ISSN 2278-3091
Volume 8, No.1.4, 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse7981.42019.pdf
https://doi.org/10.30534/ijatcse/2019/7981.42019

Requirement Patterns: An Approach for Streamlining Requirements Engineering in Software Product Families

Badamasi Imam Ya'u[1,2], Azlin Nordin[1], Norsaremah Salleh[1], Saleh Waziri Mustapha[2]
[1]International Islamic University Malaysia, badamasi.imam@live.iium.edu.my; {azlinnordin & norsaremah}@iium.edu.my
[2]Abubakar Tafawa Balewa University, Bauchi, Nigeria, {biyau & wsmustapha} @atbu.edu.ng

ABSTRACT

Reusable structure is essential in all reuse-based software development processes. This provides a solid foundation for seamless management of reusable artefacts especially in software product line engineering (SPLE). One of the potential benefits provided by a well-defined structure is systematic reuse of these artefacts. Requirements pattern approach provides guidelines for requirement engineers to reuse and specify requirements. Although a plethora of research on requirements pattern have been reported in the literature, no research available focuses on requirement engineering (RE) activities of SPLE. In this paper, we present an anatomy of software requirement pattern (SRP) for SPLE with a structured example from e-learning domain. To enable practitioners, understand the concept of requirement pattern more, we present a meta-model for the SRP concepts and their relationships. In addition, we describe how the requirement pattern approach, streamlines RE activities, design for and with reuse in both domain and application engineering processes of SPLE. The requirement pattern approach thus helps in achieving systematic requirements reuse (RR) and generation of structured software requirement specification (SRS) for individual applications.

Key words: Design for and with Reuse, Meta-model, Requirements Reuse, Software Requirement Patterns, Software Product Line Engineering

1 INTRODUCTION

Consideration for appropriate structure in general software development is essential. This gives flexibility for successive and recurring processes involved in the software development processes. According to Structure-Process-Outcome model, construction of structure should be considered before other stages of development [1].

Software requirements pattern (SRP) plays a vital role in RE especially in guiding requirement engineers for effective writing of requirement specification [2], [3]. Furthermore SRP, facilitates the processes of retrieving and presenting requirements, which express certain goal [4]-[6]. SRP simplifies RE activities, which include requirements analysis, validation and documentation [7]. With respect to system design, SRP supports the process of gathering system features, which can be revamped with multiple designs and implementations [8], [9]. Thus, requirements pattern is characterized as a framework, which aids and enhances a systematic requirement reuse (RR) [2], [6], [10].

SPLE methodology focuses on management of common and variable artefacts such as requirements. Thus, SPLE offers opportunity for systematic reuse of requirements [11]. However, to optimize reuse in SPLE, a substantial effort is invested for a design and construction of a structure that supports and enables reuse of software requirements artefacts. One of the activities that complicates RE in SPLE is dealing with delta requirements. Delta requirements are called requirements for enhancement of an existing system [12]. As such, delta requirements should be specified separately to truncate the extraneous effort to specifying complete requirements of the system from the scratch.

This paper presents an SRP structure for SPLE based on RePa, which is requirements pattern template proposed at the international workshop for requirements pattern [13]. With this requirement pattern structure, we present a detailed example of e-learning domain requirement pattern and a metamodel of SRP in SPLE.

This paper is structured as follows. Section 2 presents the related work on SRP; an anatomy of software requirements pattern from e-learning domain is presented in Section 3; while Section 4 presents a metamodel for SRP; Section 5 describes design for and with reuse in SPLE; Section 6 presents discussion of the paper and 7 presents the conclusion and future work.

2 RELATED WORK ON SRP

In approach, known as pattern-based requirements elicitation (PABRE), an SRP is proposed and presented to facilitate requirement elicitation and reuse [14]-[16].

The literature of requirements patterns shows that a number of studies on requirements patterns focus on software security [17]-[21]. For example, the certain researches focus on exploiting the advantages of requirements patterns structure to improve the principle of trust engineering and legal requirements [21]-[23]. Roher

argued that requirements patterns should be applied to the requirements of sustainability domain [24]. Other category of researchers focus on functional requirements [25], non-functional requirements [4], [5].

