

Software Product Lines for Mobile Patient Monitoring Systems Using FODA a Grammar

Paliwal G^{1*} and Bunglowala A²

¹Department of Computer Engineering, RC Patel Institute of Technology (RCPIT), Shirpur 425405, Dhule (MS), India

²Department of ENTC Engineering, Narsee Monjee Institute of Management Studies, Shirpur 425405, Dhule (MS), India

*Corresponding author: Paliwal G, Department of Computer Engineering, R C Patel Institute of Technology (RCPIT), Shirpur 425405, Dist Dhule (MS), India, Tel: 9970206604; E-mail: gvpaliwal@gmail.com.

Received: February 8, 2017; Accepted: March 02, 2017; Published: March 07, 2017

Abstract

A common Architectural design, for a diverse set of components is not a trivial task. It involves preparation for mass customization, focusing first on what is common to all products, and then in what is different. Components must be engendered for being reused by all, or most of the possible configurations. These components can be developed from scratch or derived from other platforms. This flexible design sanctions to reuse these components with different configurations for a particular solution to provide a mass customized set of well-defined products. Product Line (PL) engineering aims to fortify engenderment of wide range of customized products with flexible design. Variability design is about incorporating components that represents the range of possible configurations for a product in the PL. This variability, is defined during the domain engineering process. When we utilize the term variability we are referencing to the capability that something has to change in time, which makes it categorically appropriate for the domain of Mobile Patient Monitoring Systems (MPMs) where we require a wide range of mass customized product for different group of chronically ill patients. In this paper we have derived the Product Lines for the domain of MPMs utilizing the fodaA grammar.

Keywords: *Mobile patient monitoring system; Software product lines; Domain modeling; FodaA grammar; Context analysis; Feature analysis*

Introduction

Information and communication technology (ICT) is one of the fastest growing technologies in the world. Mobile Health (mHealth) basically denotes adoption of these (smartphones and other hand held devices) contrivances to improve to the patient monitoring services [1]. It is primarily adopted to provide efficient, economical and easily accessible health care solutions. Adoption of ICT for MPMs will benefit millions of chronically ill patient and it will provide an effective substitute for traditional patient monitoring practices.

Numerous MPMs have been designed earlier that offers a diverse range of functional and non-functional features. In this paper, we have provided PL for MPMs that basically captures the commonalities and variability's of these systems. PL is a software design approach particularly developed for progressive domains to achieve a higher degree of reuse and

Citation: Paliwal G, Bunglowala A. Software Product Lines for Mobile Patient Monitoring Systems Using FoDA a Grammar. *Biochem Ind J.* 2017;11[2]:112.

© 2017 Trade Science Inc.

customization within the budget and time limits. The elementary of a PL is an amassment of features, termed as the scope and an accumulation of reusable components known as the core assets [2].

The elementary unit of a production line is called feature. A feature can be a particular module, the utilization of a categorical technology, and in general any reusable functional component. The relationships between features in a PL builds product configurations and relationships and that features set is called feature model. The feature analysis output can be further used to explore reusable components for PL development.

Generic Architecture for MPMs

The generic architecture for MPMs [3] comprises systematic development processes for various MPM product variants. From the product perspective, the generic architecture for MPMs has two components, namely, Body area network (BAN) and back end system (BESys) as shown in FIG. 1. From product perspective BAN, can be defined as a grid of different communication devices worn on or around the body for acquiring the physiological data. The data is then transferred to mobile health service providers using different communication devices. The BESys comprises of a mobile base unit (MBU) which initially processes the data and then transfer it to back-end server. Data is then further processed by the back-end system using different health care application.

From the process viewpoint, BAN acquires the physiological data actively and passively by interacting with the patient. Data is then bundled for delivery to the caregivers and third parties applications through different communication Medias. Healthcare workers and/or algorithms interpret and review the data and evaluate if there are any areas for concern. From process perspective we can define the MPM architecture in five steps:

- I. Data acquisition
- II. Initial Data Processing and Transmission
- III. Bio signal Interpretation and Processing
- IV. Notification and
- V. Intervention

