J. Appl. Environ. Biol. Sci., 4(7S)495-503, 2014

ISSN: 2090-4274
Journal of Applied Environmental and Biological Sciences
www.textroad.com

Gap Analysis in Software Engineering Process Adoption in Implementing High End Embedded System Design

Umm-e-Habiba, Shoab Ahmed Khan, Muhammad Younus Javed

Department of Computer Engineering, College of EME, National University of Sciences and Technology (NUST), Islamabad, Pakistan

Received: September 1, 2014
Accepted: November 13, 2014

ABSTRACT

Software engineering practices are customarily imposed on pure software projects. Embedded systems, where cost of the system, quality and performance are more important factors, practices imposed on software engineering are becoming more important. The embedded system software has to meet very critical reliability requirements and hard real time constraints. To meet these requirements and constraints it is essential to enforce software engineering practices in embedded systems. In this research work it is discussed how process of requirement engineering impact on the whole system development process in context of embedded systems. A framework for requirements specification of embedded systems is developed.

KEYWORDS: Embedded sytem, Requirement engineering, RSD (Requirements specification document).

1 INTRODUCTION

Software engineering techniques are customarily applied on software projects. Embedded systems where the cost of the system, quality and performance are more important factors, software engineering practices are becoming more important. Embedded system is combination of both hardware and software components, which must be specified, design and implemented as well as it has to meet crticical reliability requirements and hard real time constraints. These critical requirements make software engineering practices even more important for embdedded systems. To ensure timely completion of project within specified budget with desire quality and performance it is important that software engineering practices shloud be followed. There is a need for new and improved development methods in embedded system industry to address these important factors. All the development process of the embedded system is depends on well how the requirements engineering process is carried out. More than 50% of the problems occur in embedded system after system is delivered to the customer [1, 2]. A good requirement engineering process is considered as a prerequisite for successful project. Apart from this implementing these requirements in every phase of development is also important. Correct implementation of requirements in embedded system is especially critical where both hardware and software are affected by them. However, there is a no systematic approach to address the requirement engineering process for embedded system. This research is carried out by the means of embedded system development. Though, requirements are specified, testing and evaluation is performed but there is no set format that can ensure correct specification of requirements. Testing of embedded system is another challenging task which should esure full coverage of requirement specification. Testing and validation of these requirements are also performed but there is no set format and not any assurance that each test case covered each requirement.

In this research paper, discussed how the process of requirement engineering impact on whole system development process is discussed.

In section two, problem statement is stated. In section three literature review of existing requirement engineering processes and requirement engineering related to embedded systems is dicussed. In section four research methodology is defined and in section five conclusions and future work is painted.

2 Problem statement:

There are many requirements engineering processes which are customarily applied to software project, but there is no standard process for embedded system. Many research studies have been carried out in the

* **Corresponding Author:** Umm-e-Habiba, Department of Computer Engineering, College of EME, National University of Sciences and Technology (NUST), Islamabad, Pakistan.

past regarding requirement engineering for embedded systems. They focused on different aspects of requirement engineering. This research specifically addresses embedded system development and production division. Requirements specification, testing and evaluation are performed, but there is no set format. There is no any framework which can ensure that each test case cover each requirement specification. There is gap in software engineering process adoption in implementing defense embedded systems. Different problems are identified in requirement specification for embedded system. Requirement for embedded are hard to specify because of the absence of software engineering practices. When any change in requirement occurs, it create inconsistencies among the requirements and system and make it difficult to trace the requirement throughout the development process.

There are a lot of requirement specification tools are available in the market, but these tools can only help you in requirement management. The tool is not a process itself, these tools can't help you in defining requirement engineering process for embedded system.

3 LITERATURE REVIEW

Embedded systems where the reliability requirements and hard real-time contratins are important factors the software engineering practices are becoming more essential to meet these requirements. The existing practices and some new methods are adopted by software engineers to effectively support the particularities of embedded systems [2, 3, 4].

On the basis of the literature review, it is shown that only few requirement engineering practices and tools are available to address the embedded system particularities [5, 6, 7]. As embedded system development is taken, where security and hard real-time constraints come together, which is a great challenge for software engineers. In embedded system development where the security is the main concern, to ensure security it is necessary for embedded software works correctly according to specified requirements [8].

With the help of software engineering practices, we can solve problems by systematically convert the problem into software solution and after conversion, maintenance of the solution can also be carried out using software engineering principals. A project which is developed through using software engineering processes, has to pass through different phases. These are life cycle models of the project. Phases of software engineering were first introduced by Royc [9] in the waterfall model.