In different circumstances, researchers emphasize on catalogue of particular domains, which include (1) *i-star* models for Online Social Network (OSN) [26]; (2) 30 extensive requirements patterns for development of web applications [8]; (3) Seismology Requirement patterns [9]; (4) SRP for online examination system [7]; (5) SRP for Call-for-tenders processes [5]; requirements patterns for content management systems domain [25]; (6) SRP for embedded system [27], [28]; and (7) SRP for information systems [29].

The majority of the requirements pattern approaches above targeted different SPLE domains. However, their goals did not focus on RE activities in SPLE, which provides a requirements structure for systematic reuse. In the subsequent section, a SRP structure for SPLE with example from e-learning domain is presented. We chose e-learning because the following reasons (1) during our search for SPLE domain requirements specification document, e-learning domain has the highest turnover (2) the sources of requirements of e-learning are more authentic (3) the availability of resource and participants for the evaluation of our approach. Details of the SRP structure can be found in our previous research [30].

3 ANATOMY OF REQUIREMENTS PATTERN FOR SOFTWARE PRODUCT LINES

This section presents a structure of SRP with example from e-learning domain. We use e-learning as a case study because of its popularity as a global learning community [31] In addition to RePa template [13], the structure is augmented with three more sub-sections, which include considerations for design, development and testing. In software development, design is a critical issue especially when reuse is emphasized. In our proposed requirement pattern restructure, consideration for design sub-section provides detailed information to requirement engineering for what kind of design antecedents is required before continuing to specify a requirement of that type. The use of two sections, that is *Consideration for Development* and *Consideration for Testing* is adopted from Withal [3]. This suggestion also complements the solution of requirement patterns in SPLE RE problems. Table 1 illustrates the anatomy of requirement pattern structure for an *authentication* requirement in SPLE. The sections and sub-sections with an asterisk denote that such sections and sub-section are mandatory in the pattern template. A number of studies present the discussion on the current template, which include pattern forms, template's sections and parts (for example, fixed and extended parts) [4], [5], [13], [25].

The *Solution Section* provides detailed description of *requirement pattern* for the *Authentication requirement* as a pattern product type. As can be seen Table 2, our "Solution Section" is unique from existing templates by introducing the commonality, variability sub-sections. This enables the requirement engineer to vividly present details of common and variable requirements separately to facilitate reuse of SPLE common requirements, which are planned for all products and specification of variable requirements according to the type and constraints of variation points in the requirements. Thus, the *Solution Section* comprises of three subsections, which are (1) *Common Requirement*; (2) *Variable Requirement*; and (3) *Variability Model*.

Table 1: Requirement Pattern for Authentication Requirement

Section/Subsection		Remark	
***Pattern ID**		RP2	
***Pattern Name**		Authentication	
Also Known As		Login	
Authors		Stephen Withal	
Date Created		2007	
***Context/ Applicability**	****RE Activity***	Specification	
	****Pattern Type***	Product	
	Business Domain	E-learning	
	Organization Environmental Factors	Teaching and Learning Environment	
	Stakeholders	***Role***	Students, Instructors, Teachers, Administrators
		Goal	To use e-learning application in running and delivering their organization responsibilities
***Problem AKA Intent and Objective**		Poor security measures to protect unauthorized access to information system	
***Force**		A cutting-edge e-learning security facility to protect teaching and learning applications	
***Solution**	****Solution ID***	PS2.1	Refer to "Solution" Section
***Application and Example**		***Application:*** The pattern is used in building security measures to access e-learning application	

