BRIDGING GAP OF USER GOAL REQUIREMENTS AND IMPLEMENTED APPLICATIONS BASED ON FEATURE MODEL

Ahmad Nurul Fajar, Eko K. Budiardjo, and Zainal A. Hasibuan

Faculty of Computer Science, Universitas Indonesia, Kampus Baru UI Depok, 16424

Email: ahmad.nurul@ui.ac.id

Abstract

Feature modeling is a conceptual thinking for identifying and classification feature in order for support software product lines. However, there are lack of the user goal requirements. It related with a technique for managing of features commonalities and variability. It has a hierarchy of features with variability and the purpose is to organize features. In practice of implemented applications, the feature model development lack of goal user requirement. The goal of user requirement in Indonesian government has described in document regulations. It should be a fundamental concern to develop e-government applications. However, In order to capture degree of software feature importance, some of features compared with implemented e-government applications. We have extracted some of features which can be compared with the implemented e-government applications. Our technique is extracted are derived from document regulations to business process model and feature model also. We Choose SIPKD and SIMDA applications which has implemented in Indonesian local government which has variation from one and another. We use extended AHP and S-AHP to find the prioritization of software features. The results are 80 features in SIPKD and 90 features in SIMDA. There are 65 features common and 25 variant features .This make un-optimization usage applications.

Keywords: *feature modeling, user goal requirements, e-government*

Abstrak

Feature modelling adalah sebuah konsep untuk mengidentifikasi dan mengklasifikasikan fitur untuk mendukung produk perangkat lunak. Namun, terdapat kekurangan dalam *user goal requirements*. Hal ini terkait dengan teknik untuk mengelola keumuman dan variabilitas fitur. *Feature modelling* memiliki hirarki fitur dengan variabilitas yang tujuannya adalah untuk mengelola fitur. Dalam praktiknya, pengembangan *feature modelling* kurang mendukung *user goal requirements*. *User goal requirements* di Republik Indonesia telah dijelaskan dalam dokumen peraturan yang baku. Hal ini harus menjadi perhatian mendasar untuk mengembangkan aplikasi *e-government*. Namun, untuk mendapatkan fitur-fitur dari *software* yang penting, beberapa fitur dibandingkan dengan menerapkan aplikasi *e-government*. Kami telah mengekstrak beberapa fitur yang dapat dibandingkan dengan menerapkan aplikasi *e-government*. Teknik ini diturunkan dari dokumen peraturan ke *bussines process model* dan *feature model*. Kami memilih aplikasi SIPKD dan SIMDA yang telah diimplementasikan di pemerintah daerah di Indonesia yang memiliki variasi satu sama lain. Kami menggunakan *extended* AHP dan S-AHP untuk menemukan prioritas fitur perangkat lunak. Hasilnya adalah 80 fitur dalam SIPKD dan 90 fitur dalam SIMDA. Ada 65 fitur umum dan 25 fitur varian. Hal ini membuat penggunaan aplikasi menjadi tidak optimal.

Kata kunci: *feature modeling, user goal requirements, e-government*

1. Introduction

Feature modeling has interesting technique of domain modeling, which particularly used in software product line development [1]. Feature Modeling can be reach to interesting topics among practitioners and researchers [18]. The purpose is for managing commonality and variability [1]. Feature modeling is a conceptual thinking for identifying and classification feature in order for support software product lines. However, there are

lack of the user goal requirements. A prerequisite for SPLE needed for Feature modeling. However, most users of feature modeling have difficulty in applying it to product line engineering [18].It also a technique for commonality and variability modeling in SPLE [1]. Beside that, feature model is a hierarchy of features with variability and the purpose of hierarchy is to organize features from multiple levels into detail [1]. Cardinality-based feature modeling is extended from FODA notation [2]. Common features among different products are modeled as mandatory features, while different features among them may be optional or alternative. Optional features [18]. The mandatory features represent selectable features for products of a given domain, the alternative features indicate that no more than one feature can be selected for a product [18].

