Management Gaps of Health Care Software Used in Devices

Karen Flores

University of San Diego

Management Gaps of Health Care Software Used in Devices

The invention and introduction of the microprocessor has incremented exponentially the computing power of computers, allowing them to be incorporated to medical use. The inclusion of these devices into health care had proved to be paramount for diagnostic and monitoring of patients. Microprocessors has helped to evolve almost every aspect of health care delivery through the acquisition, transformation, streaming and storage of data (Hewitt, Dolezel, & McLeod, 2017) (Giacomozzi & Martelli, 2016) (Ronquillo & Zuckerman, 2017). Implementations and uses of health care devices has evolved to the point that there is no need of human operation or supervision (Santos, 2019). In hand with computing power and device implementation in health care, there has been an evolution on the software that control these devices. Initially, this software was created as a set of written instructions. Now, software applications involve complex implementations that perform different functions and communicate with other devices.

The software in a health care device is the main element that determines the device functionality. Therefore, this software must be subject to a series of tests to guarantee that is free of design, implementation, business rules, and/or security flaws (MITRE, 2018). In addition, device and software vendors are required to disseminate and update constantly software for IoT devices in order to keep up with the demand for new functionality and the incorporation on new technology (Giacomozzi & Martelli, 2016). Therefore, vendors in charge of device software have the acquired responsibility of managing and responding adequatley to safety alerts and exposed vulnerabilities (Ronquillo & Zuckerman, 2017).

As a result of the increased use of health care device, the proper management of the device software has become one of the main areas of concern since 2017 for ECRI (2017; 2019). Proper software management is an important way to leverage the risks associated with incorrect functionality or security; and, it requires not only the participation of software vendors, but the vigilance from other stakeholders and organizations. The proposal is to have a set of metrics that will allow manufacturers and software developers to assess the reliability and availability of the software for a device. A second set of metrics will be added in order to allow the assessment of the effectiveness of current certification standards and regulations.

Literature Review

Management of software development cycle is a complex task that requires the application of different methodologies to guarantee that all functional requirements are accomplished. In addition to this, software is a critical component for devices that are implemented in the health care organizations (Ronquillo & Zuckerman, 2017). Therefore, this type of software should reliable during its use. This literature review include several research articles that are related to this issue. The articles were searched on Pubmed database using the following keywords: *software, health, care, device, quality, assurance, security*. The criteria to include the articles was the following: articles dated on 2014 or later; articles related to health care devices; articles related to software development.

The incorporation of devices and their software into the health care settings, nowadays, is a common practice. Medical device designs have increased their functionality and the complexity of their controls, as a result, their implementation is more complex (Furniss, Masci, Curzon, Mayer, & Blandford, 2015). Furthermore, the implementation of technologies, such as the cloud, has created more fragmented implementations (Furniss et al., 2015; Saxon, 2016). The development of content, diagnostic tools and treatments using mobile platforms has transformed the way patients are engaged with their treatments (Saxon, 2016). The continuous adoption of medical devices has helped to optimize resource and improve quality of care (Giacomozzi, C; Martelli, 2016).

Although, the introduction of medical devices has benefited health care provision there has been some risks associated with it. During the introduction of the health care devices into health care settings, the first problems to arise are related to hardware, yet, as hardware becomes more reliable, software problems are the ones that begin to surface (Furniss et al., 2015). These failures can result in unintended consequences that are hard to detect, challenging to monitor and harmful to the patient (Ronquillo & Zuckerman, 2017). Furthermore, these failures can also be attributed to the addition of use cases for these devices (Furniss et al., 2015). Even more, the integration of devices with communication capabilities with internet and other devices has introduced cybersecurity issues to the management of devices (Ronquillo & Zuckerman, 2017).

On the other hand, the regulatory landscape for health care devices is inconsistent and remain highly controversial (Ronquillo & Zuckerman, 2017). Furthermore, the health care industry has not yet adopted standards for mobile devices (Hewitt, Dolezel, & McLeod, 2017). In a review by Ronquillo & Zuckerman (2017), they detected that between 2011 and 2015 most medical devices were recalled due to software defects that would likely cause serious harm to patients. Even with Food and Drug Administration regulatory measures (§807.81 CFR) (§820 CFR), Sorenson & Drummond (2014) point out that there are many high risk devices that are not fully evaluated and a great majority has not gone through a premarket assessment. Therefore, it becomes necessary to establish a common path of certification that allows manufacturers, end-users and organizations to evaluate whether a health care device is safe to be integrated into a health care setting.

As pointed out by Sorenson & Drummond (2014), health care devices are not the same as drugs, therefore, they should not be submitted to the same regulatory standards. It is necessary to improve device regulation by enhancing the existing regulatory framework, strengthening premarket evidence standards and requirements, and , improving post-market monitoring (Sorenson & Drummond, 2014). All of these improvements can be implemented through the design of a device management framework that will allow to leverage risks and complex systems (Furniss et al., 2015; Rasmussen & Svedung, 2000). These framework can take in account different aspects of device vigilance, such as the incorporation of certification modes based existing certifications (Neto, Figueiredo Damásio, Monthaler, & Morais, 2015), requirement of sufficient evidence for use cases, a unique device identifier system (Sorenson & Drummond, 2014), and software analysis (Dey, Ashour, Shi, Fong, & Tavares, 2018; Höss et al., 2014; Neto et al., 2015).

