doi:10.20944/preprints201912.0130.v1

Software Quality Assurance in Safety Critical Systems

Mehreen Sirshar
Faculty of Software Engineering
Fatima Jinnah Women
University
Rawalpindi, Pakistan
mehreensirshar@fjwu.edu.pk

Ghanwa Ejaz
Department of Software Engineering
Fatima Jinnah Women
University
Rawalpindi, Pakistan
ghanwaejaz2@gmail.com

Maria Mumtaz
Department of Software Engineering
Fatima Jinnah Women
University
Rawalpindi, Pakistan
mariamumtaz221@gmail.com

Zoya Alam
Department of Software Engineering
Fatima Jinnah Women
University
Rawalpindi, Pakistan
zoyabibi555@gmail.com

***Abstract*— The complex systems that require safety are the Safety Critical Systems. Maintaining these systems is a big challenge. Now a days, safety is a very critical requirement for the latest systems. Safety critical systems must be safe. Different approaches to ensure quality and safety in safety critical systems has been discussed in this paper. A comparison is also conducted between these various approaches. Safety critical systems must remain more influential in future.

***Keywords* –** risk management; safety critical systems; safety assessment; methodology

I. INTRODUCTION

Those software whose failure result in some kind of harm such as loss, damage to life, property or environment are known as safety critical system. Life critical system is another name of safety critical system and also important for all the applications field i.e. railway, automobile, air defense, medical and nuclear etc. As the time passes these systems become vital part of our life. All the medical equipment like radiation therapy machine, heart-lung machine, Insulin pumps and robotic surgery that are common today in medical science are all safety critical system. Transportation means like automobiles have a system for the brakes and also the steering of car. In addition we have also life support systems inside airplanes. Another safety critical system is autopilot. We have also fire alarms, automatic doors, amusement in our daily life. These are all safety critical systems as their malfunction can cause life injury or property damage.

As we know that earlier software's need more security but today because of the safety critical system software are measured more secure and reliable. Software critical systems are related to engineering like industrial. We considered them part of software development because of the computer based operation of safety critical systems. The most important concern of the safety critical system is the failure or malfunction because it can be more dangerous. There are many techniques used for such problems like measure and overcome the risk. For safety critical system there are many architecture introduced for reducing the risk.

We have conferred different approaches in this paper that are used to overcome the risk of safety critical systems such as "Control based systematic approach", "HSI case model", "Systematic mapping approach", " Communicational Architectures", "Conventional Approach", "Model Based Safety Assessment", "Model-Combinatorial based testing". Moreover we have also included a comparative study of different approaches for the safety critical system. Verification and validation techniques are developed for the medical software. For the improvement of safety and reliability Formal methods such as "Theorem proving" and "Model checking" are used for the medical software's. Different fields uses different standard for the software quality.

II. LITERATURE REVIEW

The author presented the "Control Based Systematic Approach" in this paper. This approach consists of two metrics that is PSC and TSC. Another method is also launched that relates the result from PSC and infer result to TSC. This approach is successfully practical on the runway system for safety valuation in this paper. With the help of this example the author confirmed the three stage method. [1]

Human System Integration (HSI) case model for safety critical system was presented by another author. This case model applied in air defense project. The whole model had four stages. Reduce the risk is the major task of HSI model. [2]

Another important work presented by author for safety critical system is the systematic mapping. Combining the safety critical system with the agile development is the major chore of this mapping. Summary of prevailing research, benefits, challenges and outcome of agile development was provided by the author in this paper. Author introduced two methodologies that is a cross company workshop and second one is mapping study. The workshop was attended by the six Swedish companies. The purpose of mapping study is to guide the research. [3]

The most difficult task in the critical software industries is to provide a safety and reliability. In another paper the author improved the assessment model by several architectures between different organizations and within organization. But it also had several safety problems and a stimulating task for the developer. To improve the communication architectures the author provides comprehensive analysis of protection system. [4]

The design of operating system for the safety critical systems is explained in this paper. The product must conform safety standards. Commercial safety critical systems cannot be re used and they are non-flexible. Different cores of operating system are identified by the author such as interface and scheduler. Its purpose is that without wasting the time on changing the core part of operating system, it can be reused. Reusing thirty percent of operating systems is shown by the results. The use of inheritance and polymorphism ensures hardware independency. [5]