There are many other models like spiral model, iterative model, V-shaped model and component-based software engineering. In embedded system development hybird of these models is mostly used. All the models share some generic software engineering activities. Requirements should be captured completely before moving into the design phase. Mostly 3-V model is used for embedded system development. In 3-V model, in intial step the simulation of system is developed and once the simulation is accpected, using COTS plateform design is mapped on the prototype. After the acceptance of prototype the product development phase starts. In embedded system hardware and software are developed concurrently. The development process of embedded systems can be broken down into many parallel activities. These complex development activities influence the software engineering practices. If at any point re-desiging of hardware is required it would effect budgets and timelines of project so the requirement should be specified completely as the rework will require and how it would impact on project.

There is another challenge in embedded system development, is to produce a system with high quality and before the system turn out to be obsolete supply it to the market. To improve the efficiency of the developers and to assure the quality of the software, as discussed in [10], software engineering techniques must be adopted by development team. Stakeholders can define their requirements with the help of requirement engineering processes [11, 12]. It also provides a clear understanding to the supplies, of what is being implemented in embedded system. In requirement engineering process different users with different skills interact, they may include end user, customers, project managers, system engineers, software engineers etc. With the help of RE process, all kind of stakeholders can be facilitated to define baseline requirements, estimate the cost and schedule of project, provide a basis for agreement and a basis for system testing. whole process of RE results in trustworthy artifacts to drive the embedded system development process. The requirements definition process is shown in Table 1:

Table 1. Requirement definition Process activities and artifacts

Phase	Activities	Sub-activity	Artifacts Generated
1. Feasibility study			***Feasibility Report***
2. Requirement Elicitation	2.1 Define requirement Development process		
	2.2 Define vision and scope		Vision and scope Document
	2.3 Identify user classes		
	2.4 Select product champions		
	2.5 Establish focus group		
	2.6 Identify use cases		
	2.7 Identify system events and responses		
	2.8 Hold facilitated elicitation work shop		
	2.9 observe user performing their jobs		
	2.10 Examine problem reports		
	2.11 Reuse requirements		
3. Requirement Analysis	3.1 Draw context Diagram		
	3.2 Create prototypes		
	3.3 Analyze feasibility		
	3.4 Prioritize requirements		
	3.5 Model the requirements	3.5.1 Visually modeling the requirements	DFD, ERD, STD, Dialog maps, use-case diagrams, class diagrams and activity diagrams
	3.6 Create data dictionary		
	3.7 Allocate requirement to subsystems		
	3.8 Apply Quality function deployment		
	---	---	***System Models***
4. Requirement Specification	4.1 Adopt SRS template		
	4.2 Identify source of requirements		
	4.3 Uniquely label each requirement		
	4.4 Record business rule		
	4.5 Specify Quality attributes		
	---	---	***User and System Requirements***
5. Requirement validation	5.1 Inspect Requirement Document		
	5.2 Test the requirements		
	5.3 Define Acceptance Criteria		
	---	---	***Requirement Document***

In Figure 1, the whole process of requirement engineering is elaborated. This Figure elaborate how to switch between phases of while definig requirements. The oval shape represents the process and the artifacts generated from that process is represented by a rectangle.

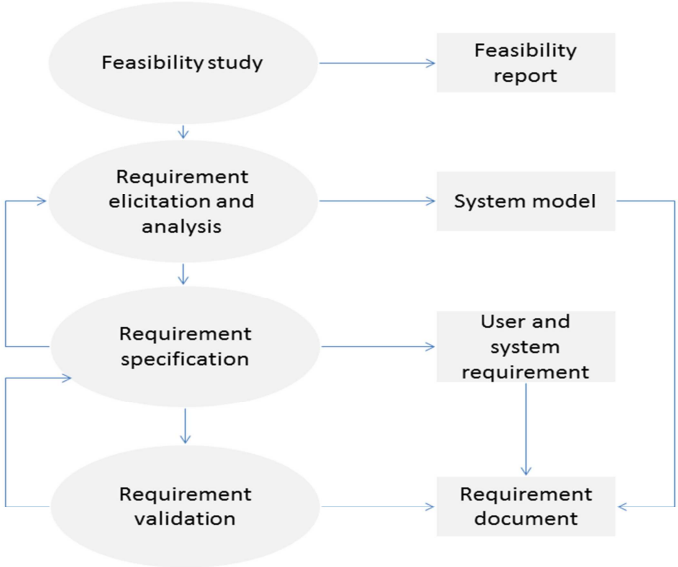

Figure 1. Requirement Engineering process flow from one phase to the next

These requirement engineering process are customarily imposed on pure software engineering projects, and do not address the particularities required by embedded system [2, 8]. The traditional requirement engineering process needs adjustment in order to apply on embedded systems. The main output of the RE process is a requirements specification document. In this research work main focus is on requirements specification process for embedded system. For this purpose, literature review of current requirements engineering practices in embedded system is carried out. Which reveals that difficulty occurs at the initial stages of embedded system development [13, 14, 15, 16]. But there is any not particular process for specifying requirements for embedded systems.