Identified Solution

One of the challenges to leverage the risks of using medical devices during health care provision is the development of the appropriate framework that will involve all stakeholders and processes performed during the development and use of software intended for medical devices. Furniss et al. (2015) discuss the complexity of this problem and proposes a framework that takes in consideration all of the parts involved in the software development cycle of the device and its group of users. This idea is also supported by Giacomozzi, C & Martelli (2016) and Neto, Figueiredo Damásio, Monthaler, & Morais (2015), whom also proposed different frameworks for devices certification.

Since the landscape on which medical devices is complex, the risks associated with management gaps have to include a comprehensive cycle that includes various levels of risk management. Rasmussen & Svedung (2000) on their seminal work, propose the idea of having systems within systems and multiple levels of analysis in order to perform a sociotechnical analysis for identifying risks. Rasmussen & Svedung (2000) propose six levels of risk during the analysis of risks and safety in health care. For this specific scenario, five levels of risk will be proposed to build a framework in which different stakeholders will take action. Figure 1 presents the proposed structure for the flow of information during the management of software for medical devices. This framework includes 5 levels: government; regulators and associations; company; management; and, work.

On the *government* *level*, it will be responsibility of the government to create regulations, policies and procedures for premarket preview of software devices. Government should take in account that, while devices are different from drugs, they must be reviewed requiring sufficient evidence (Sorenson & Drummond, 2014). This will avoid that potentially high risk devices to reach the market without any review process or studies demonstrating their efficacy. Furthermore, since the functionality during the commercial use and controls of medical devices are constantly becoming more complex, the government should also enforce continuous post-market vigilance (Sorenson & Drummond, 2014).

On the *regulator and associations level*, there are a few actions that this group of stakeholders can take over. First, this group should take ownership of a division for registry of medical devices (i.e.: Unique Device Registry)(Sorenson & Drummond, 2014). This group will have to take ownership of developing and creating industry standards for medical devices on which medical manufacturers can rely on. Moreover, this group should determine a certification model based on existing certification standards or alternate means of certification (argument based approach or safety quality model) (Neto et al., 2015). Finally, as recommended by Rasmussen & Svedung (2000), this group should take on the report of incidents to government level.

The *company level* should be in charge of applying best practices that align and supports processes that comply with certification requirements. In the case of using and argument based approach for certification, the company should provide safety cases for certification authorities to assess them (Neto et al., 2015). For the safety quality model certification approach, the company should provide quality questions in order to guide assessors to decide whether a software product is safe or not (Neto et al., 2015). The *company level* should rely on the documental evidence provided by the *management level*. Moreover, *company level* should be accountable of incidents reported to *regulators and associations level*.

*Management level* should provide documental evidence for the software use cases taking in account hardware, software, user interfaces, computer integrating groups within the work setting, within the different layers of interaction (Furniss et al., 2015). This level will assist to acknowledge the increase in use cases and interactions of the devices. As a result, this level should be able to identify whether a medical device has a more complex functionality or interaction. This will also help to identify risks associated risk of design and use (Furniss et al., 2015). Furthermore, this level should keep track of changes on functionality and risk with a documental log.

Finally, *work level* should take ownership of all the stages that are related to the software development cycle. This level should be able to provide documentation of the implemented code and provide track of code changes through versioning methods (Frydman, Ruiz, Heymann, César, & Miller, 2014; Höss et al., 2014) and digital signature. Moreover, this level should be able to assess architecture for procedures that detects tampering (Dey et al., 2018) and identify security threats by design or default (Frydman et al., 2014). This risk analysis should be followed by a risk mitigation process that align with certification process and company best practices (Höss et al., 2014).

The selected approach aims to leverage the increasing landscape on which medical devices are interacting. This approach aims to be a comprehensive approach that targets different levels that are in charge of the proper function of medical devices. Most approaches reviewed leverage at some level the problem of managing software for medical devices and, at times, do not take in account lower or upper levels. Therefore, these approaches were not comprehensive and represented a partial solution for the problem. This approach represents the merging of different approaches using as a base the Proactive Risk Management Framework proposed by Rasmussen & Svedung (2000) and intends to provide a framework that can adapted to the increasing complexity of medical devices use.

Failure Mode Effect Analysis

The failure mode analysis was performed, first, by reviewing current literature. After literature review, there was a review of the process that would be necessary for the management of the software for medical devices. Figure 2. The image is the graphical representation of the series of proposed steps that can be followed during the process of IoT medical software vigilance process. The process is divided by its main actors and their tasks. The process was reviewed and each step of the process was grouped by stakeholder and by chronological order. Later, a risk analysis was performed for each step. Each of the risks were identified and introduced into the Table 1. Failure Mode Analysis for IoT Medical Device Software Vigilance Flow. On the other hand, Table 2. Failure Mode and Effect Analysis for IoT Medical Device Software Vigilance Flow, identifies the actions that will be needed to leverage the risks.

Quality Measure

Table 3. Quality Measures for Monitoring Software for IoT Devices contains the recommended quality measures needed to monitor software for IoT devices for health care applications. Measurement are presented in relation to the IoT Medical Devices Software Vigilance Workflow presented in previous sections. Measurements presented for steps 1 through 3 have the main goal to help manufacturers and software developers to identify the *Mean Time Between Failures* and *Mean Time to Recovery/Repair*. These two metrics are useful while assessing reliability and availability of a device and its software (Sandler, Ohstrom, Moy, & McVay, 2010; Siemens, 2011).

