Ref	Title	No of Tasks	Ref	Title	No of Tasks
1	Prerequisites.	2	9	Software Detailed Design Process	5
2	Software Development Planning Process	16	10	Software Unit Implementation and Verification Process	28
3	Software Documentation.	25	11	Software Integration and Integration Testing Process	7
4	Software Risk Management Process	13	12	Software System Testing Process	7
5	Software Requirements Analysis Process	16	13	Software Release Process	4
6	Software Architectural Design Process	6	14	Software Archive	2
7	Software Safety Classification.	4	15	Software Problem Resolution Process	18
8	Software Configuration Management Process	11	16	Software Maintenance Process	8

ISO/IEC TR 24774:2010 Systems and software engineering — Life cycle management — Guidelines for process description [22] recommends that the number of outcomes for a process should fall within the range 3 to 7. Considering this criteria and adapting it to arrive at the optimum range that should apply to the number of tasks in any given activity, the range between 1 and 7 inclusive was chosen. As can be seen from Table 1, some of the Activities have a number of Tasks far in excess of the optimum. The Tasks were re-analysed a further three times and a number of them reconstituted from their elemental parts. This resulted in 91 Tasks being attributed to the same sixteen Activities. This outcome is detailed in Table 2.

Table 2: Number of Tasks per Activity after Reconstitution of Tasks.

Ref	Title	No of Tasks	Ref	Title	No of Tasks
1	Prerequisites.	2	9	Software Detailed Design Process	5
2	Software Development Planning Process	3	10	Software Unit Implementation and Verification Process	9
3	Software Documentation.	13	11	Software Integration and Integration Testing Process	5
4	Software Risk Management Process	9	12	Software System Testing Process	2
5	Software Requirements Analysis Process	5	13	Software Release Process	3
6	Software Architectural Design Process	6	14	Software Archive.	2
7	Software Safety Classification.	3	15	Software Problem Resolution Process	11
8	Software Configuration Management Process	6	16	Software Maintenance Process	7

As can be seen from the table, the number of Tasks per activity was still problematic.

T-Plan the Fast Start to Technology Roadmapping [23] describes the approach for developing technology roadmaps. The approach consists of four structured and facilitated workshops that guide an organization through the process of developing a technology roadmap. Four workshops titled Market, Product, Technology and Charting are conducted where the relevant managers from the organization gather together to identify the needs of the market, a product that might satisfy that need, the technology required to build the product and finally to chart the way forward once a decision is made to follow the strategy developed. The charting workshop brings all the mangers together and a plan is drawn up as to how the strategy will be implemented. This is achieved by the use of a wall chart divided into layers and then a series of post-its are written up and pinned to the wall chart in the most appropriate layer. The managers can immediately visualize the plan as the workshop proceeds and the roadmap is produced by the end of the workshop. Due to the similarity of the two processes it was decided to try and utilize this workshop method to resolve the issues that arose with the generation of the IEC 62304 Roadmap using Step 2 of the original roadmap development method.

5.2 Roadmap Workshop

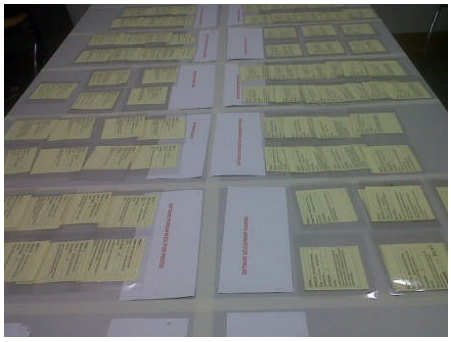
In preparation for the workshop each of the 91 Tasks were pre-printed on "post-its", as illustrated in Figure 1. The activity number and title were used as per the Activities identified during the initial generation. During the workshop these were not used and all Tasks were arranged as per the sections contained within the IEC 62304 standard. After each workshop these were updated to reflect the outcomes of the workshop.