		Example: The user/ administrator should provide valid authentication credentials before accessing the system	
***Known Uses**		Business Software Systems, refer to Withall's Catalogue [3].	
Cataloguing:	***Classification***	*Type*	Security, access control
		Default Value	Maybe
		Purpose	To check whether the functionality of the requirements of this type that shall be provided by the system is satisfied.
		Audience Role	Software and Requirement engineers
		Audience Goal	Development of software requirement specification for e-learning applications
		Allowed Value	Yes
	Related Pattern	*ID*	RP7
		Name	Accessibility
		Relation Type	***Extends*** No
			Refers Yes
Custom Section	***Consideration for Design***	*Description*	This describes the aspect of the design that should be considered for the requirements of this type.
		Purpose	This highlights the reason why the design for the implementation of the type of requirement is considered
		Constraint	This provides with those design constraints a software designer should consider.
		Design Pattern	This lists the name of the design pattern that corroborates with this requirement pattern.
		Design Guide	This highlights a step by step guide for designing the implementation of requirement of this type.
	Consideration for Development	*Description*	This describes the needs for considering the development of the functionality of requirement of this type.
		Purpose	This details the purpose for considering the implementation of requirement of this type
		Constraint	This clearly shows the kinds of constraints that affect the implementation of requirement of this type
		Development Guide	During the development, implement the system in such a way that it conceals user login password and make it undecipherable to any potential attacker.
	Consideration for Testing	*Description*	This describes the needs for testing the functionality of requirement of this type.
		Purpose	This states the reasons for considering the testing for the functionality of the requirement of this type.
		Constraint	This describes the constraints for testing the requirement of this type.
		Test Type	This state the type of testing executed for the function of the requirement of this type
		Test Guide	User authentication should be tested at two different levels: 1. Functional: test if the authentication process works properly in any situation and accessibility of the system. 2. Security: the authentication process should be tested so that the security steps cannot be easily learned, subverted and bypassed by an attacker.

Table 2: Solution Section- Authentication

Solution ID	PS2.1		
Pattern Name	Authentication		
Goal	Ensure user access control		
Description	The pattern prompts user to login with valid username and password		
Requirement	***ID***	RQ2.1.1	
	Name	Controlling user Access	
	Type	Functional	
	Description	This requirement defines that the system shall provide a mechanism for a user access	
	Priority	High	
Common Requirement	***ID***	CR2.1.1.1	
	Description	This form establishes the need to access control for all users of the system	
	Constraints	Fixed part (1) Extended part: Valid User ID and password	
	Fixed Part	*Form Text*	The system shall provide a mechanism for controlling a user access to the system
	Extended Part	*Form Text*	The system shall provide access to users with valid user ID and password
Variable Requirement	***ID***	VR2.1.1.1	
	Description	This form shows variable requirements for specifying different user authentication variation points in the product line	
	Constraints	Fixed part (1) Variable part: 1. User ID & Password 2. User ID & Password and Biometric data	
	Fixed Part	*Form Text*	The system shall provide access to users with either valid user ID consists of simple characters or valid email address and either valid simple password or with password with specified characters. The system shall either use the combination of user ID and password or user ID and biometric fingerprint scan or both.
	Variable Part	*Variation Points (VP)*	Authentication
		Variants (V)	(1) User ID & Password (2) User ID & Password and Biometric data
Variability Model	***Description***	This form establishes the need to use orthogonal variability model to show and trace the level of variations in different requirements artefacts.	
	Constraints	Focus on orthogonal variability models	
	Model (s)	Textual requirements, feature models, traditional requirement model, UML models	

4 METAMODEL FOR SRP IN SPLE

RE activities in SPLE requires appropriate strategy beforehand to enable smooth accomplishment of domain requirements elicitation, analysis and documentation that can be exploited for future developments of related but distinctive products [11].

SRP has been recognized as a desirable approach for writing reliable requirements specifications [3]. This section presents a metamodel for requirement pattern in SPLE. The metamodel represents the pattern concepts (elements) and their relationships. As can be seen in Figure 1, the metamodel constitutes of all sections and subsections of the anatomy of requirement pattern described in Table 1 and Table 2 such as the requirement pattern and solution sections. The metamodel provides a logical view to software developers to deeply understand the relationships amongst concepts and what design and development constraints should be considered.