Model Based Safety Assessment (MBSA) is discussed in detail by the author in this research paper. The complexity in safety critical systems has reduced drastically and the efficiency is improved. The author has also discussed the relationship between MBSA and the traditional safety assessment process. A validation method is also suggested for safety property. Model checking and theorem proving, two tools for formal verification, are also presented in this paper. It is more easy to use model checking tool. Ultimately, using modeling and verification process, a demonstration is given for the flight control system case study. [6]

Conventional approaches are presented in this paper to gain quality. In the whole process of developing software, methods are used for assuring software quality. [7]

Model-Combinatorial based testing (“MCbt”), for verifying safety crucial systems is presented in this paper by the author. To test the safety critical systems, this approach is used. The features of model-based testing, combinatorial testing and safety analysis that is used to test the system thoroughly are combined in this methodology. “MCbt” tasks using SPEM 2.0 meta models are also presented. To examine its role in certification process, MCbt is applied to different case studies. [8]

Another model in which acceptance, trust and the factors that are impacting the use of a safety critical technology is included is described by the author of this paper. The acceptance of computer is described by the Technology Acceptance Model (TAM) in IT field. The data of this model has been given by the surveys and previous technology acceptance models. It has then been authenticated by using the previous research documentation. Numerous parts of TAM are contained by Vehicle Autonomy and Intelligence Lab (VAIL). [9]

The architecture of safe and secure device named as “SEnSE” is presented by the author of another paper. “SEnSE” architecture has an important property of validation of function. Safety Cloud Communication Interface, “SEnSE” ID, Handshake Procedure Interface, Local Network Communication Interface and cyber-physical System Communication Bus Interface is consisted by “SEnSE” architecture. An architecture to shape up a common structure for future systems is mentioned by the author In case of future plans. [10]

A formal development procedure which is based on the Abstract State Machine (ASM) is presented by the author in another paper. At every level in the development phase of safety critical systems, different validation and verification activities are carried out. The process for difficult development of software for medical devices is introduced by this paper. The Validation & Verification activities can be planned

doi:10.20944/preprints201912.0130.v1

and performed simultaneously by the devices along the software life cycle. Defect prevention is always aimed at. The validation and verification of the system can be demonstrated by the device manufacturer. The application of the proposed process that is used for measuring the patient's stereoacuity has shown by the author. [11]

A real time scheduling protocol that uses a modeling language program for the safety assessment has described by the author. Propositional Projection Temporal Logic formulas are used by it. Using the formal verification methods such as with theorem proving and more, these properties are proved. In theorem proving method, the properties of the system and the system itself are expressed as formulas such as PVS, HOL etc. This modeling system is divided into four modules which are the clock module (Clock), task module (Task), interrupt module (Intort) and scheduler module (Scheduler). In the future author continue to improve the tool. [12]

Requirement elicitation and analysis are the most critical phases of software development life cycles. So many software fails because of this. Now a day's Avionic software development is a significant and growing area of work. The author describes a model based development life cycle for the avionic software. [13]

In this paper another author introduced two different techniques for catastrophe circumstances. The first one is "generalized stochastic Petri nets (GSPN)" and the second is "Fault Tree driven Markov processes (FTDMP)". FTDMP consists of fault tree and markov process but this method does not provide extra material. GSPN provides other statistics. [14]

Correctness and completeness of environmental expectations related problems contributed many accidents. When modification occurred in product after release then the problem becomes exacerbated. A new technique is described by this paper in which many certifying bodies require traceability are exploited by it. It provided the methods to mitigate the issues. Safety analyst is supported by the contribution of the work with useful information for the access of validation of environment related assumptions for new product. [15]

Every domain needs safety critical systems. Ensuring that expected services will be delivered by this system to its users is the main purpose of this process. In the past decades, dependability assessment of safety critical control systems has been devoted by the significant amount of attention. For the evaluation of such system, there is no proper system or technique. A literature survey is provided by this paper. Analysis of limitation of different elements is also discussed in it. Researchers are also facilitated by this work to put this into practice for the advancement of research. [16]

