Requirements for Point of Care Devices using Use Case Maps

Sivanesan Tulasidas, Josef Hausner, John Terzakis, Fred Love, Stefan Mattern, Intel Corporation, Santa Clara, CA USA,
Xiaoqing Frank Liu, Department of Computer Science, MST, USA, Chris Hudson, Dr.Nada Manivannan, Dr.Ruth Mackay,
Wamadeva Balachandran, Centre for Electronic Systems Research, College of Engineering, Design and Physical Sciences, Brunel University London, Uxbridge, UK

Abstract**—**Point of Care (PoC) testing (diagnosis) is a method for bringing medical laboratories to a patient's home to conduct diagnostic tests so that the patient does not need to go to the doctor or laboratory in person. PoC testing reduces the burden on expensive laboratory setups and provides management of patient care in cost effective manner. The design and development of the PoC device and the associated infrastructure must be done with extreme rigor, as the PoC system meets the definition of a mission critical or safety critical system. Requirements creation and management are the key processes for ensuring that a highly reliable and low defect PoC system is developed since accurate PoC testing-based diagnosis is an essential process improvement for remote patient care management. It is important that the requirements be specified accurately, completely and without any ambiguity so that the PoC device can be designed and developed with minimal errors. This provides physicians a vehicle to diagnose patients with drastically increased reliability. This paper explains how Use Case Maps (UCM), a modeling technique, can help to sufficiently model requirement specifications for a PoC system development. It illustrates PoC functional requirements and security requirements in terms of the UCM representation.

Index Terms**—***Point of Care, Use Case Map, Requirements, Mission Critical Systems, Medical Devices.*

*****

I. Introduction

The requirement format followed in this paper is as follows: FR_<sub-system name>_< a three digit numerical value>. Only the high-level requirements are discussed in this paper. Some of the key requirements are represented with UCM notation [1] [2]. The requirements were created using the EARS (Easy Approach to Requirement Syntax) [3] syntax. EARS contains known patterns for particular types of functional requirements. Table 1 shows the patterns used in constructing the requirements.

Pattern Name	Pattern
Ubiquitous	The < System name> shall < system response>
Event-Driven	WHEN <trigger> <optional precondition> the <system name > shall <system response>
Unwanted Behavior	IF <unwanted condition or event>, THEN the <system name > shall < system response>
State-Driven	WHILE <system state>, the <system name> shall <system response>
Optional Feature	WHERE <feature is included>, the <system name> shall < system response>
Complex	Combination of all the above

Table 1: EARS pattern [15]

II. PoC System Description

The PoC system is being developed at the Brunel DOC LAB [4]. It consists of modular subsystems as described in reference [5]. The primary building blocks of the system are as follows:

- P-Node: the PoC device in the patient's home
- G-Node: the gateway device in the patient's home
- P-Cloud: the private cloud over which patient data is transmitted and can be accessed by physicians

These building blocks are shown in Figure 1.

III. P-Node Requirements

The P-Node is the representation of the PoC testing device. The G-Node represents the gateway entity, which can be a smartphone or a laptop with internet access. Patient interacts with the P-Node for diagnostic testing via the G-Node. These devices are used to send measurement data to database server, as well as control the P-Node. The P-cloud represents the secure private data cloud [5].

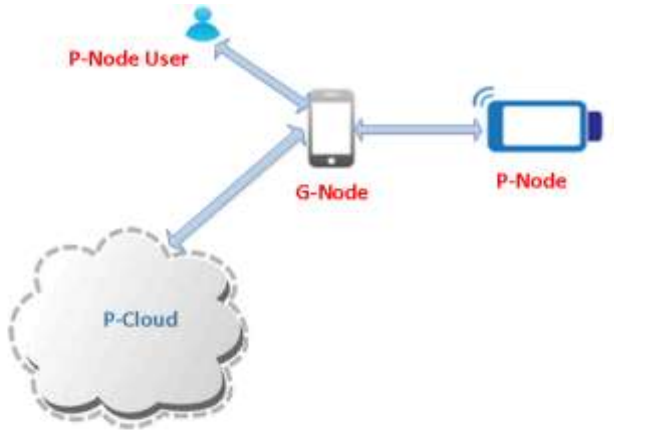

Figure 1: Basic building blocks

A. Requirements for P-Node operation

FR_PNODE_001:
The P-Node shall have two modes of operation: user control mode and reporting mode.