SPLE is one of the most promising approach to reduce the development costs, as well as to increase the quality of families of similar software product instances [4]. The most critical indicators of SPLE is how the mechanisms capture common characteristics of a set of software applications in a specific problem domain [1]. In order to improve reusability, SPLE must have an ability to exploit commonality and manage variability. SPLE allows for variants rapid development of a domain specific application with using a common set of reusable assets. The assets here might be refer to software features. There are many approach to support the management of commonality and variability in the software development process [2][3]. Many researchers in the SPLE have proposed techniques with these challenges [1][2][3][4].

Ebrahim Bagheri et. al. introduce a new method called the Stratified Analytic Hierarchy Process (S-AHP) for prioritizing (ranking) of software features. S-AHP is concerned with finding the most appropriate set of features that need to be included in the final software product given the user requirements and their important goals and objectives. The output of S-AHP will be the input of typical feature modeling configuration algorithms that specialize a feature model based on the user goal requirement and relevant with constraints of integrity [4].

2. Literature

Informally, features are key distinctive characteristics of a product. We see different domain analysis methods use the term "feature" with slightly different meanings [19,20 in 18]. FODA defines a feature as a prominent and distinctive user visible characteristic of a system [19,20 in 18]. Several methods [11, 13] have been

developed and extended based on this definition. Ebrahim bagheri et all introduce a new method called the Stratified Analytic Hierarchy Process (S-AHP) for prioritizing (ranking) and filtering the features of a product family to enhance and expedite the feature selection and product configuration process. Variability is the ability of a system to be efficiently extended, changed, customized or configured for use in a particular context [2]. Another definition presents variability as the ability of a system or a development environment to support the production of a set of artifacts that differ from each other[5]. The variability gathers characteristics that differ from one product to another, while the product derivation is defined as a complete process of building products from the product line [2]. Concepts of commonality and variability are, respectively, used to designate common and variable elements in a PL [39 in 2]. Variability can concern two main aspects: optional or variation [7, 18 in 2].

A cardinality-based feature model is a hierarchy of features, where each feature has a feature cardinality. The FODA notations allow us to specify dependency relationships, called composition rules, between domain features [2]. FODA supports two types of composition rules: the "require" rule that expresses the presence implication of two or more features, and the "mutually exclusive" rule that captures the mutual exclusion constraint on feature combinations. The Software Product Line (SPL) development approach aims to reduce costs and time to market of systems that share a common domain [3]. The cardinality-based notation for feature modeling integrates four existing extensions of the FODA notation: feature cardinalities, group cardinalities, feature diagram references, and attributes [2]. Cardinality-based feature models are based on a meta model that defines their abstract syntax [6].

3. Feature Comparation with Implemented Applications : SIMDA and SIPKD

User Goal Requirement has described in Indonesian documents regulations. According to ZEF Framework [9,10,21] , this framework has a procedure to convert from document regulations to Business Process Model (BPM). The documents consists of dependency and independent of rules. The documents also consists of constraints among the level in the hierarchy government policies. There are M…N relation between theme, rules and objective.

After BPM already created from government regulations, we verify the BPM to the expert users. However, from BPM we could not identify

candidate features. In order to identify candidate features, we make depth interview to the expert users. We use RMUC approach for analysis problem domain and solution domain. Problem domain focus to capture problem and user need. Based on problem domain, we can identify feature in solution domain. It seems like requirement gathering phase. Requirement Traceability Matrix (RTM) used for tracing users requirement.

E-government applications in Cimahi and Bandung is for users interview and observation the implemented applications. We send questionnaire by email to users in Pamekasan and Tojo Una Una. The Domain for the case study are budgeting and local government finance. The implemented applications are called SIPKD and SIMDA. The application has been implemented and already use in many Indonesian local government.