Measurements for Step 1 will be recorded and taken by the development team of the software application. Data must be delivered every team a team intends to release code for use in production. Management team will be in charge of collecting these data and reviewing it. Step 2 metrics can be generated from the metrics handed by the data team on Step 1. These metrics are intended to be summarized view of the metrics taken by the development team. Therefore, they must be delivered every team there is a plan to release a new version of the code to production.

For step 3 metric *Non-compliant certification reasons*, the main purpose of this step is the review of best practices for code development and the evaluation of risks for the code before it is released to production. The data for this metric can be generated as a checklist of items that meets the certification requirements. *Endpoint incidents* and *Mean Time to Repair* metrics will be used as a way to monitor the reliability and availability of the overall product. As a result, these two metrics must be taken as a measurement of time from the discovery to the time of code release and will be reported every 6 months.

Measurements related to steps 4 and 5 are proposed as a form of monitoring and correlating failures that can be present in one or more manufacturers and have an impact for the whole industry. The main intention of these measurements is to assess the effectiveness of current certification standards and regulations. These measurements will be a collection of events that took occurrence over one calendar year and will be reported to the upper level and public in general.

Conclusion

For some years, medical devices have been increasingly taking a central role in the provision of health care (Giacomozzi, C; Martelli, 2016; Sorenson & Drummond, 2014). The use case scenarios, design, functionality and controls have become increasingly complex within the clinical setting and, as a result, the systems that use these devices have become more complex and fragmented (Furniss et al., 2015). Initial problems detected on medical devices are related to hardware malfunction and, later, to their software (Furniss et al., 2015). These software problem can result in unintended consequences that are challenging to detect and monitor (Giacomozzi, C; Martelli, 2016). In the case of United States, the FDA is the organism in charge of reviewing the marketing of medical devices, yet, there is no consistent regulatory landscape for devices on commercial use (Ronquillo & Zuckerman, 2017)(§807.81 CFR; §820 CFR).

One of the challenges to leverage the risks of using medical devices during health care provision is the development of the appropriate framework that will involve all stakeholders and processes performed during the development and use of software intended for medical devices. Furniss et al. (2015) discuss the complexity of this problem and proposes a framework that takes in consideration all of the parts involved in the software development cycle of the device and its group of users. This idea is also supported by Giacomozzi, C & Martelli (2016) and Neto, Figueiredo Damásio, Monthaler, & Morais (2015), whom also proposed different frameworks for devices certification.

The paper proposes a process that includes most of the stakeholders involved in the design, development, implementation, and review of health care devices. This process can act as a framework that is divided into 5 steps that holds the tasks necessary to assess the risks of releasing a new device and/or use case to the market. The process intention is to become a self-evaluating framework that can serve as a tool for evaluating risks and measuring the availability and reliability of software vendors. This process also provides a framework for the assessment of the timeliness of current regulations and certifications in regards to the introduction to new technologies and use cases. Nevertheless, this process is limited by the communication and compliance of all stakeholders, therefore, in order to be effective all stakeholders must be committed to it in order to work.

# References

§807.81 CFR. (n.d.). *Subpart E--Premarket Notification Procedures.* Retrieved from Electronic Code of Federal Regulations: https://www.ecfr.gov/cgi-bin/text-idx?SID=6ff11aad52df1bd1f4dc3cc5fcc97fc9&mc=true&node=sp21.8.807.e&rgn=div6

§820 CFR. (n.d.). *QUALITY SYSTEM REGULATION.* Retrieved from Electronic Code of Federal Regulations: https://www.ecfr.gov/cgi-bin/text-idx?SID=6ff11aad52df1bd1f4dc3cc5fcc97fc9&mc=true&node=pt21.8.820&rgn=div5

Dey, N., Ashour, A. S., Shi, F., Fong, S. J., & Tavares, J. M. R. S. (2018). Medical cyber-physical systems: A survey. *Journal of Medical Systems*, *42*(4), 1–13. https://doi.org/10.1007/s10916-018-0921-x

ECRI. (2017). Top 10 Health Technology Hazards for. *Health Devices*, (November 2016).

Frydman, M., Ruiz, G., Heymann, E., César, E., & Miller, B. P. (2014). Automating risk analysis of software design models. *Scientific World Journal*, *2014*. https://doi.org/10.1155/2014/805856

Furniss, D., Masci, P., Curzon, P., Mayer, A., & Blandford, A. (2015). Exploring medical device design and use through layers of Distributed Cognition: How a glucometer is coupled with its context. *Journal of Biomedical Informatics*, *53*, 330–341. https://doi.org/10.1016/j.jbi.2014.12.006

Giacomozzi, C; Martelli, F. (2016). Health software: a new CEI Guide for software management in medical environment. *Annali Dell’Istituto Superiore Di Sanita*, *52*(4), 524–529. https://doi.org/10.4415/ANN\_16\_04\_11

Hazards, H. T. (2019). 2019 Top Ten Health Technology Hazards (10/2019).