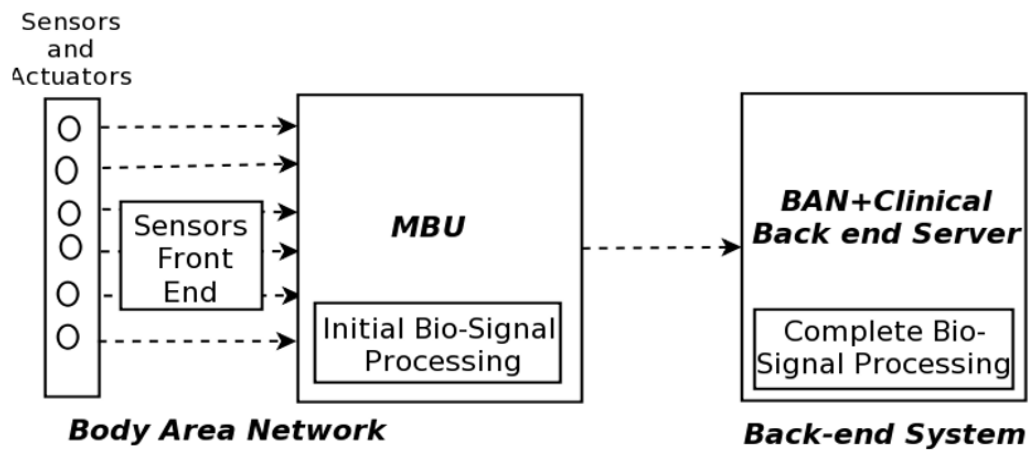


FIG. 1. A Generic architecture for mobile patient monitoring system [3].

This separation amongst process and product perception helps system designers to independently focus on functional and structural elements of MPMs facilitating systematic development of these systems.

Feature Analysis for Mobile Patient Monitoring Systems

Feature Oriented Domain Analysis (FODA) is a method of requirements analysis which defines both the products and the processes of domain analysis. In FODA, a feature is a prominent or distinctive user-visible aspect, a quality, or a characteristic which can be recognized as mandatory, optional or alternative property for a system [4,5,20].

Feature models were first introduced as part of the FODA method by Kang et al. [4]. FODA is a methodology originated from the study of different approaches of domain analysis. It gives a view of the requirements and architecture aspects over the assets to be built [6]. This methodology focuses on the identification of product features in a well-defined domain [5]. Moreover, the commonalities and variabilities of these features must be specified. Also, this model allows to graphically represent features and their relationships, in order to define products in a PL. The graphical structure of a FODA model is presented by a FODA Diagram. A FODA Diagram is fundamentally a graph, which is an intuitive and easy way to represent relevant information about features [7]. It represent PL features as nodes and arcs represents relationships among these features.

FIG. 2 shows the feature diagram of MPMs in three levels. The level 1 of the diagram shows the components and functional features present in the system. The second level of the diagram comprises of features that are expected to be delivered by respective components and selected features has been further extended in the third level. The optional features are represented by a white circle at the end of the connector and the mandatory features are represented by a black circle at the end of the connector whereas the alternative features are represented by an arc on the connectors.

The fundamental features required for MPMs are biosignal delivery and biosignal processing. Since these features are not sufficient for proactive usability we have included certain features which are not essential for the system but we have treated them as necessary features to improve the performance and accessibility of these systems. Some of these selected features are elaborated below:

- **Genericity:** The system should be available to a wide variety of patients, community or group of persons. It should be modified according to the patient monitoring requirements. We have introduced generosity with the help of generic architecture [3,6]. By following these architecture we can modify system functionalities and sensors according to the needs.
- **Intelligence:** System should provide assistance to the physician/Doctor/Caregiver in understanding the physiological data, decision making and mechanizing the monitoring process through data management systems [7-9] and artificial intelligence. The system should be capable of data extraction, data interpretation and effectively encapsulating the actual context aware information to ensure the correct data delivery every time. Clustering algorithm technics for predicting current health status [10] of a patient have been proposed to support these type of functionalities.
- **Unique patient identification:** Patient should be globally uniquely identified with a patient ID. The most promising choice for the feature is RFID [11] tags; in addition to unique identity to a patient it also provide solution for patient tracking and security issues.
- **Availability and Speed:** The System should be capable of processing tones of data in real time and should be available 24*7. Cloud servers are the most viable options due to its theoretically unlimited processing power and it can also ensure the availability of the system.

- Data delivery: For these type of systems accurate biosignal data delivery is very important since clinical decision making can be affected with a minor amendment. Looking at the Goodput requirements and availability 3G/4G connections are most viable options to fulfill the requirements.
- Multi-touch support: Medical images needs to be interpreted in more than one dimension for clinical decision making called as multimodal analysis. For multimodal analysis devices must have multi touch support feature.
- Biosensors: Biosignals are acquired using different type of biosensors. The biosensors deployed in MPMs should be accurate, easily wearable, power efficient and, small in size.
- Medicine infusion capability: The ability of medicine infusion is very important for cardiac, diabetic [14], and asthmatic patients. It can improve the emergency response time and quality of scheduled medications by mechanically triggered infusion.