There is a major concern of Off-nominal behaviors in the area of embedded systems and safety critical systems. Model-based approaches have been proposed by some researchers to address ONB problems which analyze natural language requirement document to expose ONBs. While promising results are produced by these approaches, a lot of human effort and are required by them. A combinational based approach is proposed by this paper for the reduction of human effort and time. By using IPOG algorithm, structured requirements patterns and combination are used by Combinational Causal Component Model. The result of this paper indicates that human effort and time can be reduced by the proposed approach. [17]

In 2016, the use of declarative streaming languages is discussed in a research work. StreamQRE is a declarative streaming programming language used for the analyzing real-time streaming applications. It is efficient and portable implementation. Constant cost is guaranteed by its evaluation algorithm per data item and upper bounds are also calculated on per-item cost. Algorithmic possibilities are explored by this cost estimate, on the basis of this hardware can be designed. [18]

For the development and maintenance of safety critical systems, requirements engineering is very important. To discuss the approaches for capturing and identifying safety requirements and for determining the challenges, literature is studied and the experts are interviewed by the researchers. [19]

Requirements elicitation and analysis which is the most important phase of software development is given special emphasis by all the software development life cycle methodologies. This is because the failure to manage requirements could lead to total project failure. Avionic system development is very important nowadays. In 2016, a research effort for model-based software development life-cycle for avionic systems is defined in a research paper, and it focuses on the phase of gathering requirements and modeling. [20]

Two important questions related to safety critical software engineering and software intensive systems engineering are addresses by this study. The first question is safety critical should be considered by which software and the other one is which processes, design and tools have been used by the practitioner to build a system. Their answers are given by this study in which unstructured interviews are analyzed with experienced people who involved in such systems. Guidance is provided by the results of this study to those systems for policymakers, corporate governors and insurance executives. [21]

In this paper the author confers using the agile system development methods, how the safety should be guaranteed in complex system. For designing the complex software the probe-sense-learn approach is used which are best for the agile software development. Author discusses how the quality assurance is integrated into agile software development. Those software systems that are refined by the agile teams will require verification and validation by individual member for the safety critical system. In last the author discusses the current quality practices for medical device software. [22]

III. Safety Assessment Techniques

A. Control based systematic approach

Two system control safety metrics and a three stage method are used for this approach for comparative result of the system.

- *Probabilistic System Control (PSC):* The controller work with the process to achieve its goal in safety controller system. The metric PSC are used for reliability and failures of risk are present at the controller side.
- *Temporal System Control (TSC):* This metric is also used for reliability and in this metric risk of failures are on the process side.
- *Three stage Method:* In this method, the first step is to recognize all safety critical process. As we know that these systems are at risk and they cause destruction so they should be detecting and inspected properly. Then next step is to diagnose the control model for safety critical process. Through system modeling and comparison with simulation, the control model can be done. To confirm the maximum safety this is used for all process. Then at last control capacity should be evaluate.

B. Human System Integration (HSI) case model

Most of the accidents are triggered by the human factors. For reducing the safety risk that are caused by human factor, HSI is introduced. This approach includes the integration concerns such as man-power, training human factor engineering etc. Shortly there are no essential requirements to examine the human factors. It is a step-by-step method. This approach consists of four process.

- Stage-1: This stage can be done easily. In this it identifies the needs of human and their limitation. Also the human issues are discussed in HSI model. To describe the human factors multiple international standard are used.
- Stage-2: This stage controls the description of technical functionality such as communication devices and training services. Several techniques are available that includes modelling, simulation etc.

  In this stage technical functionality description like trainings services and communication devices are controlled. Techniques like modeling, simulation etc. are also available.
- Stage-3: HSI provides recommendations regarding time, manpower and power for system and safety development.
- Stage-4: Different trials starts at this stage.

In the last three years "Friend Protection" is the most vital area related to human factors. For supporting the human factors, it becomes the clatter between customer, designer and budget owner. Different standards are used for the human factor.

C. Systematic Mapping Approach

Agile method, a very popular technique presently, is based on incremental development, uses systematic mapping approach discussed in this paper. Each phase can be tested in agile development. The development of embedded system is enhanced due to agile method. Preparation, conducting and reporting the review are the three stages involved in this approach.