Besides this, further investigation of the relationship between requirement specification and testing of the whole system is performed and a relationship between RSD (requirement specification document) and system testing document is defined.

4 METHODOLOGY

Conceptual-theoretical research method is used which is further sub-divided into two phases:

1. Analysis phase
2. synthesis phas

Analysis phase is based on literature review and study how the previous reaserch on the embedded system requirements specification is carried out. Anaylsis phase is followed by synthesis phase in which new framework is constructed.

Developing a framework for requirement specification which is suitable for all kinds of embedded system is challenging task. Formal methods are used in embedded systems but they considered costly [17]. Requirement specification is necessary to test the system against the what is required and what is developed. Requirement specification mainly relates to the system testing. Requirements should be specified in such manner that it should be easy to trace them and enables the system tester to generate correct test cases. The requirement specification document must have a validation criteria definition to check either system is running as expected. Literature review of IEEE Std. 830-1998 recommendation [18] and CMMI version 1.1[19] is carried out.

Proposed framework has been divided into two phase first one is initial phase and second is main phase.In first phase overall system description is elaborated with the list of stakeholders. As depicted in Table 2. And in second phase which is main phase of Requrirement specification document we categorized requiremnets on the base of logical arctitecure of embedded system shown in Figure 2.

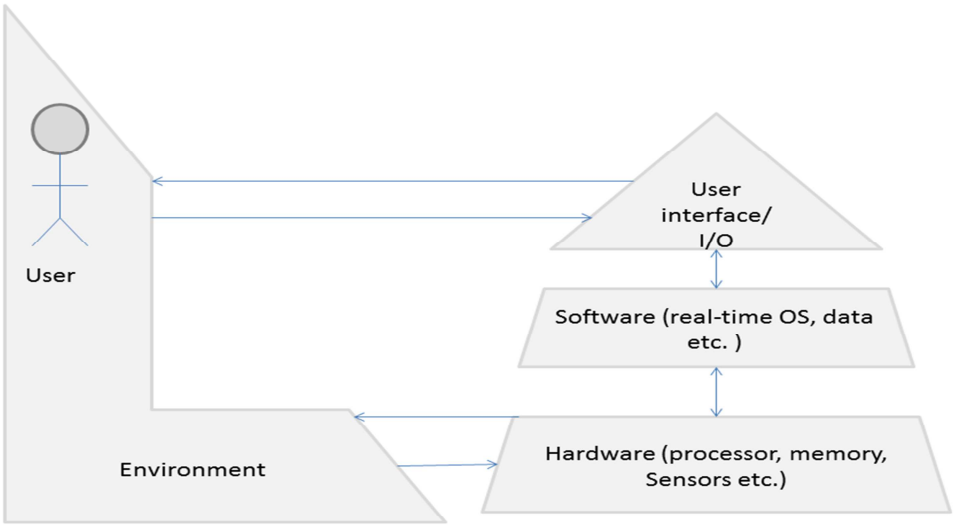

Figure 2. Logical Architecture of Embedded system

According to the logical architecture of the embded system shown in figure 2 requirements for embedded system can be categorized in as follow:

1. Hardware Requirements
2. Software Requirements
3. Interface Requirements
   a. User interface
   b. Hardware interface
   c. Software interface
4. User Requirements
5. Environment Requirements
6. Security Requirements
7. Performance requirements

During the initial phase of framework stakeholders will be able to get the overall idea of system. It will help them to understand overall functionalities of system. Business rules will be established during this phase, legal requirements and standard will be stated, all the system stakeholders will be identified and finally version of document will be documented. In Table 2 activities of initial phase are listed and their descripton is also provided. The artifacts generated after initial phase will help in main phase where the requiremnts will be specified.

Table 2. Initial phase of RSD

Activity	Description
1. Introduction	Overview of the Document
2. Introduction to project	a. Project scope b. Project purpose c. Reference
3. Business requirements	Include all the business needs which should be address by system. Definition of business rules include.
4. Legal requirements	Establish legal requirement check list.
5. Stakeholders	List all the stakeholders which are directly or indirectly interact with system.
6. Version History	List the previous version and current version of document.