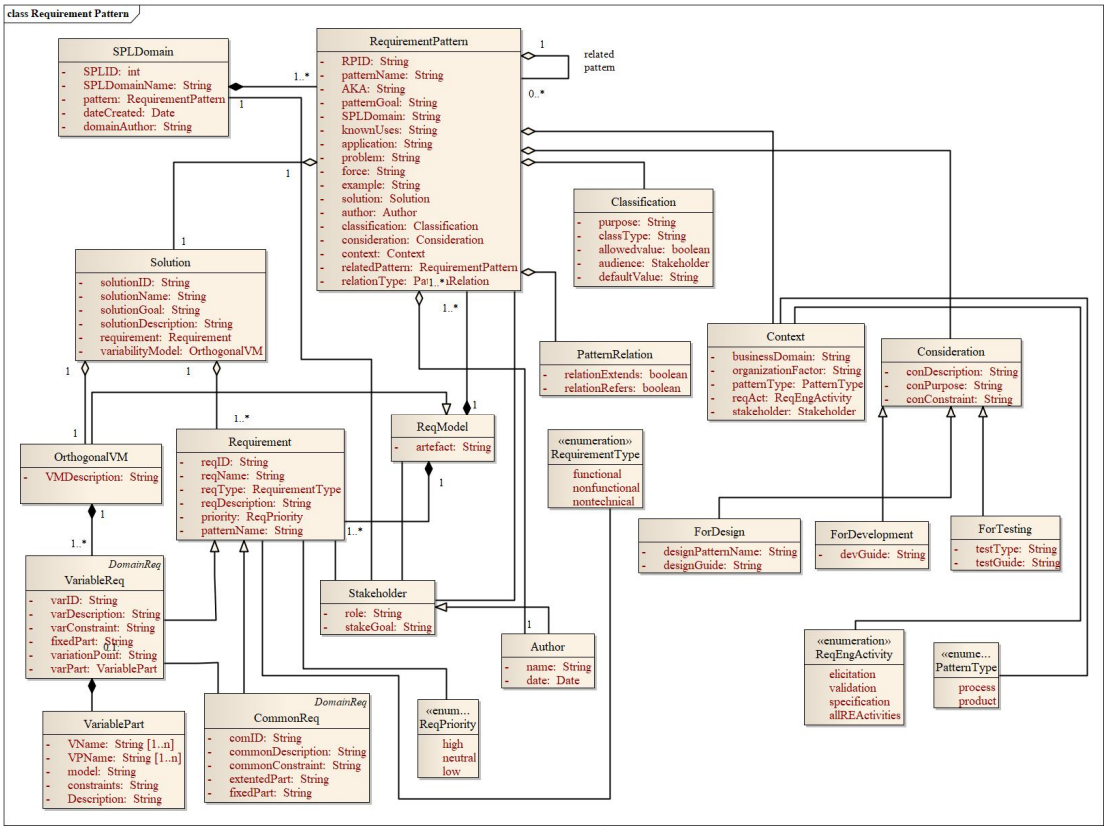

Figure 1: Metamodel for Requirement Pattern

5 DESIGN FOR REUSE AND WITH REUSE IN SPLE USING SRP

To avoid discrepancies in reusing requirements artefacts at different development stages, consideration for design for and with reuse should be planned beforehand. In this section, an excerpt of SPLE processes, covering domain and application engineering processes is presented. The two processes describe the concept of design for reuse and with reuse respectively.

Figure 2 demonstrates the activities of domain and application engineering processes, a repository of requirements artefacts and the interaction of both SPLE processes with a stakeholder.

Normally, SPLE process commences at product management subprocess of domain engineering. During the product management, the SPLE domain goal and visualized variable features of different product are determined. The output of product management is passed to the next subprocess, which is domain RE. During domain RE, major activities such as analysis of common and variable requirements, documentation of analyzed requirements coupled with variability model are conducted. At this stage, this research proposes a requirement pattern structure aimed at streamlining the RE activities by forming a base for requirements reuse.

At the next stage, a pattern template containing common and variable requirements together with a model of variabilities are kept in a repository for future use. At the time of application engineering activities, all new requirements for applications are crosschecked to conform with the existing domain requirements. This results in a systematic reuse of requirements and thus production of a well-structured software requirement specification (SRS) for specific applications.