FR_PNODE_002:
While in the user control mode, the P-Node shall receive commands from the G-Node for initiating the PoC testing.

FR_PNODE_003:

When completed test results are ready for transmitting to the G-Node, the P-Node shall switch to reporting mode.
FR_PNODE_004:
While in reporting mode, the P-Node shall report the test data to the P-Node user (via the G-Node) and to the P-Cloud.

B. UCM representation of P-Node operation requirements

Error! Reference source not found. shows the UCM representations of the requirements FR_PNODE_001, FR_PNODE_002, FR_PNODE_003 and FR_PNODE_004. Path 1 conveys a message to downstream implementation team that the P-Node user shall abe able to control the P-Node via the G-Node. Path 1 shows the execution path for this use case (that the user can control the P-Node), which is the user control mode of the PoC.

Path 2 represents the reporting mode of the P-Node to the P-Cloud. Path 2 conveys the message to the development team that the data needs to be sent to the P-Cloud by some communication means. Note that Path 2 does not go through the G-Node, implying that the P-Node needs to have a communication subsystem.

Path 3 conveys the message that during the reporting mode, the P-Node also needs to send data to the user via the G-Node. The 'AND' UCM artifact [6] in **Error! Reference source not found.**(the vertical line with 2 horizontal lines on one side and one horizontal line on the other side) serves as a way of enforcing the "and condition" that the report needs to be sent to both the user and P-Cloud.

Path 4 conveys the intent to the developer community or the testing community that the P-Node must send the data also via the G-Node to the P-Cloud for redundancy.

C. Requirements for P-Node construction and system interconnetion

FR_PNODE_005:
The PoC system shall be designed as a collection of modular based subsystems.

Note: Because this to encourage design of mission critical system such as PoC as a loosely coupled system [7].

FR_PNODE_006:
The PoC subsystems shall interconnect using industry standard interface technology.

Note: the interconnect technology may include the I2C [8] interface but is not limited to it.

FR_PNODE_07:
If there is a fault detected in an isolated systems module, and then the PoC shall contain that fault within the individual module.

FR_PNODE_008:
The PoC modularized system shall meet the mandatory system design and development process as outlined in the international standard for medical device software and software cycle processes standard IEC 62304 [7].

FR_PNODE_009:
The PoC system shall interconnect with a third party system.

Note: this interconnection can be via an industry standard interface like USB or Wi-Fi.

D. Requirements for P-Node fail safe mechnism

FR_PNODE_010:
The PoC system shall be implemented using multiple, independent hardware modules.

Note: The rationale for this requirement is to have a system that does not have an SPOF (single point of failure).

E. Requirements for PoC data storage

FR_PNODE_011:
When the PoC completes a diagnostic test, and the communication system is disabled, the PoC shall store the test data in local non-volatile memory.

Note: Local non-volatile memory can include an SD card or a Flash drive.

FR_PNODE_012:
If a communication link failure occurs, then the PoC shall identify the source of the failure.

FR_PNODE_013:
The PoC shall include an agent to monitor communication link failures.

FR_PNODE_014
The P-Node shall encrypt data transmitted to the P-Cloud using SSL revision 3.0.

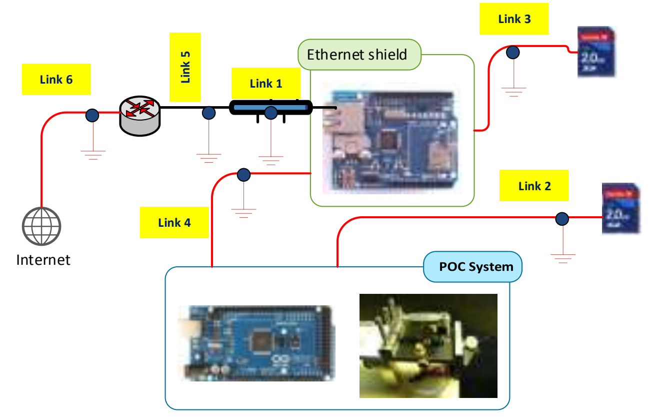

Figure 2: Identification of failure pointes

The links (Link 1,2,3,4,5 and 61-6) that are shown in Figure 3 are related to the local data storage when the SD card and Ethernet connection are used in the PoC system. The failure points (indicated by ground signals as per the UCM notation [8]) are marked in the communication links. The ground signal is the UCM representation for showing failure points in addition to the text-based requirements (FR_PNODE_011 and FR_PNODE_012). Both the textual requirements and the UCM representation diagram will provide an unambiguous way of conveying the intent to the development and testing organizations. Also, this modeling paves the way for applying the failure mode analysis methodologies applicable to medical devices [9][10][11].