Hewitt, B., Dolezel, D., & McLeod, A. (2017). Mobile Device Security: Perspectives of Future Healthcare Workers. *Perspectives in Health Information Management*, *14*(Winter). Retrieved from https://www.scopus.com/inward/record.uri?eid=2-s2.0-85045048263&partnerID=40&md5=974ec6b5e4790bad1d85e9992d1f5275

Höss, A., Lampe, C., Panse, R., Ackermann, B., Naumann, J., & Jäkel, O. (2014). First experiences EN 62304. *Radiation Oncology (London, England)*, *9*, 79. https://doi.org/10.1186/1748-717X-9-79

MITRE. (2018, March 30). *About CWE*. Retrieved from CWE: https://cwe.mitre.org/about/index.html

Neto, J. A., Figueiredo Damásio, J., Monthaler, P., & Morais, M. (2015). Product-based Safety Certification for Medical Devices Embedded Software. *Studies in Health Technology and Informatics*, *216*, 227–231. https://doi.org/10.3233/978-1-61499-564-7-227

Rasmussen, J., & Svedung, I. (2000). *Proactive Risk Management in a Dynamic Society*. *Karlstad: Swedish Rescue Services …*. https://doi.org/10.16250/j.32.1374.2016270

Ronquillo, J. G., & Zuckerman, D. M. (2017). Software-Related Recalls of Health Information Technology and Other Medical Devices: Implications for FDA Regulation of Digital Health. *Milbank Quarterly*, *95*(3), 535–553. https://doi.org/10.1111/1468-0009.12278

Sandler, K., Ohstrom, L., Moy, L., & McVay, R. (2010). *Killed by Code : Software Transparency in Implantable Medical Devices*. *Heart*.

Santos, S. (2019, January). *Virutal health clinics are expanding in China*. Retrieved from AndroidPIT: https://www.androidpit.com/virtual-health-clinics-in-china

Saxon, L. A. (2016). Mobile Health Application Solutions. *Circulation: Arrhythmia and Electrophysiology*, *9*(2), 1–11. https://doi.org/10.1161/CIRCEP.115.002477

Siemens. (2011). Mean Time Between Failures ( MTBF ) Background information on MTBF, 5.

Sorenson, C., & Drummond, M. (2014). Improving medical device regulation: The United States and Europe in perspective. *Milbank Quarterly*, *92*(1), 114–150. https://doi.org/10.1111/1468-0009.12043