To aid in the identification of individual Tasks, each one was assigned a unique Task Ref Number in the range of 1 to 91 and detailed on the post-its. In addition, to aid in traceability to the original standard, the IEC ref number and section title of each Task was recorded on the post-its.

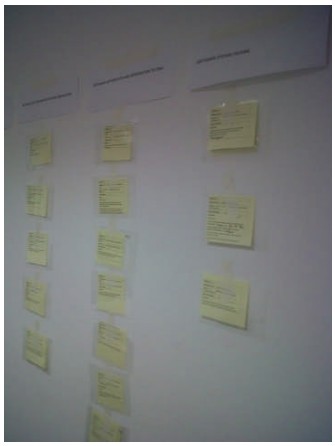
Activity: 13
Activity Title: Software Release
IEC Ref 5.8.5 **IEC Title**: Document how released software was created.
Task Ref Number 62
Task Text: Document the procedure and environment used to create the released software.

Figure 1 Example of Printed Post-it

Each activity was assigned a title and then each Task was assigned to an activity. The IEC 62304 reference and title from the standard were recorded on the post-it along with the Task text as detailed in figure 1. A room was laid out with a table on which the post-its were arranged as per the sections of IEC 62304 (see Photograph 1). A wall was designated on which the post-it's would be pinned in their final designated Activities (see Photograph 2). A number of three hour workshops were then conducted where a facilitator and three experts gathered to go through each Task and determine to which activity they belonged.



Photograph 1: Table laid out with arranged post-its



Photograph 2: Post-its pinned to wall under Activities

The facilitator introduced the aim of the workshop and gave a broad overview of the roadmapping process and what the output – the roadmap – might look like. A detailed discussion on each Task and which activity it belonged to took place and when agreement was arrived at, the new Task was pinned to the wall under the appropriate activity. This process continued until all the individual post-it's were allocated to Activities.

Due to the extent of the standard, three such workshops were used to finally determine the grouping of the Tasks to their Activities. The number of Tasks now totals 82, quite a number were combined on the basis of one of the underpinning ideas behind the standard – "if a process is undertaken then document it." To give an example, Tasks 7.1.2 and 7.1.4 were combined as "Identify and document in the risk management file potential causes of the software item identified in the medical device risk analysis activity (of ISO 14971) contributing to a hazardous situation."

5.3 Ordering the Activities

Step 3 of the method requires that the Activities be ordered in a manner by which they can be introduced to a medical device software development organization. Table 3 details the Roadmap that was developed during the course of the workshops and compares it to the one prior to the workshops. The Tasks associated with the Activities of Software Documentation and Software Archive were redistributed to other Activities as an outcome of the workshops. The consensus of the experts at the workshop was that as documentation plays a crucial role in the demonstration of compliance that the Tasks associated with documentation should be integrated into the performance of the Task rather than keeping them as a separate Task. In addition the

experts concluded that it would be more beneficial to merge the Tasks of Software Archive with Software Release to optimise the implementation of the roadmap.

The Change Request Process was added as an Activity and covers Tasks from section 5, 6, 7 and 8 of the IEC 62304 standard and now includes seven Tasks. These Tasks came from a range of other Activities, including Software Maintenance, Software Risk Management and Software System Testing. After a lengthy discussion the experts agreed that the Change Request Tasks would be implemented together rather than in their respective original Activities and therefore should be implemented as an Activity in their own right.

The Software Risk Management Process with nine, Software Architectural Design Process with ten and Software Problem Resolution Process with eight Tasks remain with their total number of Tasks above the optimum. However this is unavoidable due to the complex and rigorous nature of these Activities.