Table I. described the sample of features compared with implemented applications. The features below it is not the all of the feature, it is only a sample of the features. The sample of features focus in creating transaction in major business processes at a domain.

4. Ranking of software feature and feature model

4.1. Concern Prioritization Ranking

According to S-AHP, the purpose of this stage are to compare and rank the list of concerns from user goal. All defined concerns on the feature model and their relative importance for the target application are considered, and the pair-wise comparison process (AHP) is undertaken to produce a ranked list of concerns. Table II described the concern and tag qualifier

In order to capture the concern of prioritization/ranking, the value is fullfilled by users. Table III described the prioritization of the concern.

After that, we use conventional AHP that having values 1, 3, 5, 7, and 9 have been used to represent the degree of importance showing equal value, slightly more value, strong valued, very strong value and extreme value, respectively. The steps of AHP how the relative importance of concern with respect to all other concerns. The priority of each concern is calculated and a relative ranking for the list of concerns is developed. Table IV described the priority of concern.

From the table IV, the list of ranking are information correctness, regulation correctness, time efficiency, and ease of use. Then, the next stage are consistency checking for validation the matrix calculation. The result are lambda Max =

6,731, CI = 0,122, CR = 0,098. From the result, it was valid because 0,098 < 0,1.

TABLE I
SOFTWARE FEATURES COMPARE

Software Feature	SIPKD		SIMDA	
	Cimahi	Pamekasan	Bandung	Tojo Una Una
Pembuatan Surat Perintah Membayar (SPM)	V	V	V	V
Pembuatan Data Kontrak	V	V	V	X
Pembuatan Surat Permintaan Pembayaran	V	V	V	V
Pembuatan Panjar TU (Tambahan Utang) dan Surat Pertanggung jawaban (SPJ) Pajak	V	V	V	X
Pembuatan Panjar TU dan SPJ Panjar TU	V	V	V	V
Pembuatan SPJ	V	V	V	V
Pembuatan mutasi kas tunai bank	X	X	X	X
Pembuatan Pendapatan UP	V	V	V	X
Pembuatan Jurnal konsolidator	V	V	V	X
Pengendalian Realisasi Triwulan	V	X	X	X
Pengendalian Perubahan Surat Penyediaan Dana (SPD) Bantuan Langsung (BL)	V	X	X	X
Pengendalian Dokumen Pelaksana Anggaran (DPA)	V	X	X	X
Pengendalian Surat Perintah Pencairan Dana (SP2D)dengan Surat Penyediaan Dana (SPD)	V	X	X	X
Integrasi Surat Tanda Setoran (STS) dengan Bank	X	X	X	X
Pemrosesan APBD	X	X	X	X

TABLE II
CONCERN TAG QUALIFYING

Concern	High	Middle	Low
Time Efficiency	A	B	C
Regulation Correctness	D	E	F
Information Correctness	G	H	I
Ease of Use	J	K	L

TABLE III
CONCERN PRIORITIZATION

	Time Efficiency	Regulation Correctness	Information Correctness	Ease of Use
Time Efficiency	1	1/2	1/3	2
Regulation Correctness	2	1	½	3
Information Correctness	3	2	1	4
Ease of Use	½	1/3	¼	1

TABLE IV
PRIORITY OF CONCERN

	Time Efficiency	Regulation Correctness	Information Correctness	Ease of Use	Priority
Time Efficiency	0,154	0,130	0,160	0,200	0,161
Regulation Correctness	0,308	0,261	0,240	0,300	0,277
Information Correctness	0,462	0,522	0,480	0,400	0,466
Ease of Use	0,077	0,087	0,120	0,100	0,096

4.2. Software Feature Ranking

Purpose of this stage is to earn the ranking of the relative importance of the qualifier tags of the remaining concerns. After that, we use them to rank the available features within the feature model. This will provide a final ranking over the most important concerns and their most significant qualifier tags. Given such a ranking, each feature can be prioritized based on the significance of its annotated concerns. The case study are related to the list of concern ranking, we choose information correctness and regulation correctness. The tag qualifier matrix compared described in table V. Table VI described the tag qualifier matrix with Priority.