Appendix

Tables

Table . Failure Mode Analysis for IoT Medical Device Software Vigilance Flow

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Step | Failure Mode | Failure Cause | Failure Effect | Likelihood Occurrence | Likelihood Detection | RPN | Actions to Reduce Occurrences of Failure |
| 1. 1. Release candidate for medical device software | There is no established repository for code | Code may be stored locally on a machine or server | Loss of code | 7 | 10 | 70 | Have at least to strategies for storing code that include local and remote repository |
| 1. 1. Release candidate for medical device software | There is no versioning strategy for code and there is no form to identify the code that will be released with the device | There is no versioning strategy applied for continuous integration | Loss of code | 7 | 10 | 70 | Implement strategies for code versioning and continuous integration |
| 1. 1. Release candidate for medical device software | Code has been loss due to disaster | Natural disaster that may cause loss of hardware | Loss of code | 7 | 10 | 70 | Have at least to strategies for storing code that include local and remote repository |
| 1.2. Perform software analysis for risks | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 1.2. Perform software analysis for risks | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 1.2. Perform software analysis for risks | Loss of information after delivery by the reviewer | Natural disaster and manmade actions | Delay in process review | 5 | 6 | 30 | Implement strategies for remote and local documentation storing |
| 1.3. Risk detected? | Too many risks detected | Lack of implementation of best practices during software development | Delay in code release | 8 | 8 | 64 | Implement strategies for detecting risks at every iteration of code development cycle |
| 1.3. Risk detected? | No technical capacity for detecting risks | Lack of training or documentation | Increase in cybersecurity risks | 9 | 9 | 81 | Implement training strategies of third party services |
| 1.3. Risk detected? | No detection of high risk issues | Lack of training or documentation | Increase in cybersecurity risks | 9 | 9 | 81 | Implement training strategies of third party services |
| 1.4. Apply risk mitigation methods | No technical capacity for mitigating risks | Lack of training or documentation | Increase in cybersecurity risks | 7 | 8 | 56 | Implement training strategies of third party services |
| 1.4. Apply risk mitigation methods | No financial capacity for mitigating risks | Lack of financial resources | Increase in cybersecurity risks | 9 | 9 | 81 | Implement open-source technologies |
| 1.4. Apply risk mitigation methods | Mitigation methods do not decrease risks | Lack of training or documentation | Increase in cybersecurity risks | 10 | 10 | 100 | Implement training strategies of third party services |
| 1.5. Provide documental evidence of Risk Analysis | Loss of information by the stakeholder | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 1.5. Provide documental evidence of Risk Analysis | Loss of information by receiver | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 1.5. Provide documental evidence of Risk Analysis | Loss of information due to disasters | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 1.6. Release Candidate Digital Signature | Company is not able to issue a certificate | Company may be on first steps of development and foundation | Loss of trust | 6 | 9 | 54 | Digital sign strategies with SHA-254 |
| 1.6. Release Candidate Digital Signature | Loss of Certificate or Digital Signature | Information loss | Loss of code validation | 5 | 4 | 20 | Implement strategies for storing document or keys that include local and remote repository |
| 1.6. Release Candidate Digital Signature | Expiring key | Lack of vigilance over key management processes | Software malfunction | 5 | 10 | 50 | Have programmed processes for replacing keys |
| 1.7. Document Code | Loss of information by the stakeholder | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 1.7. Document Code | Loss of information by receiver | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 1.7. Document Code | Loss of information due to disasters | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 1.8. Release Software documentation | Loss of information by the stakeholder | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 1.8. Release Software documentation | Loss of information by receiver | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 1.8. Release Software documentation | Loss of information due to disasters | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 2.1. Is this a new use case for device? | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 2.1. Is this a new use case for device? | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 2.1. Is this a new use case for device? | Insufficient knowledge on business rules | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 2.2. Review previous use cases and incorporate previous risk analysis | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 2.2. Review previous use cases and incorporate previous risk analysis | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 2.2. Review previous use cases and incorporate previous risk analysis | Loss of information after delivery by the reviewer | Natural disaster and man-made actions | Delay in process review | 5 | 6 | 30 | Implement strategies for remote and local documentation storing |
| 2.3. Define and document use cases | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 2.3. Define and document use cases | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 2.3. Define and document use cases | Insufficient knowledge on business rules | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 2.4. Incorporate: Software documentation, Risk analysis, Use Cases | Loss of information by the stakeholder | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 2.4. Incorporate: Software documentation, Risk analysis, Use Cases | Loss of information by receiver | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 2.4. Incorporate: Software documentation, Risk analysis, Use Cases | Loss of information due to disasters | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 3.1. Review documentation and release candidate | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 3.1. Review documentation and release candidate | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 3.1. Review documentation and release candidate | Loss of information after delivery by the reviewer | Natural disaster and manmade actions | Delay in process review | 5 | 6 | 30 | Implement strategies for remote and local documentation storing |
| 3.2. Following best practices? | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 3.2. Following best practices? | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 3.2. Following best practices? | Insufficient knowledge on business rules | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 3.3.Correct development process | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 3.3.Correct development process | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 3.3.Correct development process | Insufficient knowledge on business rules | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 3.4. Compliance with certification model? | Insufficient knowledge on current certification | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 3.4. Compliance with certification model? | Insufficient documentation | Lack of strategies for storing documentations | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 3.4. Compliance with certification model? | No agreement on certification model by reviewer | Lack of regulatory authorities involvement and participation | Delay in process review | 7 | 7 | 49 | Request requirement before submission |
| 3.5. Provide certification compliance evidence | Loss of information by the stakeholder | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 3.5. Provide certification compliance evidence | Loss of information by receiver | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 3.5. Provide certification compliance evidence | Loss of information due to disasters | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 3.6. Incorporate: Software documentation, Risk analysis, Use Cases, Certification Compliance | Loss of information by the stakeholder | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 3.6. Incorporate: Software documentation, Risk analysis, Use Cases, Certification Compliance | Loss of information by receiver | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 3.6. Incorporate: Software documentation, Risk analysis, Use Cases, Certification Compliance | Loss of information due to disasters | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code and documents that include local and remote repository |
| 4.1. Review documental evidence | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 4.1. Review documental evidence | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 4.1. Review documental evidence | Loss of information after delivery by the reviewer | Natural disaster and manmade actions | Delay in process review | 5 | 6 | 30 | Implement strategies for remote and local documentation storing |
| 4.2. Software complies with certification standards? | Insufficient knowledge on current certification | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 4.2. Software complies with certification standards? | Insufficient documentation | Lack of strategies for storing documentations | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 4.2. Software complies with certification standards? | No agreement on certification model by reviewer | Lack of regulatory authorities involvement and participation | Delay in process review | 7 | 7 | 49 | Request requirement before submission |
| 4.3. Issue recommendations | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 4.3. Issue recommendations | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 4.3. Issue recommendations | Loss of information after delivery by the reviewer | Natural disaster and manmade actions | Delay in process review | 5 | 6 | 30 | Implement strategies for remote and local documentation storing |
| 4.4. Device Registry with software digital signature | No strategy for device registry | Lack of agreement between Regulators & Associations | Delay in process | 5 | 10 | 50 | Implement a common repository for device registry |
| 4.4. Device Registry with software digital signature | Loss of information due to disasters | Storage of information in only one repository | Loss of information | 7 | 10 | 70 | Implement strategies for storing data and documents that include local and remote repository |
| 4.4. Device Registry with software digital signature | Data Breach | Lack of cybersecurity strategies | Data compromised | 4 | 10 | 40 | Implement strategies for data encryption and firewall security |
| 4.5. Unique Device Registry with software digital signature | No strategy for device registry | Lack of agreement between Regulators & Associations | Delay in process | 5 | 10 | 50 | Implement a common repository for device registry |
| 4.5. Unique Device Registry with software digital signature | Loss of information due to disasters | Storage of information in only one repository | Loss of information | 7 | 10 | 70 | Implement strategies for storing data and documents that include local and remote repository |
| 4.5. Unique Device Registry with software digital signature | Data Breach | Lack of cybersecurity strategies | Data compromised | 4 | 10 | 40 | Implement strategies for data encryption and firewall security |
| 4.6. Incorporate: Software documentation, Risk analysis, Use Cases, Certification Compliance, Device Registry | Loss of information by the stakeholder | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code that include local and remote repository |
| 4.6. Incorporate: Software documentation, Risk analysis, Use Cases, Certification Compliance, Device Registry | Loss of information by receiver | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code that include local and remote repository |
| 4.6. Incorporate: Software documentation, Risk analysis, Use Cases, Certification Compliance, Device Registry | Loss of information due to disasters | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code that include local and remote repository |
| 5.1. Device and software review | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 5.1. Device and software review | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 5.1. Device and software review | Loss of information after delivery by the reviewer | Natural disaster and manmade actions | Delay in process review | 5 | 6 | 30 | Implement strategies for remote and local documentation storing |
| 5.2. Sufficient documental evidence? | Loss of information by the stakeholder | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code that include local and remote repository |
| 5.2. Sufficient documental evidence? | Loss of information by receiver | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code that include local and remote repository |
| 5.2. Sufficient documental evidence? | Loss of information due to disasters | Storage of information locally | Loss of documentation | 7 | 8 | 56 | Implement strategies for storing code that include local and remote repository |
| 5.3. Issue recommendations | Information is not sufficient or non-existent | Lack of strategies for storing documentations | Loss of documentation | 5 | 7 | 35 | Implement strategies for remote and local documentation storing |
| 5.3. Issue recommendations | Stakeholder does not have all the knowledge on the process | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 5.3. Issue recommendations | Loss of information after delivery by the reviewer | Natural disaster and manmade actions | Delay in process review | 5 | 6 | 30 | Implement strategies for remote and local documentation storing |
| 5.4. Compliance with existing regulations? | Insufficient knowledge on current regulations | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 5.4. Compliance with existing regulations? | Insufficient documentation | Lack of strategies for storing documentations | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 5.4. Compliance with existing regulations? | No agreement on regulations by government | Lack of regulatory authorities involvement and participation | Delay in process review | 7 | 7 | 49 | Request requirements before submission |
| 5.5. Grant market release of Medical Device | Loss of information after delivery by the reviewer | Natural disaster and manmade actions | Delay in process review | 5 | 6 | 30 | Implement strategies for remote and local documentation storing |
| 5.5. Grant market release of Medical Device | Insufficient knowledge on current regulations | Lack of strategies for knowledge transfer and documentation | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |
| 5.5. Grant market release of Medical Device | Insufficient documentation | Lack of strategies for storing documentations | Delay in process review | 5 | 6 | 30 | Implement checklists of requirements before submitting information |