Table 3: Final Order of the Activities and the Number of Tasks

Ref	**Activity**	**No of Tasks prior to Workshops**	**No of Tasks after Workshops**
	Software Documentation	13	redistributed
	Software Archive	2	redistributed
1	QMS	1	1
2	RMS	1	1
3	Software Safety Classification	3	3
4	Software Development Planning Process	3	5
5	Software Configuration Management Process	6	4
6	Software Risk Management Process	9	9
7	Software Requirements Analysis Process	5	4
8	Software Architectural Design Process	6	10
9	Software Detailed Design Process	5	4
10	Software Unit Implementation and Verification Process	9	5
11	Software Integration and Integration Testing Process	5	6
12	Software System Testing Process	2	3
13	Software Release Process	3	6
14	Software Problem Resolution Process	11	8
15	Change Request Process	n/a	7
16	Software Maintenance Process	7	6

During the final workshop a discussion was held on the ordering of the Activities with particular reference as to how the roadmap might be graphically represented. Concern was expressed that the tabular representation with the Activities numerically identified might give an impression that one process must be complete before the next process can be undertaken. In consideration of this and with regard to the form of roadmaps that are generated in the technology domain a metaphor for the roadmap was generated and is detailed in figure 2.

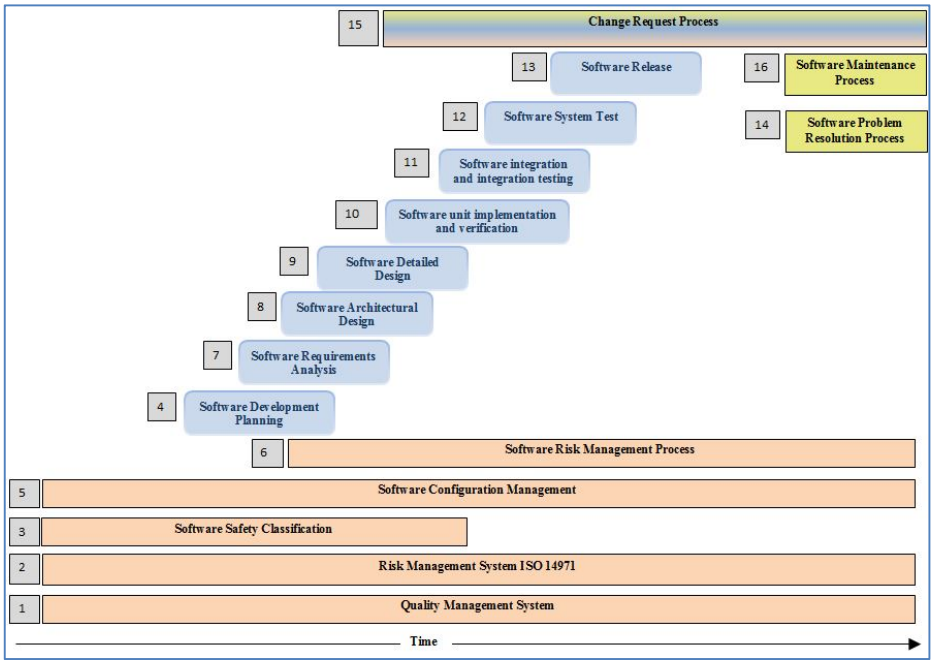


Figure 2: Metaphor for the Roadmap

The metaphor presented above was designed to highlight the stage at which each of the Activities may be applied during the development of a medical device software project. It can be seen that a number of the processes above may be ongoing for the duration of the software development process.

During the initial phase of the development of the product, a software safety classification of C is assigned to the device. During the architectural phase this may be revised in light of the risks posed by various components of the system therefore the software safety classification is ongoing right through the software architectural design phase.

Each of the phases in the software development lifecycle is depicted to overlap as a number of Tasks may be performed in parallel. Taking an example of the Software Unit Implementation and Verification Process and the Software Detailed Design Pro-

cess, it is feasible that during the second Task of the Software Detailed Design Process – "*Document a design with enough detail to allow correct implementation of each software unit*", the organization may commence the first Task of the Software Unit Implementation – "*Implement each software unit*".