IV. G-Node Requirements

The G-Node represents the gateway entity that has the responsibility for controlling the PoC device and collecting the test data from the PoC device. The G-Node can be a smartphone or a computing device such as a PC.

FR_GNODE_001:
The G-Node shall be subscribed to the PoC test completion events for receiving test data from the PoC device.

FR_GNODE_002:
The G-Node shall encrypt data transmitted to the P-Node using SSL revision 3.0.

(Note: a corresponding P-Node requirement has been added under P-Node requirments: FR_PNODE_014)

FR_GNODE_003:
The PoC shall utilize each of the following connectivity solutions for communication between the G-Node and the P-Node:

- Cellular radio (either 2G, 3G or 4G)
- Wi-Fi
- USB

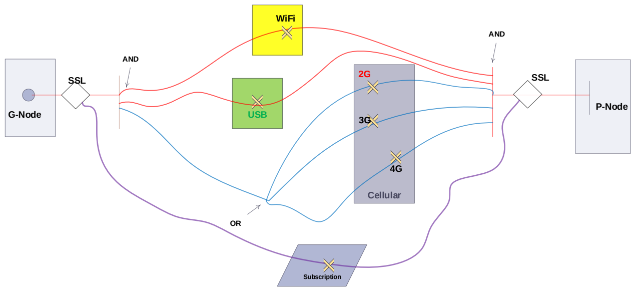

Figure 3: UCM for representing G-Node requirements

Figure 4 represents the G-Node requirements FR_GNODE_001, FR_GNODE_002 and FR_GNODE_003 using UCM notation. The SSL requirement is shown as a static stub (UCM terminology for special functionality) at both ends. The static stub informs the development team that SSL is needed in the G-Node as well as in the P-Node. The use of the 'AND' junction conveys the intent that Wi-Fi, USB, and the cellular connectivity are needed simultaneously as described in the requirement FR_GNODE_003. The 'OR' junction indicates that only one of the technologies among the 2G or 3G or 4G are active at a time. Subscription process block shows the subscription of test completion event by the G-Node.

The UCM map shown in Figure 4 shows another important consideration that the system design must look at the coexistence challenges [12] between the 4G and Wi-Fi. It is difficult to miss any implementation of the requirements through the UCM representation shown in the diagram.

V. P-Cloud Requirements

The data collected from the PoC device is stored in a non-volatile memory device, such as SD card, first. This data is transmitted to Cloud storage (P-Cloud) via the G-Node as shown in Figure 1. The following section lists the requirements pertaining to the P-Cloud.
The P-Cloud provides a way of sharing the test data among various healthcare organizations. The purpose of the P-Cloud can be divided into four domains (PoC test data, PoC provisioning data, PoC operational data and PoC test data for data mining)([5] section 6). The p-cloud is the storage for the raw data and the transformed data from the P-Node. The transformed data is needed for data mining analytics as an aid to diagnosis. The P-Node to P-Cloud connectivity is implemented using M2M communication. Physicians use the transformed data for diagnosis. Therefore, the P-Node needs two separate access gateways ([5] sections 6.1 and 6.2).

Provisioning data from PoC devices and deployment data of the PoC installations need to use the P-Cloud. The access path for this kind of use can be managed by a separate access gateway [5].

PoC device operational data also needs to use the P-Cloud, and it requires another access gateway to the P-Cloud ([5] section 6.3).

Based on the above concepts, the following requirements are formulated.

FR_PCLOUD_001:
The P-Node shall provide secure access based on SSL 3.0 encryption for storing the PoC device provisioning data and the PoC system deployment data.

Note: The connectivity between the P-Node and the P-Cloud shall be classified as M2M communication.

FR_PCLOUD_002:
The P-Cloud shall provide secure access based on SSL 3.0 encryption for storing the PoC devices operational data.

FR_PCLOUD_003:
The P-Node shall provide secure access based on SSL 3.0 encryption for a physician for diagnosing patient data.

FR_PCLOUD_004:

The P-Cloud shall encrypt data transmitted to the P-Node using SSL revision 3.0.