Once the initial phase is completed it is necessary to transform these high-level requirements into detailed requirement document. In main phase, a framework for documentation of requirements along with the test-case for each requirement is proposed. This will help both development team and testing team to check either each requirement is traceable or not. All the information about a particular requirement will be manged. Table 3a and 3b illustrate how the requirements will be manged along with integration testing document.

Tables 3a and 3b: Requirement specification

Table 3b

Table 3a

In Table 3a all the description about requirement is specified and in Table 3b all the description about test-case will be recoreded. If test cases are traceable to the requirement specification document than tester can ensure that all requirements are covered in testing. With the help of this framework requirement engineers can specify the requirement for embedded system and also validate those requirements. Embedded system also include hardware requirements which are different from other requirements. The way they are specified and the information which is required to record for hardware is different. So Table 3a cannot be

used to record hardware requirement. To record hardware requirements table 3a will be replaced by Table 4 and Table 3b will remain same.

Table 4: Hardware Requirement specification

1. Requirement number	
2. Requirement name	
3. Requirement description	
4. Hardware description	
5. Indigenous manufacturing	
6. Dependency	
7. Repair	
8. Environmental specification	
9. Special features	
10. Technical characteristics	
11. Technical constraints	
12. Fit criteria	
13. Version history	

Testing consist of sequence of step in which module testing, integration testing and system testing are include. Requirement specification activity is strongly related to the testing of a system. RSD and testing document relation is shown in Figure 3. For this requirements must be written in unambiguiously and should be traceable. Requirement documunets must have validation criteria to which test result must conform. Requirement document must be mangeable if any change occurs it should be recorded. With the help of this framework testers would be able to plan test cases while requirements are recoreded. This will help in reavling flaws at early stages which is inexpensive as compare to reavling flaws after implementation. Module and intergration testing is use to verify the architectural requirements. Module testing and integration are performed with development activities to ensure the accuracy of each module and interface requirements. However purpose of system testing is to validate the overall system functionality. Checking consistency between requiremnts and test-case will also help in to ensure that correct system functionality is being testes. Tracing test-cases back to requirements will ensure that no requirement has been left. This will also help testers to arrange test-case which required extra resources.

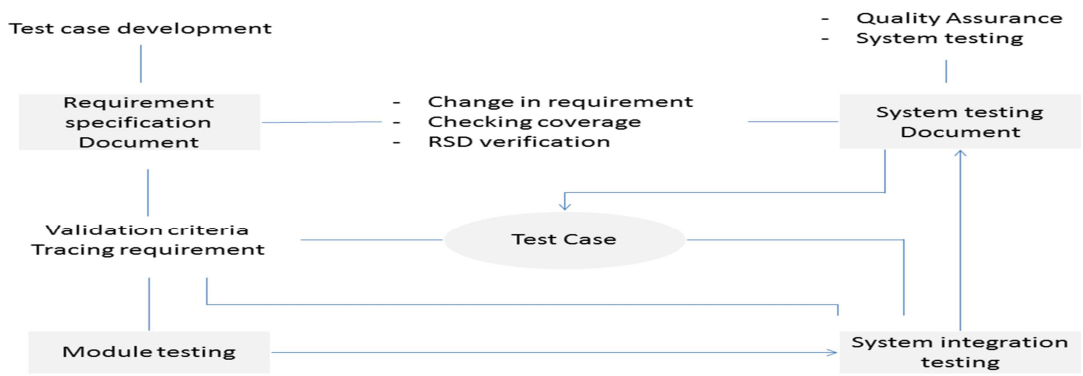

Figure 3. Framework for identifying relationship between RSD and system testing document

6 Conclusion and Future work:

In this research work, a framework for Requirement specification for embedded system along with test-cases has been proposed. Further analysis of the relationship between requirement specification document and system testing documents is done. This framework is for embedded systems where the correct requirement specification is critical to ensure quality, performance, reliability, security and other critical requirements.The quality attributes of systems come from non-functional requirements [20-21]. However, this framework can be used for software projects by ignoring the hardware related requirements.

In future work, implementation of this framework on an embedded system project will be done and the tool by using this framework to manage the requirements will be developed.