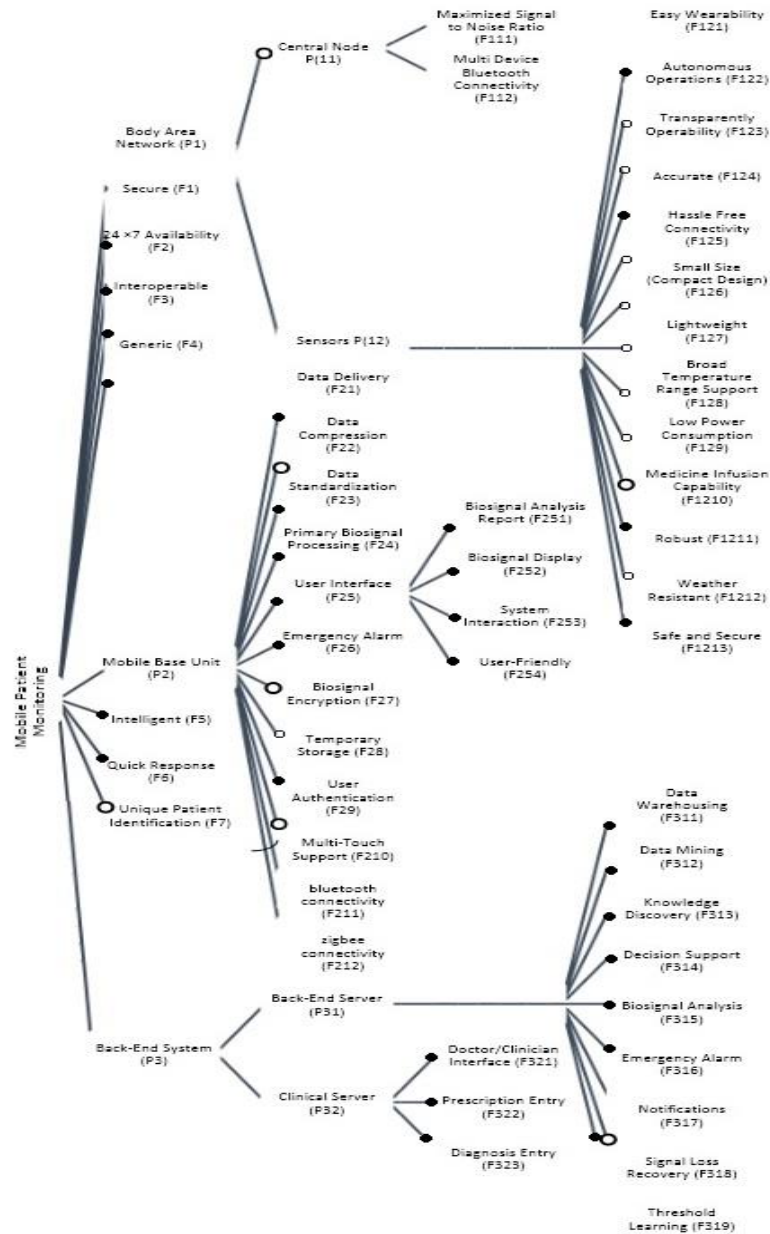


FIG. 2. Feature diagram for MPMs.

Translation rules for FODA diagram to FODA language


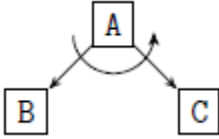
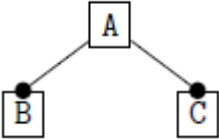
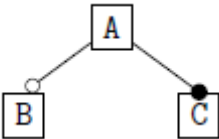
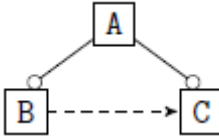
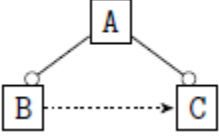
Using the classical Process Algebras notations, we can translate any FODA diagram into a FODA a formal language. This approach was defined by Carlos Camacho [15-17] using Process Algebras as basis to build a formal representation of FODA graphics, which in the first place, lacks of a formal representation. Let us note, that FODA methodology was originated from a study of different approaches of domain analysis. It provides an assessment of the requirements and architectural aspects of the assets to be constructed [18-22]. In particular, FODA can represent different relations between features belonging to a PL [23,24]. These relations are: optional, mandatory, choose 1, and parallel. It also represents requires and excludes constraints. This section describes the steps of FODA formalization methodology consists of representation and translation from FODA diagrams to fodaA terms in TABLE 1 and translation from FODA Diagrams into fodaA grammar in TABLE 2.

TABLE 1. FODA diagram to FodaA term [15].