6 Discussion

One of the reasons the method described in previous works [18][21] used in developing SPI roadmaps for ISO 13485, ISO 14971 and IEC 62366 achieved a successful outcome was due to the limited size and extent of the standards. IEC 62304 covers a much broader set of processes and the scalability of the method was not there when applied to IEC 62304. Three other methodologies were identified, "STAR "[24], Qupar [25] and "T-Plan the fast start to Technology Roadmapping" [26]. The work of Phaal et al. was of the greatest interest as it has gained a lot of traction in the technology roadmapping domain. A method for developing SPI roadmaps for the implementation of the regulatory standards which includes a workshop element can only enhance the roadmapping process. Having the opinion of experts in the medical device software development domain during the generation stage of the roadmap and in particular the discussion that was held on the ordering of the Activities was invaluable and consideration will be given to modifying the method to take into account the value of these types of workshop.

IEC 62304 defines the processes required for the development of safe software for the medical device domain but does not tell the organization "how to" carry out the processes. The generated roadmap when completed will fill this gap.

7 Future Work

The next stage of this work is to validate the roadmap through expert review. A number of experts will be recruited for the validation from a diverse range of backgrounds including those who work in the medical device domain and use the standards on a regular basis, assessors who regulate organizations using the standard, academics with the appropriate expertise, and members of the standards committee.

Once the roadmap is validated, work will commence on the identification of the "how to's" for the achievement of the Tasks defined in the generated roadmap and the building of a repository to house them. This will be achieved through interaction with organizations that are close to regulatory compliance and assessment of their processes. This will enable future implementations in medical device organizations.

The roadmap will be evaluated within medical device organizations of varying maturity. For each organisation the roadmap will be customized to suit their own circumstances including criteria such as the lifecycle that is being employed, the size of the organisation and their existing process. This will enable the method to be truly tested and validated in a real world setting.

8 Conclusions

Organizations that are engaged in or wish to become engaged in the medical device software development domain are placed under a high level of scrutiny by the regulatory bodies tasked with ensuring that the medical device organization is compliant with all the standards. These standards identify the requirements the medical device organization must satisfy without telling them how to achieve compliance. This can hinder both the development of new medical devices and existing software houses entering the medical device domain due to the range of methods available for implementing the standards.

Building on previous work in the area, which developed a set of SPI roadmaps for ISO 14971 and IEC 62366, this paper has introduced a roadmap for the implementation of IEC 62304. To develop this roadmap a number of workshops were conducted with experts in IEC 62304 which examined not only the arrangement of Tasks into Activities, but also examined the order in which these Activities should be introduced into an organization. Through this roadmap, organizations that are entering the medical device domain will be guided through the process of implementing the IEC 62304 standard in an efficient and effective manner.

Acknowledgement

This research is supported by the Science Foundation Ireland Principal Investigator Programme, grant number 08/IN.1/I2030 and by Lero - the Irish Software Research Centre (http://www.lero.ie) grant 10/CE/I1855 & 13/RC/20194.

REFERENCES

[1] J. R. Ward and P. J. Clarkson, "An analysis of medical device-related errors: prevalence and possible solutions.," *J. Med. Eng. Technol.*, vol. 28, no. 1, pp. 2–21, Jan. 2004.

[2] H. Alemzadeh, R. K. Iyer, Z. Kalbarczyk, and J. Raman, "Analysis of Safety-Critical Computer Failures in Medical Devices," *IEEE Secur. Priv.*, vol. 11, no. 4, pp. 14–26, Jul. 2013.

[3] "ISO 13485:2003 - Medical devices -- Quality management systems -- Requirements for regulatory purposes," *Geneva, Switzerland, ISO*. 2003.

[4] "ISO 14971:2007 - Medical devices -- Application of risk management to medical devices," *Geneva, Switzerland, ISO*. 2007.

[5] "IEC 62366-1:2007 - Medical devices -- Part 1: Application of usability engineering to medical devices," *Geneva, Switzerland, ISO*. 2007.

[6] M. McHugh, F. McCaffery, and V. Casey, "Standalone software as an active medical device," in *Communications in Computer and Information Science*, 2011, vol. 155 CCIS, no. 2011, pp. 97–107.