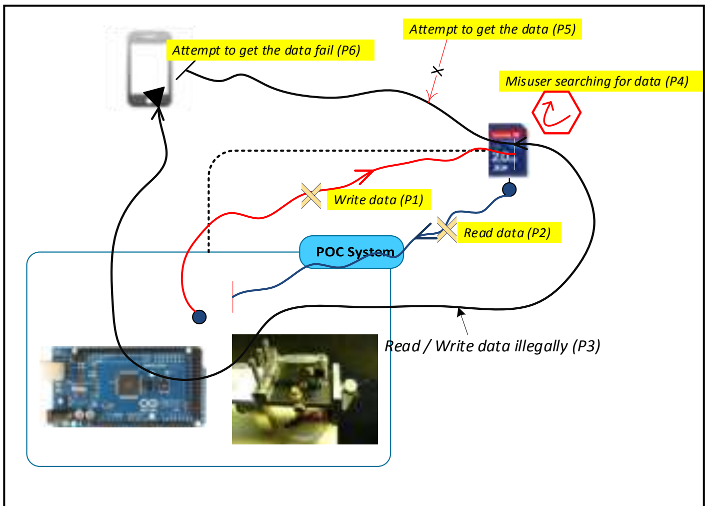
Attempt to get the data (P5)
Attempt to get the data fail (P6)
Misuser searching for data (P4)
Write data (P1)
Read data (P2)
POC System
Read / Write data illegally (P3)

Figure 5: MUCM Example

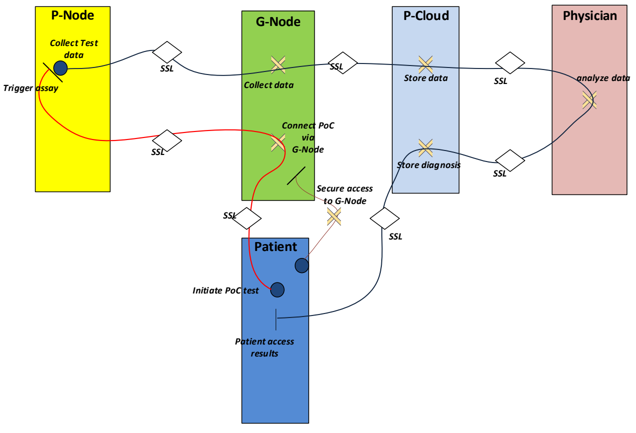

Figure 4: P-Cloud requirements

Figure 5 shows the UCM representation of the PoC system indicating all the P-Clod requirements stated. A similar approach is shown in reference [13]. The responsibilities (or the functionalities) required within each domain are displayed as 'X'. The diamond indicates the SSL connectivity between all the players in the data collection to data storage.

VI. PoC Security Requirements

In the following section, security requirements are discussed using the technique called MUCM (missed use case maps) [14]. The security with respect to the data is elaborated.

FR_PoC_DATA_001:
Data stored in SD card shall be protected for authorized read and write activity.

Figure 6 shows the UCM and the MUCM paths for accessing data from and to the SD card. Since the initial data storage is the SD card, the data on the SD card must be protected. The path P1 in the figure is a normal operation for writing data to the SD card during the test. Path P2 shows, reading data from the SD card. These are the UCM representation for R/W data from/to with respect to the SD card.

The path P3 shows and an intruder attempts to modify or read data on the SD card illegally, by connecting to the PoC System from an unauthorized smartphone. The path P4 indicates that the misuser (unautherized user) searches the SD card for the data. P5 indicates that misuser's attempts to read the data. At the end, attempting to access the data fails because of the close-loop nature of the system and the security measures that are in place (which is beyond the scope of this paper).

The above section explains section the use of MUCM for a security related use case.

Conclusions

The requirements defects are often the most expensive defects because requirements form the basis of so many other work products. Correcting defects earlier in the product development is very cost effective than attempt to correct them during the testing phase. By the use of the modeling techniques such as the UCM and the MUCM, requirements can be explained more clearly in addition to textual descriptions. The paper shows a novel way of using the Use Case Maps in requirements creation. The other type of modeling technique, the MUCM is used to explain a security requirement.

Because of the visual representation of the UCM, process of adopting the UCM to express complex requirements will certainly help to visulalize requirements in addition to text contents. It will also helo to manage the project in all stages of product development; requirement elicitation, analysis and validation, creation of product specifications, verification and management. An ambiguous requirement can be expressed in terms of the UCM visual representation. It is safe to state that if a requirement is difficult to express in the UCM, then the implementation the requirement will be difficult for the downstream development teams.