The value from tag qualifier are : D (priority 0,130), E (priority 0,064), F (priority 0,050), G (priority 0,382), H (priority 0,223), I (priority

0,152). We choose case study in budget transaction and implementation business processes. Budget transaction and implementation business processes refer to the user goal. The user goal are budget incoming and out coming has created. We convert document regulations to create business processes. After that, user expert verifies the business processes. From business processes, we got 9 software features that related. After that, we explore implemented applications SIMDA in Bandung and SIPKD in Cimahi. We compare the software features which created from document regulations with implemented applications. First, user fulfill the questionnaire. Then, we distribute questionnaire to user, the purpose are for fulfill software features with tag qualifier. Each of software features has tag qualifier according to the user. It depends on the degree of user concern. Figure 1 shows the tag qualifier of software features.

TABLE V
PRIORITY OF CONCERN

	D	E	F	G	H	I
D	1	2	3	½	½	1/2
E	1/2	1	2	1/3	1/3	1/3
F	1/3	½	1	¼	¼	1/4
G	2	3	4	1	5	6
H	2	3	4	1/5	1	5
I	2	3	4	1/6	1/5	1

TABLE VI
PRIORITY OF TAQ QUALIFIER

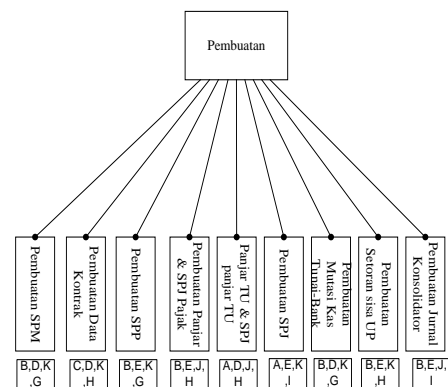|	D	E	F	G	H	I	Priority
D	0,128	0,160	0,182	0,204	0,069	0,038	0,130
E	0,064	0,080	0,130	0,136	0,046	0,025	0,064
F	0,043	0,040	0,061	0,102	0,034	0,019	0,050
G	0,255	0,240	0,242	0,408	0,686	0,459	0,382
H	0,255	0,240	0,242	0,082	0,137	0,382	0,223
I	0,255	0,240	0,242	0,068	0,027	0,076	0,152

Fig 1. Feature tag qualifier

Table VII shows calculation of software feature ranking. It described the value from each of software features.

TABLE VII
RANKING CALCULATION

No	Feature	Ranking Calculation
1	Pembuatan Surat Perintah Membayar (SPM)	B, D, G, K = 0 + 0,130 + 0382 + 0 = 0,512
2	Pembuatan Data Kontrak	C, D, K, H = 0 + 0,130 + 0 + 0,223 = 0,353
3	Pembuatan Surat Permintaan Pembayaran	B,E,K,G = 0 + 0,064 + 0 + 0,0382 = 0,287
4	Pembuatan Panjar TU (Tambahan Utang) dan Surat Pertanggung Jawaban (SPJ) Pajak	B,E,J,H = 0 + 0,064 + 0 + 0,223 = 0,287
5	Pembuatan Panjar TU dan SPJ Panjar TU	A, D, J, H = 0 + 0,130 +0 + 0,223 = 0,353
6	Pembuatan SPJ	A, E, K, I = 0 + 0,064 + 0 + 0,152 = 0,216
7	Pembuatan mutasi kas tunai bank	B, D, K, G = 0 + 0,130 + 0 + 0,382 = 0,512
8	Pembuatan Pendapatan UP	B, E, K, H = 0 + 0,064 + 0 + 0,223= 0,287
9	Pembuatan Jurnal konsolidator	B, E, J, I, = 0 + 0,064 + 0 + 0,152 = 0,216

From the calculation above, the list of ranking are : feature number 1 and 7 > feature number 2 and 5 >feature number 3 and 4 and 8 > feature number 6 and 9. Feature Create SPM and Feature Create bank cash mutation > Feature Create Contract and Feature > Feature Create DP TU and Feature Feature SPJ DP TU and Feature Create SPP and Feature Create DP and Feature Tax of SPJ > Feature Create incoming UP > feature Create SPJ > Feature Create Journal consolidator.