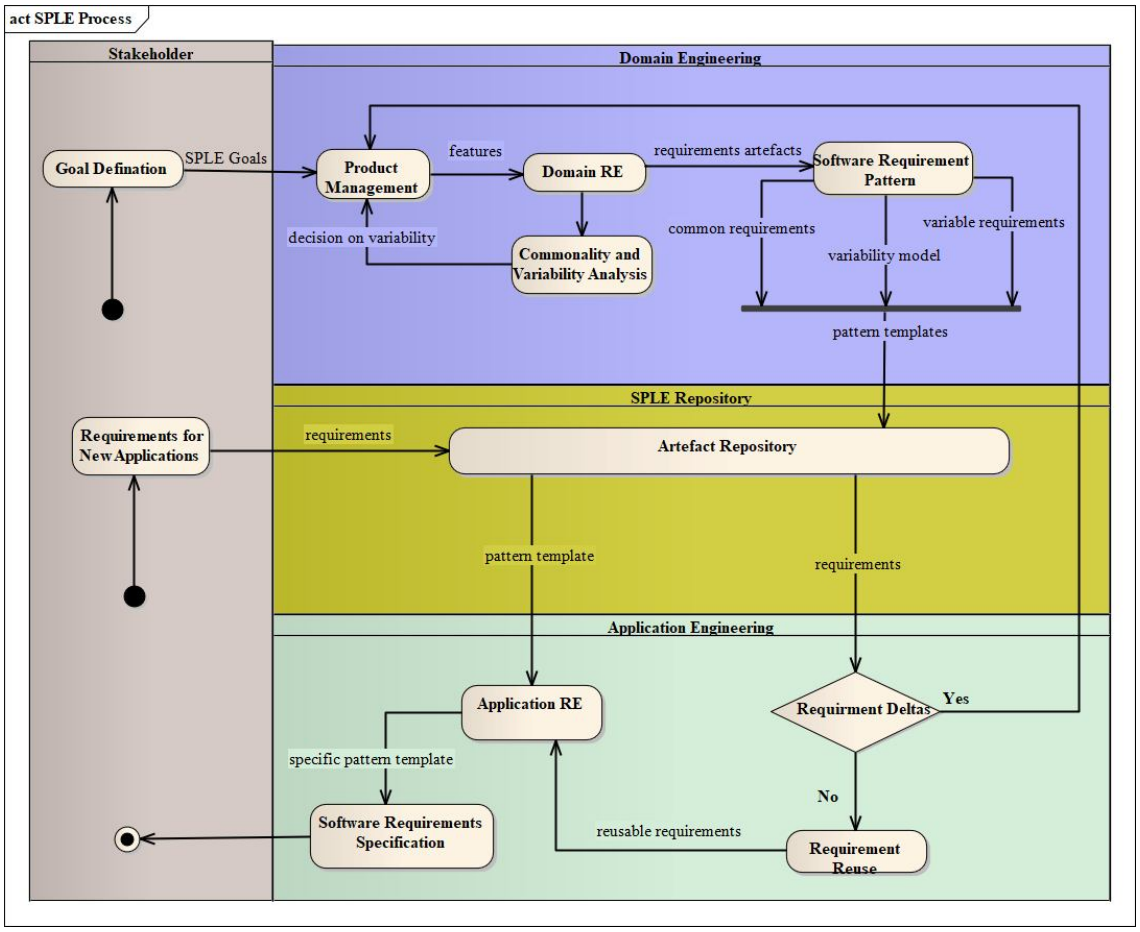

Figure 2: Design FOR and WITH Reuse Activities in SPLE

6 DISCUSSION

It is obvious that reuse depends on appropriate planning especially when developing very large and evolutionary systems such as in SPLE. In a study [32], it is revealed that the process of discovering the variability of requirements artefacts at the product management and domain RE is done in an ad-hoc manner. This indicates that supplementary research is required on systematic way of streamlining and improving requirements engineering activities in software product lines.

Requirement patterns approach can offer a seamless design for reusing requirements artefacts during domain engineering and derivation of requirements artefacts with reuse during application engineering. Our approach describes how SRP could streamline RE activities in the domain engineering and application engineering processes in software product lines. For this reason, we argue that requirements patterns are potential players to boost RE activities in software product lines.