- *Search Strategy*: First of all, the search term is presented by the author that is related to agile development. The search terms that are not added into the document are not reserved. Otherwise they will be reserved.
- *Inclusion and Exclusion criteria*: For the search terms, the author used the inclusion and exclusion criteria. In the start the author selected 1986 papers and by gradually

limiting the papers, only 34 papers are selected in the end.

- *Data Extraction and synthesis*: For the selected papers predefined templates are given through which the information required for examination can be extracted. The attained result is evaluated and grouped according to the templates.
- *Limitations and threat to validity*: Search terms are limited to "Elsevier Scopus" database. This is a very complete database.

Agile method of safety critical system provides different advantages.

- Sponsor participation are improved.
- Safety philosophy are also improved.
- It results in improved prioritization.
- It results in cost reduction.
- It also enhanced the quality.
- It provides the chances for reuse.

With advantages there are also some challenges in agile method of safety critical system.

- Difficulty in managing the knowledge stream between different sponsors.
- In agile methods there is a lack of belief.
- Inflexibility in the regulation of time for upfront planning.

D. *Communication Architectures*

All the possible ways in which different components of different systems are interconnected are identified by this approach. Single controller architecture as well as multiple controller architectures is consisted by safety critical system. In multi controller architectures there is an additional "communication" function block. There are more flexible Varied kinds of communication. "Failure Modes and Effect Analysis (FSMA)", "Fault Tree Analysis (FTA) and "Reliability Block Diagram (RBD)" techniques are there for the safety of safety critical system. For safety analysis, control and protection system is discussed by the Author. The aim of "reactor control and protection system" is to provide protection in emergency and controlling the unit power. It has three layers.

- Neutron Flux Monitoring Systems (NFMS) layer: Only 2 sets of "NFMS"
- Reactor Trip Systems (RTS) and Reactor Power Control and Limitation Systems (RPCLS) layer: Many sets of "RTS and RPCLS"
- Rod Control Systems (RCS) layer: Only 1 set of "RCS"

E. *Conventional Approaches*

Conventional approaches are used for achieving the system quality. Three types of plan can be established for any system that is functional requirements, quality requirements and resource requirement. Quality is the most common concern of the safety critical system. The quality of the product is concerned by the Quality Model. Based on the method it can categorize into three types. The first is "theoretical model" which is based on the hypothesis. The next is "data control" that is based on statistical analysis. The last is "Combined model". Different standards are defined for the quality of software. In the life cycle of software system, the software quality assurance is associated with the methods, tools, and design and coding. Using V and V model, software quality is ensured. The V model has two sides.

- *Left Side:* The design and verification is shown by Left side. The first step in this process is to collect the requirements and should do some planning for these requirements. In the next step requirements should be analyzed and specified. In next step high level designed of the requirements is developed. Then detailed design or low level design is made. Middle step between the verification and validation is coding.
- *Right Side:* The implementation and validation is shown by the Right side. When the coding is done this process performs its own steps. The first step is module testing this mean individual testing of every component. In second step there is assembly testing of the system. Then whole the system is tested and if there is some error it should be resolved. In the last malfunction or operation is performed.

Verification is concerned with the meeting the requirements of the sponsor done. Because of the verification an error are removed and validation provides the mechanism to improve the errors. Verification process is accomplished by the consecutive execution of view and examination of documentation.

F. *Model based safety assessment*

The complexity of the system design during the development of safety critical systems can be decreased by the Model based safety assessment. System-level safety assessment is supported by the safety critical systems. Thus in order to do so formal system models is required. Hardware, software and mechanical structure of the entire system should be included by them. The basic process of Model based safety assessment includes

- *Normal system modeling:* Normal execution behavior of each component in the system that it is expected to perform is described by it.
- *Failure system modeling:* Failure behavior of every component in the system is described by it.
- *Model Combination:* Failure models are combined by it with the nominal models of each component in the system.
- *Model Transformation:* Models of each component into specific automata is transformed by it.
- *Safety Analysis:* Simulation and formal verification is done of the models so that failures can be identified.
- *Output:* Transforms result into safety artifacts

Safety modeling based on model checking consists of the following

a) *Model Checking Definition:* It is most important part of formal method for safety assessment. Finite State Machine (FSM) in the model checking is also called ”Kripke structure”. Temporal Logic is used to define the system description. It is divided into two types: “Linear Temporal Logic (LTI)” and “Computation Tree Logic (CTI)”.

b) *System Modeling Method:* System description depends on the language being used. It consists of the following steps. Step 1 defines the basic structure of the system and each part defines the “Module”. Second step performs the internal work; define the input and output and also the variables type. In the 3rd step Keywords such as “DEFINE” and “ASSIGN” defines the constituent behavior, constituent failures.

c) *Definition of System Property:* To choose the CTL or LTL depends on the features of system. Current state can be defined by the CTL.