Type	FODA Diagram	fodaA term
Feature		$F; \checkmark$
Mandatory		$F; (F1; \checkmark \wedge \triangle P \dots \wedge Fn; \checkmark \triangle Pn)$
Optional		$F; (\overline{F1}; \checkmark \triangle P \wedge \dots \wedge \overline{Fn}; \checkmark \triangle Pn)$
Choose 1		$F; (F1; \checkmark \triangle P \dots Fn; \checkmark \triangle Pn)$
Requires		$A \Rightarrow B \text{ in } \triangle P$
Excludes		$A \not\Rightarrow B \text{ in } \triangle P$

TABLE 2. FODA diagram to fodaA grammar [15].

Sr. No.	FODA Diagram	fodaA grammar
1		$\sim A; \overline{B}; \checkmark$

2		$\sim A; B; \checkmark$
3		$\sim A; (B; \checkmark \wedge C; \checkmark)$
4		$\sim A; (B; \checkmark \wedge C; \checkmark)$
5		$\sim A; (\bar{B}; \checkmark \wedge C; \checkmark)$
6		$\sim B \neq C \text{ in } A; (\bar{B}; \checkmark \wedge \bar{C}; \checkmark)$
7		$\sim B \Rightarrow C \text{ in } A; (\bar{B}; \checkmark \wedge \bar{C}; \checkmark)$

Deriving the product lines from feature mode

In Product Line software development methodology, we consider all phases of production life-cycle of the system development process [21]. PLs are established with a purpose to describe experience of system builders in the areas of data management, requirements, human-computer interactions, procedure, testing, design procedures and validation of a product [19,22].

In this section, we describe PLs for the MPMs using fodaA grammar. Here we are demonstrating product variability observed in MPMs using Extended Modal Transition Systems [15,16].

PL 1. $\sim MPM; (\sim P1; \wedge F1; \checkmark \wedge F2; \checkmark \wedge F3; \checkmark \wedge F4; \checkmark \wedge \sim P2; \wedge F5; \checkmark \wedge F6; \checkmark \wedge F7; \checkmark \wedge \sim P3;)$

PL 2. $\sim P1; (\bar{P11}; \wedge P12;)$

PL 3. $\sim P11; (F111; \checkmark \wedge \bar{F112}; \checkmark)$

PL 4. $\sim P12; (F121; \checkmark \wedge \bar{F122}; \checkmark \wedge \bar{F123}; \checkmark \wedge F124; \checkmark \wedge \bar{F125}; \checkmark \wedge \bar{F126}; \checkmark \wedge \bar{F127}; \checkmark \wedge \bar{F128}; \checkmark \wedge \bar{F129}; \checkmark \wedge \bar{F1210}; \checkmark \wedge F1211; \checkmark \wedge \bar{F1212}; \checkmark \wedge F1213; \checkmark)$

PL 5. $\sim P2; (F21; \checkmark \wedge \bar{F22}; \checkmark \wedge F23; \checkmark \wedge F24; \checkmark \wedge \sim F25; \checkmark \wedge F26; \checkmark \wedge \bar{F27}; \checkmark \wedge \bar{F28}; \checkmark \wedge F29; \checkmark \wedge \bar{F210}; \checkmark \wedge F211; \checkmark \wedge F212; \checkmark)$

PL 6. $\sim F25; (F251; \checkmark \wedge F252 \checkmark \wedge F253; \checkmark \wedge F254; \checkmark)$

PL 7. $\sim P3; (P31; \wedge P32;)$

PL 8. $\sim P31; (F311; \checkmark \wedge F312; \checkmark \wedge F313; \checkmark \wedge F314; \checkmark \wedge F315; \checkmark \wedge F316; \checkmark \wedge \overline{F317}; \checkmark \wedge \overline{F318}; \checkmark \wedge F319; \checkmark)$

PL 9. $\sim P32; (F321; \checkmark \wedge F322; \checkmark \wedge F323; \checkmark)$

In the product lines from PL1 to PL9, feature are identified using a feature identifiers. Mandatory and optional feature are represented by A; $\}$ and $\overline{A}; \}$ respectively. Conjunction and choose-one operations are represented by \wedge and Υ respectively. In the design process of different variant of MPMs the variability's exhibited through these PLs will play an important role.

Conclusion and Future Directions

The techniques developed in the field of mobile communication can be effectively used for MPMs. However, development of these systems carries the challenge of varied requirements. For successful development of mobile patient monitoring systems and to overcome the various challenges we suggest adoption of product line approach for the development of these systems. Millions of chronically ill patients will get better healthcare services at reduced cost with the successful development and deployment of MPM systems. The MPMs have both, potential and capability to change the face of customary patient care but some issues like patient privacy and system security still requires the attention. To accomplish a better response from the end user user-friendliness is another area of MPMs which should be improved. We have considered almost every technical issue but non-technical issues such as economical, managerial, and legal, that we have not included in this paper should also be taken into account for end user deployment of these systems.