REFERENCES

[1] B. Graaf, M. Lormans, and H. Toetenel, "Embedded Software Engineering: the State of the Practice", IEEE Software archive, vol. 20, no. 6, 2003, pp. 61-69

[2] P. Liggesmeyer and M. Trapp, "Trends in Embedded Software Engeneering", IEEE Software Magazine, v. 26, n. 3, 2009, pp. 19-25

[3] E. Nasr, J. Mcdermid J, and G. Bernat, "Eliciting and Specifying Requirements with Use Cases for Embedded Systems", Proceedings of the 7th International Workshop on Object-Oriented Real-Time dependable systems (WORDS), 2002, pp. 350 – 357.

[4] A. Pretschner, M. Broy, I. H. Kruger, and T. Stauner, "Software Engineering for Automotive Systems: A Roadmap Future of Software Engineering", In: International Conference on Software Engineering - Future of Software Engineering (FOSE), 2007, pp. 55 -71

[5] A. Post, I. Menzel, J. Hoenicke, and A. Podelski, "Automotive Behavioral Requirements Expressed in a Specification Pattern System: a Case Study at BOSCH", Requirements Engineering, Springer-Verlag, vol. 17, 2012, pp. 19-33

[6] H. Chae, "The Partitioning Methodology in Hardware/Software CoDesign Using Extreme Programming: Evaluation through the Lego Robot Project", Proceedings of the sixth IEEE International Conference on Computer and information Technology, 2006, p. 187

[7] E. Sikora, B. Tenbergen, and K. Pohl, "Requirements Engineering for Embedded Systems: an Investigation of Industry Needs", Proceedings of the 17th International Working Conference on Requirements Engineering: foundation for software quality (REFSQ'11). SpringerVerlag, Berlin, Heidelberg, 2011, pp. 151-165

[8] M. Broy, "Requirements Engineering for Embedded Systems", Workshop on Formal Design of Safety Critical Embedded Systems (FemSys), Munich, Germany, 1998.

[9] Royce, W. Managing the development of large software systems: Concepts and techniques. Reprinted in proceeding of the Ninth International conference on Software Engineering (Monterey, calif., March 30 - April 2), pp. 328-338, IEEE Pres, Wash., DC., 1987.

10] B. Cheng and J. Atlee, "Research Directions in Requirements Engenneering, Future of software Engeneering". IEEE Future of Software Engineering (FOSE'07), 2007, pp. 285 – 303

[11] H. Hoffmann and F. Lehner, "Requirements Engineering as a Success Factor in Software Projects". IEEE Software, vol. 18, no.4, 2001, pp. 58-66

[12] G. Kotonya and I. Sommerville, Requirements Engineering: Processes and Techniques. John Wiley and Sons, 1998

[13] A. Aurum and C. Wohlin, "Requirements Engineering: Setting the context". In: Aurum C. & Wohlin C. (Eds), Engineering and managing software requirement. Springer. Berlin, Germany, 2005, pp. 1-15

[14] R. Cancian, M. Stemmer, and A. Frohlich, "New Developments in EPOS Tools for Configuring and Generating Embedded Systems", Proceedings of the 12th IEEE International Conference on Emerging Technologies and Factory Automation, Patras, 2007, pp. 776-779

[15] L. Jiang and A. Eberlein, "Selecting Requirements Engineering Techniques Based on Project Attributes - A Case Study". Proceedings of the 14th Annual IEEE International Conference and Workshops on the Engineering of Computer Based Systems, 2007, pp. 269 – 178

[16] F. Vahid and T. Givargis, Embedded System Design: A Unified Hardware/Software Design. John Willey & Sons, 2002

[17] SIDDIQUI, M. A., AKHTER, M. S., & MIAN, N. A. (2014). A Comparative Analysis Of Conventional Software Development Approaches Vs. Formal Methods In Call Distribution Systems. *VAWKUM Transaction on Computer Sciences*, *3*(2), 7-14

[18] IEEE Computer Society Software Engineering Standards Committee, "IEEE Recommended Practice for Software Requirements Specifications". IEEE Std 830-1998

[19] CMMI 2001. Capability Maturity Model R. Integration (CMMISM), Version1.1, Software Engineering Institute. Cresswell, J. 1998. Quality Inquiry and Research Design: Choosing Among Five Traditions.Sage Publications, Thousand Oaks

[20] RASHID, U. (2014). A Methodological Approach: Formal Specification Of Quality Attributes Modeling Approcahes In The Waterfall Process Model.*VAWKUM Transaction on Computer Sciences*, *3*(2), 1-6

[21] ASHRAF, M., & ALI, N. (2013). Impact Agile Project Management: Identification And Analysis Of Problems In Scrum Implementation. *VAWKUM Transaction on Computer Sciences*, *2*(2), 1-8.