G. *Model-Combinatorial based testing (MCbt)*

Model based testing, combinatorial testing are combined in this technique. System testing takes much time as theses are two separate techniques, but both testing can be done simultaneously by using this model.

1. *Model-Based Testing:* For the description of system behavior, models are used. Component modeling is done first and then, according to models, test designing is done. It is a five step process:

- *Model building and design:* For each component of the system, a model is designed. System behavior is shown in this way.
- *Define test selection criteria:* On the basis of model testing, test selection criteria is defined.
- *Test paths generation:* To generate the fault trees, test paths are generated by the model.
- *Concrete test cases generation:* On the basis of system inputs, test cases are defined.
- Test case execution: To identify the failures, test cases are executed and the result is analyzed.

2. *Combinatorial Testing:* Combinatorial testing includes all pair wise testing during software development. Because it might be the case that a component working well by itself, fails to work correctly when combined with other components.

The MCbt includes five tests as given below:

- Task 1: Behavioral models building for each system component. It shows the normal behavior of a specific component.
- Task 2: For each component model, identify test paths, for safety analysis.
- Task 3: To show the dependency of one component to another, test paths are related together by combinatorial testing.
- Task 4: Perform safety analysis for fault tree building to identify failures.
- Task 5: On the basis of fault trees, combination of test paths are classified.

H. *SEnSE Architecture*

For the embedded safety critical systems, this architecture was introduced. Safety and security of the system are its main concerns. It consists of the following parts

- *A Safety Cloud Communication Interface:* This interface is introduced so that whether a certain function should be executed or not, all the services can communicate within themselves. This is very important as there can be a safety critical function that only one component can identify. If there is lack communication mechanism, execution of other components will be done.

- *SEnSE ID:* For safety purposes, this ID was introduced. Using this ID can activate the SEnSE device. In case a failure occurs, identification of a specific device can be useful.
- *Handshake Interface:* Handshake features allows the safe execution of the SEnSE device. Its accurate execution can minimize risks.
- *Local Network Communication Interface:* The communication between the local components of subsystems is established by this interface.
- *Communication Bus Interface:* Communication lines are used to communicate the subsystems of a device. Interface for communication is provided by this architecture.

IV. Comparative Analysis

We have discussed a lot of safety assessment approaches for the safety critical systems in this paper, and it is clear that modeling each component is the main focus in almost all of the techniques. Modeling and theorem proving is the formal method of safety assessment. TSC and PSC metrics are used in "Control-based Systematic Approach" for the comparison of risk level or failure. This is done by each component modeling and then applying metrics. This is a design level approach.

Another popular approach for developing safety critical systems is "Model based safety assessment", models are built along with their failure models for each component. For failure reduction, this approach uses different algorithms. But combinatorial testing is not allowed by this approach, so it is not only the reliance of developers, dependency between the components have to be tested separately. Hence, another model is presented that is a combination of model testing and combinatorial testing. Test paths identification should be done after model creation according to this model. Using safety analysis, through this technique, combinatorial failures can be identified and then removed. It is a two in one approach, so it saves time and give accurate results.

"Human Integration Service" is another technique discussed in this paper which is inclined towards human faults for system failure. Carelessness is the main cause of error in some cases, but the design of the system should have the tendency to find an alternate solution or a path in case of failure. Also the system should restart instead of halting whenever it is driven through wrong inputs.

In this paper, safety assessment is our main focus and it is affected by various quality attributes of system that includes efficiency, reliability, safety, security etc. Verification and validation should be done at every stage of system development to avoid any errors. Quality attributes of system are ensured by V&V model that can be used together with agile system development. The another approach we have discussed is to make sure that every component of the system should communicate properly because if the system components cannot communicate as intended, it can cause risk of system failure. The analysis performed on the given techniques is also represented in the form of table given below.