4.3. Feature Model : Feature Mandatory, Alternative, and Optional

We developed feature model from user survey. According to the SAHP method, the feature model consists of 7 major features that include mandatory, alternative and optional features.

From the case study above, especially in transaction and implementation business processes, there are no optional software features. We do not found the optional software features. It indicates that when user fulfill the questionnaire, the concern is ignored.

From the ranking calculation, we classifying feature mandatory, feature alternative and feature optional. Table VIII shows the software features. Software feature number 1 and 7 > feature number 2 and 5 >feature number 3 and 4 and 8 > feature number 6 and 9.
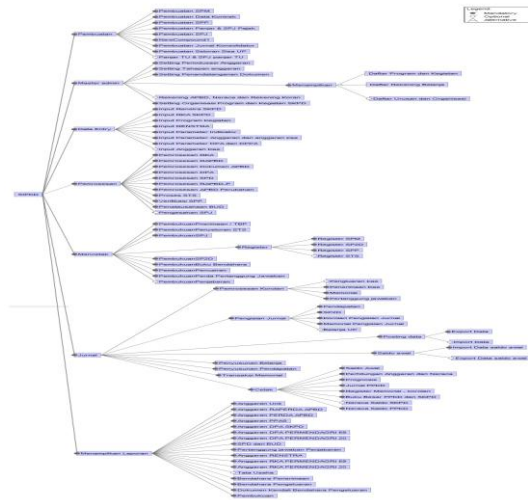

Figure 2. Feature model

TABLE VIII
FEATURE MANDATORY, ALTERNATIVE AND OPTIONAL

No	Feature	Mandatory	Alternative	Optional
1	Pembuatan Surat Perintah Membayar (SPM)	V		
2	Pembuatan Data Kontrak	V		
3	Pembuatan Surat Permintaan Pembayaran	V		
4	Pembuatan Panjar TU (Tambahan Utang) dan Surat Pertanggungjawaban (SPJ) Pajak	V		
5	Pembuatan Panjar TU dan SPJ Panjar TU	V		
6	Pembuatan SPJ		V	
7	Pembuatan mutasi kas tunai bank	V		
8	Pembuatan Pendapatan UP	V		
9	Pembuatan Jurnal konsolidator		V	

5.   Conclusions and Future Work

The results of S-AHP and AHP calculation in the simple case are Feature Create SPM and Feature Create bank cash mutation > Feature Create Contract and Feature > Feature Create DP TU and Feature Feature SPJ DP TU and Feature Create SPP and Feature Create DP and Feature Tax of SPJ > Feature Create incoming UP > feature Create SPJ > Feature Create Journal consolidator. From the case study above, especially in transaction and implementation business processes, there are no optional software features (0 optional features). We do not found the optional software features. It indicate, when user

fulfill the questionnaire, the concern is ignore. The purpose of combination AHP and S-AHP is to communicate with the user and understand their priorities with regards to business objectives and high-level goals and to use this information to find the most important features of a feature model for the user goal. Future work, we would like to experiment in more domain application such as public services, citizen administration and PAD (Local Incoming Tax).