References

1. Whittaker R. Issues in mHealth: Findings from key informant interviews. *J med Internet res.* 2012;14(5):e129.
2. Mohalik S, Ramesh S, Millo JV, Krishna SN, Narwane GK. Tracing SPLs precisely and efficiently. In *Proceedings of the 16th International Software Product Line Conference-Volume 1 2012* (pp. 186-195). ACM.
3. Pawar P, Jones V, Van Beijnum BJ, et al. A framework for the comparison of mobile patient monitoring systems. *J Biomed Inform.* 2012;45(3):544-56.
4. Kang KC, Cohen SG, Hess JA, et al. Feature-oriented domain analysis (FODA) feasibility study. Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst; 1990 Nov.
5. Krut R, Zalman N. Domain analysis workshop report for the automated prompt and response system domain. Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst; 1996 May.
6. Jones V, Gay V, Leijdekkers P. Body sensor networks for mobile health monitoring: Experience in Europe and Australia. In *Digital Society, 2010. ICDS'10. Fourth International Conference on 2010 Feb 10* (pp. 204-209). IEEE.
7. Paliwal G, Kiwelekar AW. A comparison of mobile patient monitoring systems. In *International Conference on Health Information Science 2013 Mar 25* (pp. 198-209). Springer Berlin Heidelberg.
8. Fotiadis DI, Likas A, Protopappas V. Intelligent patient monitoring. *Wiley Encyclopedia of Biomedical Engineering.* 2006 Apr.
9. O'Donoghue J, Herbert J. Data management system: A context aware architecture for pervasive patient monitoring. In *Proceedings of the 3rd International Conference on Smart Homes and Health Telematic (ICOST 2005) 2005 Jul* (pp. 159-166).

10. Patil D, Wadhai D, Vijay M, et al. An adaptive parameter free data mining approach for healthcare application. arXiv preprint arXiv:1211.1788. 2012 Nov 8.
11. Chen M, Gonzalez S, Leung V, et al. A 2G-RFID-based e-healthcare system. *IEEE Wireless Communications*. 2010 Feb;17(1).
12. Bourouis A, Feham M, Bouchachia A. Ubiquitous mobile health monitoring system for elderly (UMHMSE). arXiv preprint arXiv:1107.3695. 2011 Jul 19.
13. Pawar P. Context-aware vertical handover mechanisms for mobile patient monitoring (Doctoral dissertation, Twente University Press).
14. Ichimori S, Nishida K, Shimoda S, et al. Development of a highly responsive needle-type glucose sensor using polyimide for a wearable artificial endocrine pancreas. *J Artif Organs*. 2006;9(2):105-13.
15. González CC. Software product lines using FODA: A formal approach.
16. Fischbein D, Uchitel S, Braberman V. A foundation for behavioural conformance in software product line architectures. In *Proceedings of the ISSTA 2006 workshop on role of software architecture for testing and analysis*. 2006; (pp. 39-48). ACM.
17. Paliwal G, Kiwelekar AW. A Product line architecture for mobile patient monitoring system. In *Mobile Health 2015* (pp. 489-511). Springer International Publishing.
18. Lee J, Muthig D, Naab M. A feature-oriented approach for developing reusable product line assets of service-based systems. *J Syst Software*. 2010 Jul 31;83(7):1123-36.
19. Perry DE. Generic architecture descriptions for product lines. In *International Workshop on Architectural Reasoning for Embedded Systems 1998* Feb 26 (pp. 51-56). Springer Berlin Heidelberg.
20. Halim SA, Jawawi DN, Ibrahim N, et al. An approach for representing domain requirements and domain architecture in software product line. *Software Product Line-Advanced Topic*. 2012:23.
21. Ahmed F, Capretz LF. The software product line architecture: An empirical investigation of key process activities. *Inform Software Tech*. 2008;50(11):1098-113.
22. Zhu C, Lee Y, Zhao W, et al. A feature oriented approach to mapping from domain requirements to product line architecture. In *software engineering research and practice*. 2006 Jun (pp. 219-225).
23. Klatt B, Krogmann K. Model-driven product consolidation into software product lines. In *workshop on model-driven and model-based software modernization (MMSM'2012)*, Bamberg 2012 Mar.
24. Northrop LM, Clements PC. A framework for software product line practice, Version 5.0. Software Engineering Institute 2005.