7 CONCLUSION

SRP offers a desirable structure for delineating, reusing and specifying requirements' artefacts. In this paper we report that plethora of studies on requirement patterns exist in the literature. However, we have discovered an important gap on RE of SPLE, which left unfilled. To fill this gap, we present the structure of requirement patterns proposed in our previous study with a clear example from e-learning domain. We also present a metamodel consisting of all pattern concepts/ elements and their relationships. This gives a clear understanding for practitioners on what to consider during design and development SPLE requirements. Furthermore, we show how requirement patterns approach plays a vital role for enhancing *design for* and *with reuse*, which leads to systematic reuse of requirements and generation of detailed SRS.

We are currently working on an instrument for expert validation to evaluate the applicability and suitability of requirement patterns approach in SPLE RE activities.

ACKNOWLEDGEMENT

This research was sponsored by Abubakar Tafawa Balewa University (ATBU) Bauchi and Tertiary Education Trust Fund (TETFund) of Nigeria. We would also like to thank the Department of Mathematical Sciences, Faculty of Science, ATBU for the support and promotion of research activities.

REFERENCES

[1] C. Gonzalez-Perez and B. Henderson-Sellers. ***Metamodelling for software engineering***: Wiley Publishing, 2008.

[2] B.I. Ya'u, A. Nordin, and N. Salleh. **Investigation of Requirements Reuse (RR) Challenges and Existing RR Approaches**. *Advanced Science Letters*, **23**(5): p. 4101-4105, 2017. https://doi.org/10.1166/asl.2017.8232

[3] S. Withall. ***Software requirement patterns***: Pearson Education, 2007.

[4] C. Palomares, C. Quer, X. Franch, C. Guerlain and S. Renault. **A catalogue of non-technical requirement patterns**, *in Second IEEE International Workshop on Requirements Patterns (RePa)*, 2012. https://doi.org/10.1109/RePa.2012.6359969

[5] S. Renault, O. Méndez Bonilla, J. Franch Gutiérrez, B. Quer and C. Maria. **A Pattern-based Method for building Requirements Documents in Call-for-tender Processes**, *International journal of computer science & applications,*.**6**(5): p. 175-202, 2009.

[6] B.I. Ya'u, A. Nordin, and N. Salleh. **Software Requirements Patterns and Meta model: A Strategy for Enhancing Requirements Reuse (RR)**, in IEEE *International Conference on Information & Communication Technology for the Muslim World (ICT4M 2016)*, Jakarta Indonesia, 2016. https://doi.org/10.1109/ICT4M.2016.047

[7] S. Srivastava. **A Repository of Software Requirement Patterns for Online Examination System.** *International Journal of Computer Science Issues (IJCSI '13*), 2013.

[8] R.S. Wahono and J. Cheng. **Extensible requirements patterns of web application for efficient web application development,** in *Proceedings* of *First* IEEE *International Symposium on Cyber Worlds*, 2002.

[9] Y. Li, C. Pelties, M. Käser, and N. Nararan. **Requirements patterns for seismology software applications,** in *IEEE Second International Workshop on Requirements Patterns (RePa)*, 2012.

[10] F.B.V. Benitti and R.C.d. Silva. **Evaluation of a Systematic Approach to Requirements Reuse.** *Journal of Universal Computer Science*, **19**(2), 2013.

[11] K. Pohl, G. Böckle, and F.J. van Der Linden. **Software product line engineering: foundations, principles and techniqu**e*s*, *Springer Science & Business Media:* 2005.

[12] A. Herrmann, A. Wallnöfer, and B. Paech. **Specifying changes only–a case study on delta requirements,** in *International Working Conference on Requirements Engineering: Foundation for Software Quality*, Springer 2009. https://doi.org/10.1007/978-3-642-02050-6_5

[13] L. Chung, B. Paech, L. Zhao, L. Liu, and S. Supakkul. **Repa requirements pattern template**, in *International Workshop on Requirements Patterns (RePa '12)*, 2012.

[14] S. Renault, O. Méndez-Bonilla, X. Franch, and C. Quer. **PABRE: pattern-based requirements elicitation,** in IEEE *Third International Conference on Research Challenges in Information Science, RCIS*, 2009. https://doi.org/10.1109/RCIS.2009.5089271

[15] C. Palomares, C. Quer, and X. Franch. ***PABRE-Man: Management of a requirement patterns catalogue,*** 2011.