References

[1] Chang Hwan Peter Kim, *"On the Relationship between Feature Models and Ontologies"*, University of Waterloo, Waterloo, Ontario, Canada, 2006

[2] Krzysztof Czarnecki, Chang Hwan Peter Kim, Cardinality-Based Feature Modeling and Constraints*: A Progress Report, OOPSLA'05Workshop on Software Factories,* October 17, 2005, San Diego, California, USA. ACM 1-59593-193-7/05/0010

[3] T. Ziadi and J.-M. Jézéquel, *"Software Product Line Engineering with the UML: Deriving Products"*, inria-00512550, version 1 - 30 Aug 2010

[4] Ebrahim Bagheri et all , " *Stratified Analytic Hierarchy Process: Prioritization and Selection of Software Features"*

[5] Rodrigo Bonif´acio de Almeida, *"Modeling Software Product Line Variability in Use Case Scenarios An Approach Based on Crosscutting Mechanisms"*, Final Thesis, Universidade Federal de Pernambuco Centro de Inform´atica P´os-gradua¸c˜ao em Ciˆencia da Computa¸c˜ao, 2010

[6] Bassam Atieh Rajabi, Sai Peck Lee, *"Modeling and Analysis of Change Management in Dynamic Business Process"*, International Journal of Computer and Electrical Engineering, Vol. 2, No. 1, February, 2010 1793-8163

[7] Sven Apel et all, *"An Algebra for Features and Feature Composition"* , J. Meseguer and G. Ro¸su (Eds.): AMAST 2008, LNCS 5140, pp. 36–50, 2008

[8] P. Clements and L. Northrop, *Software product lines: practices and patterns*, Addison- Wesley Longman Publishing, 2001

[9] Eko K Budiardjo,Ahmad Nurul Fajar,Zainal A Hasibuan,*"ZEF Framework for e-government applications: Featuring SOA &*

BPM Allignment", International Journal Computer Theory and Engineering, Vol.5, number 2, 2012

[10] Ahmad Nurul Fajar, Eko K Budiardjo, Zainal A Hasibuan *"System Architecture In The Dynamic Environment Based on Commonality and Variability Business Processes*, 8th ICCM, Seoul, 2012

[11] Tommi Myllymäki,*"Variability Management in Software Product Lines"*, Tampere University of Technology Software Systems Laboratory ARCHIMEDES, 2010

[12] I. Sommerville and P. Sawyer, *"Viewpoints: principles, problems and a practical approach to requirements engineering,"* Annals of Software Engineering, vol. 3, 1997, pp. 101–130.

[13] T.L. Saaty, *The Analytic Hierarchy Process*, New York: McGraw-Hill, 1980.

[14] Krzysztof Czarneck et all, *"Cool Features and Tough Decisions: A Comparison of Variability Modeling Approaches"*, VaMoS'12, January 25–27, 2012, Leipzig.Germany. Copyright 2012 ACM 978-1-4503-1058-1 ...$10.00

[15] Franck Fleurey et all, *The Common Variablility Language (CVL)*

[16] Sebastian Oster et all, *"Integrated Modeling of Software Product Lines with Feature Models and Classification Trees"*

[17] Franck Fleure et all , *"Standardizing Variability – Challenges and Solutions"*, I. Ober and I. Ober (Eds.): SDL 2011, LNCS 7083, pp. 233–246, 2011. © Springer-Verlag Berlin Heidelberg 2011

[18] Kwanwoo Lee et all, *"Concepts and Guidelines of Feature Modeling for Product Line Software Engineering"*

[19] Griss, M. L., Favaro, J., d'Alessandro, M.:*" Integrating Feature Modeling with the RSEB"*, Proc. Fifth International Conference on Software Reuse, Victoria, BC, Canada (1998) 76-85

[20] Kang, K., Kim, S., Lee, J., Kim, K., Shin E., Huh, M.: *"FORM: A Feature-Oriented Reuse Method with Domain-Specific Reference Architectures"*, Annals of Software Engineering, 5 (1998) 143-168

[21] Ahmad Nurul Fajar, Eko K.Budiardjo, Zainal A.Hasibuan, *"SOA and BPM Allignment on ZEF Framework : The Indonesian Government Case"*, ICCSIT, Chengdu, China, 2011