Table . Failure Mode and Effect Analysis for IoT Medical Device Software Vigilance Flow

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 1.1** | **1** | **Process Step** | Release candidate for medical device software |
| **2** | **Potential Failure Mode** | There is no established repository for code | There is no versioning strategy for code and there is no form to identify the code that will be released with the device | Code has been loss due to disaster |
| **3** | **Potential Cause(s)** | Code may be stored locally on a machine or server | There is no versioning strategy applied for continuous integration | Natural disaster that may cause loss of hardware |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Have at least two strategies for storing code that include local and remote repository. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for code versioning and continuous integration. Apply branching strategy for code. Implement number versioning strategy. | Have at least two strategies for storing code that include local and remote repository. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 1.2** | **1** | **Process Step** | Perform software analysis for risks |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Loss of information after delivery by the reviewer |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Natural disaster and manmade actions |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 1.3** | **1** | **Process Step** | Risk detected? |
| **2** | **Potential Failure Mode** | Too many risks detected | No technical capacity for detecting risks | No detection of high risk issues |
| **3** | **Potential Cause(s)** | Lack of implementation of best practices during software development | Lack of training or documentation | Lack of training or documentation |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | FREQUENT | FREQUENT | FREQUENT |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | ELIMINATE | ELIMINATE |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for detecting risks at every iteration of code development cycle. Implement quality assurance strategies at all cycles of code development. Implement acceptance criteria for all requirements. | Implement training strategies of third party services. Implement quality assurance strategies at all cycles of code development. Implement acceptance criteria for all requirements. | Implement training strategies of third party services. Implement quality assurance strategies at all cycles of code development. Implement acceptance criteria for all requirements. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 1.4** | **1** | **Process Step** | Apply risk mitigation methods |
| **2** | **Potential Failure Mode** | No financial capacity for mitigating risks | Mitigation methods do not decrease risks | Loss of information by the stakeholder |
| **3** | **Potential Cause(s)** | Lack of financial resources | Lack of training or documentation | Storage of information locally |
| **4** | **Severity** | MAJOR | MAJOR | MODERATE |
| **5** | **Probability** | COMMON | FREQUENT | FREQUENT |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | ELIMINATE | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement training strategies of third party services. Implement quality assurance strategies at all cycles of code development. Implement acceptance criteria for all requirements. | Implement open-source technologies. Implement quality assurance strategies at all cycles of code development. Implement acceptance criteria for all requirements. | Implement training strategies of third party services. Implement quality assurance strategies at all cycles of code development. Implement acceptance criteria for all requirements. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 1.5** | **1** | **Process Step** | Provide documental evidence of Risk Analysis |
| **2** | **Potential Failure Mode** | Loss of information by the stakeholder | Loss of information by receiver | Loss of information due to disasters |
| **3** | **Potential Cause(s)** | Storage of information locally | Storage of information locally | Storage of information locally |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 1.6** | **1** | **Process Step** | Release Candidate Digital Signature |
| **2** | **Potential Failure Mode** | Company is not able to issue a certificate | Loss of Certificate or Digital Signature | Expiring key |
| **3** | **Potential Cause(s)** | Company may be on first steps of development and foundation | Information loss | Lack of vigilance over key management processes |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | OCCASIONAL | FREQUENT |
| **7** | **Action** | CONTROL | ACCEPT | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Digital sign strategies with SHA-254. User versioning in scripts. Add build number to code. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Have programmed processes for replacing keys. Add alerts to stakeholder in charge of task. Have a strategy for reminder. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 1.7** | **1** | **Process Step** | Document Code |
| **2** | **Potential Failure Mode** | Loss of information by receiver | Loss of information due to disasters | Loss of information by the stakeholder |
| **3** | **Potential Cause(s)** | Storage of information locally | Storage of information locally | Storage of information locally |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 1.8** | **1** | **Process Step** | Release Software documentation |
| **2** | **Potential Failure Mode** | Loss of information by the stakeholder | Loss of information by receiver | Loss of information due to disasters |
| **3** | **Potential Cause(s)** | Storage of information locally | Storage of information locally | Storage of information locally |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 2.1** | **1** | **Process Step** | Is this a new use case for device? |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Insufficient knowledge on business rules |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Lack of strategies for knowledge transfer and documentation |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 2.2** | **1** | **Process Step** | Review previous use cases and incorporate previous risk analysis |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Loss of information after delivery by the reviewer |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Natural disaster and manmade actions |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 2.3** | **1** | **Process Step** | Define and document use cases |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Insufficient knowledge on business rules |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Lack of strategies for knowledge transfer and documentation |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 2.4** | **1** | **Process Step** | Incorporate: Software documentation, Risk analysis, Use Cases |
| **2** | **Potential Failure Mode** | Loss of information by the stakeholder | Loss of information by receiver | Loss of information due to disasters |
| **3** | **Potential Cause(s)** | Storage of information locally | Storage of information locally | Storage of information locally |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 3.1** | **1** | **Process Step** | Review documentation and release candidate |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Loss of information after delivery by the reviewer |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Natural disaster and manmade actions |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 3.2** | **1** | **Process Step** | Following best practices? |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Insufficient knowledge on business rules |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Lack of strategies for knowledge transfer and documentation |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 3.3** | **1** | **Process Step** | Correct development process |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Insufficient knowledge on business rules |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Lack of strategies for knowledge transfer and documentation |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 3.4** | **1** | **Process Step** | Compliance with certification model? |
| **2** | **Potential Failure Mode** | Insufficient knowledge on current certification | Insufficient documentation | No agreement on certification model by reviewer |
| **3** | **Potential Cause(s)** | Lack of strategies for knowledge transfer and documentation | Lack of strategies for storing documentation | Lack of regulatory authorities involvement and participation |