[7] "IEC 62304:2006 - Medical device software -- Software life cycle processes," *Geneva, Switzerland, ISO*. 2006.

[8] C. P. Team, "CMMI for Development, Version 1.3," *Carnegie Mellon Univ.*, no. November, 2010.

[9] "ISO/IEC 15504-5, Information technology - Process Assessment - Part 5: An Exemplar Process Assessment Model.," *Geneva, Switzerland, ISO*. 2012.

[10] F. McCaffery and A. Dorling, "Medi SPICE Development," *Softw. Process Maint. Evol. Improv. Pract. J.*, vol. 22, no. 4, pp. 255 – 268, 2010.

[11] "IEC 62304:2006 (2006) Medical device software—Software life cycle processes. Geneva, Switzerland, IEC.," vol. 3. 2006.

[12] "ISO/IEC 12207:2008, Systems and software engineering — Software life cycle processes." Geneva, Switzerland, ISO, 2008.

[13] "ISO/IEC 15504-2:2003, Software engineering — Process assessment — Part 2: Performing an assessment." Geneva, Switzerland, ISO, 2003.

[14] R. Phaal, "Public Domain Roadmaps." Centre for Technology Management University of Cambridge, 2011.

[15] L. Hall, "Space Technology Roadmaps: The Future Brought To You By NASA." Brian Dunbar, 07-Jun-2013.

[16] R. McFeeley, D. McKeehan, and T. Temple, *Software Process Improvement Roadmap*. Software Engineering Institute, 1995.

[17] A. Höss, C. Lampe, R. Panse, B. Ackermann, J. Naumann, and O. Jäkel, "First experiences with the implementation of the European standard EN 62304 on medical device software for the quality assurance of a radiotherapy unit.," *Radiat. Oncol.*, vol. 9, p. 79, 2014.

[18] D. Flood, F. McCaffery, V. Casey, and G. Regan, “A Methodology for Software Process Improvement Roadmaps for Regulated Domains - Example with IEC 62366,” *Commun. Comput. Inf. Sci.*, vol. 364 CCIS, pp. 25–35, 2013.

[19] D. Flood, F. Mc Caffery, V. Casey, R. McKeever, and R. Peter, “A Roadmap to ISO 14971 Implementation,” *J. Softw. Process Evol. Accept. Awaiting Publ.*

[20] “62304 and TIR32 Training Slides - Eagles_Dundalk_04Sept13.pdf.” [Online]. Available: https://www.dkit.ie/system/files/Eagles_Dundalk_04Sept13.pdf. [Accessed: 12-Mar-2015].

[21] D. Flood, F. McCaffery, V. Casey, and G. Regan, “A critical evaluation of a methodology for the generation of software process improvement roadmaps,” *Syst. Softw. Serv. Process Improv.*, vol. 425, pp. 36–47, 2014.

[22] “ISO/IEC TR 24774:2010 - Systems and software engineering -- Life cycle management -- Guidelines for process description,” *Geneva, Switzerland, ISO*. 2010.

[23] C. J. P. Farrukh, R. Phaal, and D. Probert, “Technology roadmapping: linking technology resources into business planning,” *Int. J. Technol. Manag.*, vol. 26, p. 2, 2003.

[24] A. H. N N Z Gindy, Maged S Morcos, B ulent Cerit, “Strategic Technology Alignment Roadmapping STAR® Aligning R&D Investments with Business Needs - document,” *International Journal of Computer Integrated Manufacturing, Taylor & Francis: STM, Behavioural Science and Pub- lic Health Titles*, 2008. .

[25] B. Regnell, R. B. Svensson, and T. Olsson, “Supporting Roadmapping of Quality Requirements,” *IEEE Softw.*, vol. 25, no. 2, pp. 42–47, Mar. 2008.

[26] R. Phaal, C. J. P. Farrukh, and D. R. Probert, *T-plan: The Fast Start to Technology Roadmapping. Planning Your Route to Success*. University of Cambridge. Institute for Manufacturing, 2001.