The UCM will help to narrow down gaps between the design and requirement because of the visual representation. Using UCM to express requirements will provide a clear traceability between test cases to the requirements. Hence, the miscommunication between testing and design teams can be minimized. Also a new development project can be managed with low risk, because the ambiguity in requirements are less with the use of UCM.

A way of measuring the success of using the UCM for expressing the requirements can be done by gathering defects containment rates and defects finding rates.

ACKNOWLEDGMENT

This paper would not have been completed without help and support I received from a number of people at Intel Corporation, University of Brunel and Missouri University of Technology and Science (MST). I would like to thank the authors for their contributions and review feedbacks. I would like to thank the Intel team for their prompt feedback. I would like to thank the research staff from Brunel University and the MST for the fruitful discussions that helped to provide enhancements to the original text.

REFERENCES

[1] D. Amyot, “Use Case Maps Quick Tutorial,” *See http//www. usecasemaps. org/pub/UCMtutorial/UCMtutorial. pdf,* no. September, 1999.

[2] “Bridging the Requirements/Design Gap in Dynamic Systems with Use Case Maps (UCMs).” [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.29.1221&rep=rep1&type=pdf. [Accessed: 01-Mar-2015].

[3] A. Mavin, P. Wilkinson, A. Harwood, and M. Novak, “Easy Approach to Requirements Syntax (EARS),” in *2009 17th IEEE International Requirements Engineering Conference*, 2009, pp. 317–322.

[4] “Brunel DOC LAB.” [Online]. Available: http://bruneldoclab.com/. [Accessed: 09-Feb-2015].

[5] S. Tulasidas, R. Mackay, P. Craw, C. Hudson, V. Gkatzidou, and W. Balachandran, “Process of Designing Robust, Dependable, Safe and Secure Software for Medical Devices: Point of Care Testing Device as a Case Study,” *J. Softw. Eng. Appl.*, vol. 06, no. 09, pp. 1–13, Aug. 2013.

[6] “jUCMNav Tutorial 1: Creating a simple path, components, stubs and plug-in maps.” [Online]. Available: http://www.youtube.com/watch?v=kuXvxmcfzh8. [Accessed: 01-Mar-2015].

[7] “IEC 62304 Ed. 1.0 b:2006 Medical device software - Software life cycle processes.” [Online]. Available: http://webstore.ansi.org/RecordDetail.aspx?sku=IEC 62304 Ed. 1.0 b:2006&source=google&adgroup=iec&gclid=CjwKEAiA68WnBRCJxZr5qoaL3iMSJAAXIrr3_muvjitFt8NGj6Lk8EEFigpeQCZLSfQVL1IoI44s5hoCShfw_wcB. [Accessed: 01-Mar-2015].

[8] D. Amyot and G. Mussbacher, “Minitutorial - UCM Minitutorial - UCM Table of Contents to Use Case Maps (UCMs) and the UCM Notation Minitutorial - UCM Minitutorial - UCM Use Case Maps Web Page,” 2001.

[9] *Procedures for Performing a Failure Mode, Effects and Criticality Analysis*. U.S. Department of Defense, 1949.

[10] “The Use and Misuse of FMEA in Risk Analysis | MDDI Medical Device and Diagnostic Industry News Products and Suppliers.” [Online]. Available: http://www.mddionline.com/article/use-and-misuse-fmea-risk-analysis. [Accessed: 01-Mar-2015].

[11] *Standard Practice for System Safety*. U.S. Department of Defense, 1998.

[12] Z. Hu, R. Susitaival, Z. Chen, I.-K. Fu, P. Dayal, and S. Baghel, “Interference avoidance for in-device coexistence in 3GPP LTE-advanced: challenges and solutions,” *IEEE Commun. Mag.*, vol. 50, no. 11, pp. 60–67, Nov. 2012.

[13] X. Liu, L. Peyton, and C. Kuziemsky, “A requirement engineering framework for electronic data sharing of health care data between organizations,” 2009.

[14] P. Karpati, G. Sindre, and A. L. Opdahl, “Visualizing cyber attacks with misuse case maps,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6182 LNCS, no. 7491, pp. 262–275, 2010.

[15] “ A Pragmatic Guide to Best Practices .” [Online]. Available: http://www.uploads.pnsqc.org/2011/slides/Simmons_21st-Century_Requirements_slides.pdf. [Accessed: 28-Feb-2015].