| **4** | **Severity** | MODERATE | MODERATE | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Request requirement before submission. Create processes for knowledge transfer. Request peer review on checklists. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 3.5** | **1** | **Process Step** | Provide certification compliance evidence |
| **2** | **Potential Failure Mode** | Loss of information by the stakeholder | Loss of information by receiver | Loss of information due to disasters |
| **3** | **Potential Cause(s)** | Storage of information locally | Storage of information locally | Storage of information locally |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 3.6** | **1** | **Process Step** | Incorporate: Software documentation, Risk analysis, Use Cases, Certification Compliance |
| **2** | **Potential Failure Mode** | Loss of information by the stakeholder | Loss of information by receiver | Loss of information due to disasters |
| **3** | **Potential Cause(s)** | Storage of information locally | Storage of information locally | Storage of information locally |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 4.1** | **1** | **Process Step** | Review documental evidence |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Loss of information after delivery by the reviewer |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Natural disaster and manmade actions |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 4.2** | **1** | **Process Step** | Software complies with certification standards? |
| **2** | **Potential Failure Mode** | Insufficient knowledge on current certification | Insufficient documentation | No agreement on certification model by reviewer |
| **3** | **Potential Cause(s)** | Lack of strategies for knowledge transfer and documentation | Lack of strategies for storing documentation | Lack of regulatory authorities involvement and participation |
| **4** | **Severity** | MODERATE | MODERATE | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Request requirement before submission. Create processes for knowledge transfer. Request peer review on checklists. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 4.3** | **1** | **Process Step** | Issue recommendations |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Loss of information after delivery by the reviewer |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Natural disaster and manmade actions |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 4.4** | **1** | **Process Step** | Device Registry with software digital signature |
| **2** | **Potential Failure Mode** | No strategy for device registry | Loss of information due to disasters | Data Breach |
| **3** | **Potential Cause(s)** | Lack of agreement between Regulators & Associations | Storage of information in only one repository | Lack of cybersecurity strategies |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | OCCASIONAL |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement a common repository for device registry. Implement a standard format for information sharing, such as JSON or XML. Implement a web service able to allow end users to register devices. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for data encryption and firewall security. Implement role based access. Implement transparent encryption on database. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 4.5** | **1** | **Process Step** | Unique Device Registry with software digital signature |
| **2** | **Potential Failure Mode** | No strategy for device registry | Loss of information due to disasters | Data Breach |
| **3** | **Potential Cause(s)** | Lack of agreement between Regulators & Associations | Storage of information in only one repository | Lack of cybersecurity strategies |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | OCCASIONAL |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement a common repository for device registry. Implement a standard format for information sharing, such as JSON or XML. Implement a web service able to allow end users to register devices. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for data encryption and firewall security. Implement role based access. Implement transparent encryption on database. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 4.6** | **1** | **Process Step** | Incorporate: Software documentation, Risk analysis, Use Cases, Certification Compliance, Device Registry |
| **2** | **Potential Failure Mode** | Loss of information by the stakeholder | Loss of information by receiver | Loss of information due to disasters |
| **3** | **Potential Cause(s)** | Storage of information locally | Storage of information locally | Storage of information locally |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for storing code that include local and remote repository. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for storing code that include local and remote repository. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for storing code that include local and remote repository. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 5.1** | **1** | **Process Step** | Device and software review |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Loss of information after delivery by the reviewer |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Natural disaster and manmade actions |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 5.2** | **1** | **Process Step** | Sufficient documental evidence? |
| **2** | **Potential Failure Mode** | Loss of information by the stakeholder | Loss of information by receiver | Loss of information due to disasters |
| **3** | **Potential Cause(s)** | Storage of information locally | Storage of information locally | Storage of information locally |
| **4** | **Severity** | MAJOR | MAJOR | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | FREQUENT | FREQUENT | FREQUENT |
| **7** | **Action** | CONTROL | CONTROL | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for storing code that include local and remote repository. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for storing code that include local and remote repository. Create a procedure for constant backup. Implement repository mirroring process. | Implement strategies for storing code that include local and remote repository. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 5.3** | **1** | **Process Step** | Issue recommendations |
| **2** | **Potential Failure Mode** | Information is not sufficient or non-existent | Stakeholder does not have all the knowledge on the process | Loss of information after delivery by the reviewer |
| **3** | **Potential Cause(s)** | Lack of strategies for storing documentation | Lack of strategies for knowledge transfer and documentation | Natural disaster and manmade actions |
| **4** | **Severity** | MAJOR | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 5.4** | **1** | **Process Step** | Compliance with existing regulations? |
| **2** | **Potential Failure Mode** | Insufficient knowledge on current regulations | Insufficient documentation | No agreement on regulations by government |
| **3** | **Potential Cause(s)** | Lack of strategies for knowledge transfer and documentation | Lack of strategies for storing documentation | Lack of regulatory authorities involvement and participation |
| **4** | **Severity** | MODERATE | MODERATE | MAJOR |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | CONTROL |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Request requirements before submission. Create processes for knowledge transfer. Request peer review on checklists. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Process Step 5.5** | **1** | **Process Step** | Grant market release of Medical Device |
| **2** | **Potential Failure Mode** | Loss of information after delivery by the reviewer | Insufficient knowledge on current regulations | Insufficient documentation |
| **3** | **Potential Cause(s)** | Natural disaster and manmade actions | Lack of strategies for knowledge transfer and documentation | Lack of strategies for storing documentation |
| **4** | **Severity** | MODERATE | MODERATE | MODERATE |
| **5** | **Probability** | COMMON | COMMON | COMMON |
| **6** | **Hazard Score** | COMMON | COMMON | COMMON |
| **7** | **Action** | ACCEPT | ACCEPT | ACCEPT |
| **8** | **Actions to Reduce Occurrences of Failure** | Implement strategies for remote and local documentation storing. Create a procedure for constant backup. Implement repository mirroring process. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. | Implement checklists of requirements before submitting information. Create processes for knowledge transfer. Request peer review on checklists. |