[16] C. Palomares, C. Quer, and X. Franch. ***PABRE-Proj: Applying patterns in requirements elicitation***, 2013.

[17] T. Okubo and H. Tanaka. **Web security patterns for analysis and design,** in ACM *Proceedings of the 15th Conference on Pattern Languages of Programs*. 2008.

[18] M. Hafiz, P. Adamczyk, and R.E. Johnson. **Organizing Security Patterns**, *IEEE Software,*. **24**(4): p. 52-60, 2007.

[19] K. Beckers, I. Côté, and L. Goeke. **A catalog of security requirements patterns for the domain of cloud computing systems**. in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*. 2014. https://doi.org/10.1145/2554850.2554921

[20] X. Xuan, Y. Wang, and S. Li. **Privacy requirements patterns for mobile operating systems**. in *IEEE 4th International Workshop on Requirements Patterns (RePa)*, 2014. https://doi.org/10.1109/RePa.2014.6894842

[21] A. Hoffmann, M. Söllner, H. Hoffmann, and J. M. Leimeister, Jan Marco **Towards trust-based software requirement patterns**. in *IEEE Second International Workshop on Requirements Patterns (RePa)*, 2012.

[22] A. Hoffmann, M. Söllner, and H. Hoffmann. ***Twenty software requirement patterns to specify recommender systems that users will trust***, 2012.

[23] A. Hoffmann, T. Schulz, H. Hoffmann, S. Jandt, A. Roßnagel, and J. M. Leimeister. **Towards the use of software requirement patterns for legal requirements**. 2012.

[24] K. Roher and D. Richardson. **Sustainability requirement patterns,** in *IEEE Third International Workshop on Requirements Patterns (RePa)*, 2013. https://doi.org/10.1109/RePa.2013.6602665

[25] C. Palomares, C. Quer, X. Franch, S. Renault, and C. Guerlain. **A catalogue of functional software requirement patterns for the domain of content management systems**. in *Proceedings of the 28th annual acm symposium on applied computing*, 2013. https://doi.org/10.1145/2480362.2480598

[26] S. Bouraga, I. Jureta, and S. Faulkner. **Requirements engineering patterns for the modeling of online social networks features,** in *IEEE 4th International Workshop on Requirements Patterns (RePa)*, 2014. https://doi.org/10.1109/RePa.2014.6894841

[27] S. Konrad and B.H. Cheng. **Requirements patterns for embedded systems**, in *Proceedings of IEEE Joint International Conference on Requirements Engineering*, 2002.

[28] D. Dietrich and J.M. Atlee. **A pattern for structuring the behavioural requirements of features of an embedded system**, in *3rd International Workshop on Requirements Patterns (RePa)*, 2013. https://doi.org/10.1109/RePa.2013.6602664

[29] A. Durán, B. Bernárdez, M. Toro, R. Corchuelo, A. Ruiz, and J. Pérez. **Expressing customer requirements using natural language**

requirements templates and patterns, in *IMACS/IEEE CSCC'99 Proceedings*, 1999.

[30] B. I. Ya'u, A. Nordin, N. Salleh and I. Aliyu. **Requirements Patterns Structure for Specifying and Reusing Software Product Line Requirements,** in *2018 International Conference on Information and Communication Technology for the Muslim World*, KICT, International Islamic University Malaysia, 2018. https://doi.org/10.1109/ICT4M.2018.00042

[31] M. Abdellatief, A. B. M. Sultan, M. A. Jabar, and R. Abdullah. **A technique for quality evaluation of e-learning from developers perspective.** *American Journal of Economics and Business Administration*, **3**(1): p. 157-164, 2011. https://doi.org/10.3844/ajebasp.2011.157.164

[32] E. Almentero, E. Cirilo, C. Lucena, J. C. S. do Prado Leite and A. L. de Castro Leal. **Uncovering product line variability from early requirement documents**, in *2013 3rd International Workshop on Requirements Patterns (RePa)*. 2013. https://doi.org/10.1109/RePa.2013.6602670