Author(Reference)	**Proposed Method/Technique**	**Case Study**	**Dependability Attributes**	**Methodology**
Jingjing Guo[1]	Control Based Systematic Approach	Runway Incursion	Reliability, Availability	Checking system safety states, system model construction, and comparison using TSC and PSC metrics.
C. Semling [2]	Human System Integration Model	Air Defense System	Safety, Maintainability, Dependability	Human issues identification, human factor analysis, system modification.

Eugene Babeshko [4]	Communication Architecture	Reactor Control and protection system	Security, Safety, Dependability, Reliability	Construction of four layers of communication for reliability ensurance.
E.Ph. Jharko [7]	Conventional Approach	Unspecified	Software Quality	V Model for quality ensurance
Jiping Fan [6]	Model-based Safety Assessment	Flight Control System	Reliability, Efficiency	Description of normal execution behavior, failure behavior discussion, model combination, system model transformation, formal verification and result transformation.
Gannous Aiman [8]	Model Combinatorial Based Testing	Certification And Verification	Safety Analysis	Combined features of model based and combinatorial testing.
Amir Klug [10]	SEnSE Architecture	Unspecified	Safe and Secure Integration	Safety Requirements evaluation using handshake operation.
Grant [13]	Model based development life cycle	Unmanned aerial vehicle system	Safety Analysis	Safety requirement for the avionic software.
Talebberrouane [14]	Generalized Stochastic Petri Nets (GSPN) and “Fault Tree driven Markov processes (FTDMP)	KO drum level indicator (LI)	Availability	Using these two techniques to overcome the failure scenarios and provide ssystem availability.
Rahimi [15]	Technique for traceability	Unspecified	Safety Analysis	defined steps for mitigating the identified issues
Raj Kamal [16]	A comprehensive detailed literature survey	Unspecified	Safety Analysis	metrics, threat, means, techniques and methodologies are investigated
Madala [17]	Combinational based approach And IPOG algorithm	Combinational Causal Component Model	Availability	structured requirements patterns and combination are used
Abbas [18]	StreamQRE	Unspecified	Efficiency, Portability	Using declarative streaming programming languages for real time application’s modeling

Martins [19]	Requirement Engineering	Expert Interviews	Safety, Reliability	Capturing and identification of safety requirements
Grant [20]	Model Based Development of Avionic Systems	Unspecified	Reliability	Requirement gathering and modeling
Laplante [21]	Requirement Engineering	Unspecified	Reliability	unstructured interviews are analyzed
T. Mcbride [22]	Probe-sense-learn approach	Medical Device Software	Software Quality	Software quality assurance in agile development system

V. CONCLUSION

The design of the safety critical systems is very complex so engineer finds difficulty in building of such systems. Techniques and approaches to reduce the risk during the operation of such systems are discussed by this paper which is the main interest or purpose of these techniques. It becomes more common in research matter for new approaches to safety critical systems as well as to the industry, By keeping the software simple is one of the best ways for risk reduction. For the success of any software development, the skilled and qualified engineers are vital. Different tools are available for the development of any software. There are many people that are working for the creation of more reliable and efficient safety-critical systems.

VI. REFERENCES

[1] Guo, Jingjing. “A Comparative Safety Assessment Approach for Safety Critical Systems.” 2018 Annual Reliability and Maintainability Symposium (RAMS), 2018, doi:10.1109/ram.2018.8463017.

[2] C. Semling and S. Norton, "The Application of “Human Systems Integration” to Safety-Critical-Systems Design," 9th IET International Conference on System Safety and Cyber Security (2014), Manchester, United Kingdom, 2014, pp. 1-4, doi: 10.1049/cp.2014.0985.

[3] Kasauli, Rashidah, et al. “Safety-Critical Systems and Agile Development: A Mapping Study.” 2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), 2018, doi:10.1109/seaa.2018.00082.

[4] Babeshko, Eugene, et al. “Reliability Assessment of Safety Critical System Considering Different Communication Architectures.” 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT), 2018, doi:10.1109/dessert.2018.8409091.

[5] Delic, Emil, et al. “Platform Independent Safety-Critical Operating System.” 2015 International Conference on Information and Digital Technologies, 2015, doi:10.1109/dt.2015.7222952.