Table . Quality Measures for Monitoring Software for IoT Devices

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Step | Name | Description | Owner | Report to | Calendar Measure |
| 1 | Errors per KLOC | Non-corrected functional errors in code found by quality assurance team per 1000 lines of code. | QA Team | Management Team | At every code release |
| Defect per KLOC | Non-corrected non-functional errors found by quality assurance team per 1000 lines of code. | QA Team | Management Team | At every code release |
| Application Crash Rate | Number of times applications fails after a determined number of uses. ACR = F/U | QA Team | Management Team | At every code release |
| 2 | Risks per Use Case | Number of errors found per use cases reported by QA team at code release. Sum of Errors per KLOC and Defect per KLOC/Use case | Management Team | Company | At every code release |
| Patching Cadence | Amount of time needed in order to respond to incidents and patching cadence | Management Team | Company | At every code release |
| 3 | Non-compliant certification reasons | Enumeration of reasons for not complying with certification models | Company | Regulators & Associations | At every code release |
| Endpoint Incidents | Report of endpoint incidents that had been detected over a period of time | Company | Regulators & Associations | 6 months |
| Mean Time to Repair | Average span of time that involves incident discovery to when a working remedy is deployed. Time line starts at day 0 of suspected incident discovery until code release. | Company | Regulators & Associations | 1 year |
| 4 | Non-compliant certification reasons | General report for reasons detected for non-compliance of certification standards | Regulators & Associations | Government | 6 months |
| Issued Recommendations | General report of issued recommendations | Regulators & Associations | Government | 6 months |
| 5 | Reasons for non-compliance | General report for reasons detected for non-compliance of certification standards | Government |   | 1 year |
| Issued Recommendations | General report of issued recommendations | Government |   | 1 year |

Figure 1. Structure Information Flow for the Management of Software for Medical Devices



Figure 1. This figure is a schematic representation of the Structure Information Flow for the Management of Software devices. This model is based on the approach proposed by Rasmussen & Svedung (2000) on which it has six levels of hierarchy. On the proposed model, there are 5 levels of interaction that includes the following stakeholders: government; regulators & associations; company; management; and, work.

Figure 2. IoT Medical Device Software Vigilance Flow Diagram

Figure . The image is the graphical representation of the series of proposed steps that can be followed during the process of IoT medical software vigilance process. The process is divided by its main actors and their tasks.