[6] Fan, Jiping, et al. “A Model-Checking Oriented Modeling Method for Safety Critical System.” 2015 First International Conference on Reliability Systems Engineering (ICRSE), 2015, doi:10.1109/icrse.2015.7366490.

[7] Jharko, E.ph. "The Methodology of Software Quality Assurance for Safety-Critical Systems." 2015 International Siberian Conference on Control and Communications (SIBCON), 2015, doi:10.1109/sibcon.2015.7147057.

[8] Gannous, Aiman, et al. "Toward a Systematic and Safety Evidence Productive Verification Approach for Safety-Critical Systems." 2018 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), 2018, doi:10.1109/issrew.2018.00026.

[9] Hutchins, Nathan, and Loyd Hook. "Technology Acceptance Model for Safety Critical Autonomous Transportation Systems." 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC), 2017, doi:10.1109/dasc.2017.8102010.

[10] Hofig, Kai, and Amir Klug. "SEnSE – An Architecture for a Safe and Secure Integration of Safety-Critical Embedded Systems." 2018 26th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 2018, doi:10.23919/softcom.2018.8555740.

[11] Arcaini, Paolo, et al. "Formal Validation and Verification of a Medical Software Critical Component." 2015 ACM/IEEE International Conference on Formal Methods and Models for Codesign (MEMOCODE), 2015, doi:10.1109/memcod.2015.7340473.

[12] Wang, Meng, et al. "Verifying a Scheduling Protocol of Safety-Critical Systems." Journal of Combinatorial Optimization, vol. 37, no. 4, 2018, pp. 1191–1215., doi:10.1007/s10878-018-0343-1.

[13] Grant, Emanuel S. "Requirements Engineering for Safety Critical Systems: An Approach for Avionic Systems." 2016 2nd IEEE International Conference on Computer and Communications (ICCC), 2016, doi:10.1109/compcomm.2016.7924853.

[14] Talebberrouane, Mohammed, et al. "Availability Analysis of Safety Critical Systems Using Advanced Fault Tree and Stochastic Petri Net Formalisms." Journal of Loss Prevention in the Process Industries, vol. 44, 2016, pp. 193–203., doi:10.1016/j.jlp.2016.09.007.

[15] Rahimi, Mona, et al. "Diagnosing Assumption Problems in Safety-Critical Products." 2017 32nd IEEE/ACM International Conference on Automated Software Engineering (ASE), 2017, doi:10.1109/ase.2017.8115659.

[16] Kaur, Raj Kamal, et al. "Dependability Analysis of Safety Critical Systems: Issues and Challenges." Annals of Nuclear Energy, vol. 120, 2018, pp. 127–154., doi:10.1016/j.anucene.2018.05.027.

[17] Madala, Kaushik, et al. "A Combinatorial Approach for Exposing off-Nominal Behaviors." Proceedings of the 40th International Conference on Software Engineering - ICSE 18, 2018, doi:10.1145/3180155.3180204.

[18] Abbas, Houssam, et al. "Real-Time Decision Policies With Predictable Performance." Proceedings of the IEEE, vol. 106, no. 9, 2018, pp. 1593–1615., doi:10.1109/jproc.2018.2853608.

[19] Martins, Luiz Eduardo G., and Tony Gorschek. "Requirements Engineering for Safety-Critical Systems: Overview and Challenges." IEEE Software, vol. 34, no. 4, 2017, pp. 49–57., doi:10.1109/ms.2017.94.

[20] Grant, Emanuel S. “Requirements Engineering for Safety Critical Systems: An Approach for Avionic Systems.” 2016 2nd IEEE International Conference on Computer and Communications (ICCC), 2016, doi:10.1109/compcomm.2016.7924853.

[21] Laplante, Phillip A., and Joanna F. Defranco. “Software Engineering of Safety-Critical Systems: Themes From Practitioners.” IEEE Transactions on Reliability, vol. 66, no. 3, 2017, pp. 825–836., doi:10.1109/tr.2017.2731953.

[22] T. Mcbride and M. Lepmets, “Quality Assurance in Agile Safety-Critical Systems Development,” *2016 10th International Conference on the Quality of Information and Communications Technology (QUATIC)*, 2016.