

Integration of Heterogeneous Requirements using Ontologies

Ahmad Mustafa, Wan M.N. Wan-Kadir,
Noraini Ibrahim
Software Engineering Department
Universiti Teknologi Malaysia
Johor Bahru, Malaysia

Muhammad Arif Shah
Software Engineering Department
Universiti Teknologi Malaysia, Johor Bahru, Malaysia
City University of Sciences and Information Technology,
Peshawar Pakistan

Muhammad Younas
Software Engineering Department
Universiti Teknologi Malaysia, Johor Bahru, Malaysia
Department of Computer Science
Government College University, Faisalabad, Pakistan

Abstract—Ontology-driven approaches are used to sustain the requirement engineering process. Ontologies can be used to define information and knowledge semantics during the requirements engineering phases, such as analysis, specification, validation and management of requirements. However, requirement analysts face difficulties in using ontologies for requirement engineering. In this study, a framework has been proposed to integrate heterogeneous requirements by using local and global ontologies.

Keywords—Heterogeneous requirements; requirement engineering; local ontologies; global ontologies

I. INTRODUCTION

The success of software system can be quantified by the degree to which it meets the proposed envision. Software Requirements Engineering (RE) is a well-defined process to identify stakeholders and their needs. It is a way to document the requirements for agreeable analysis, communication, and further implementation [1]. Moreover, all the efforts and resources applied during RE process must be reflected in the developed system in term of quality product and delivery of product time to market. However, Standish group report illustrates that 31.1% of projects are canceled before completion, and 52.7% of projects cost increased up to 189% to its original estimated cost. Moreover, many projects are failed due to lack of user input, incomplete requirement specifications and change in requirement specifications [2].

In the early phases of software development process, RE emphasizes on elicitation analysis, specification, validation, and management of requirements [3]. It is recognized that RE highlights on the enhancement of the quality of system under development. Moreover, RE focuses on reasons that may

propagate the risks such as budget overrun, time delay and project failures [4], [5].

In software development industry and academia, ontologies are being used in requirements engineering phase. Ontology is defined as “explicit specification of a conceptualization” [6], [7]. Ontology has explicit classes and properties and used as a standard form for knowledge representation of concepts inside a domain. Furthermore, it establishes an association in such a way that is allowable for automated reasoning [6], [7].

Furthermore, the ontological concepts can be used to address or resolve various kinds of issues in RE. It is used to write complete, unambiguous and consistent requirements statements. Furthermore, ontologies can be used to manage heterogeneous requirements, accomplish consistency analysis, represents domain knowledge model and requirements changes [8]-[10].

Rest of the paper is organized as follows. Section II describes the overview of heterogeneous requirements. Section III describes the related studies. The proposed approach is mentioned in Section IV which have following subsections, ontology as shared vocabulary using Multilanguage WordNet, generic requirements formalism using Pivot Model and transformation of requirements into Pivot Model. Section V describes the limitation of the study, and the last section the conclusion and future work of the study.

II. OVERVIEW OF HETEROGENEOUS REQUIREMENTS

A Multinational Corporation (MNC)¹, has several departments situated in different countries. For instance, as shown in Fig. 1 requirements are described in Malay, Arabic and German languages.

¹ https://en.wikipedia.org/wiki/Multinational_corporation

III. RELATED MATERIAL

A. Requirements Engineering

Requirements engineering activities aim to manage requirements-related knowledge such as natural language documents, storyboards, use cases, and business process specifications. These artifacts are called Requirements Document. A requirement document is an initial process in software development process. The development of these documents is taken as one of the challenging task [11].

Requirements engineering is concerned with (a) elicitation: actors and requirements identification, (b) modeling of identified requirements, (c) analysis of requirements to detect inconsistency and ambiguity and (d) validation of requirements [12]. According to Sommerville [3], the requirement statements are the descriptions of what services the system should provide and the restrictions on their operations. These requirements imitate the needs of customers for a system that serves a specific purpose such as placing an order, finding information, and controlling a device. The process of discovering, analyzing, recording requirements documents and verifying these services and constraints is known as requirements engineering (Fig. 2).

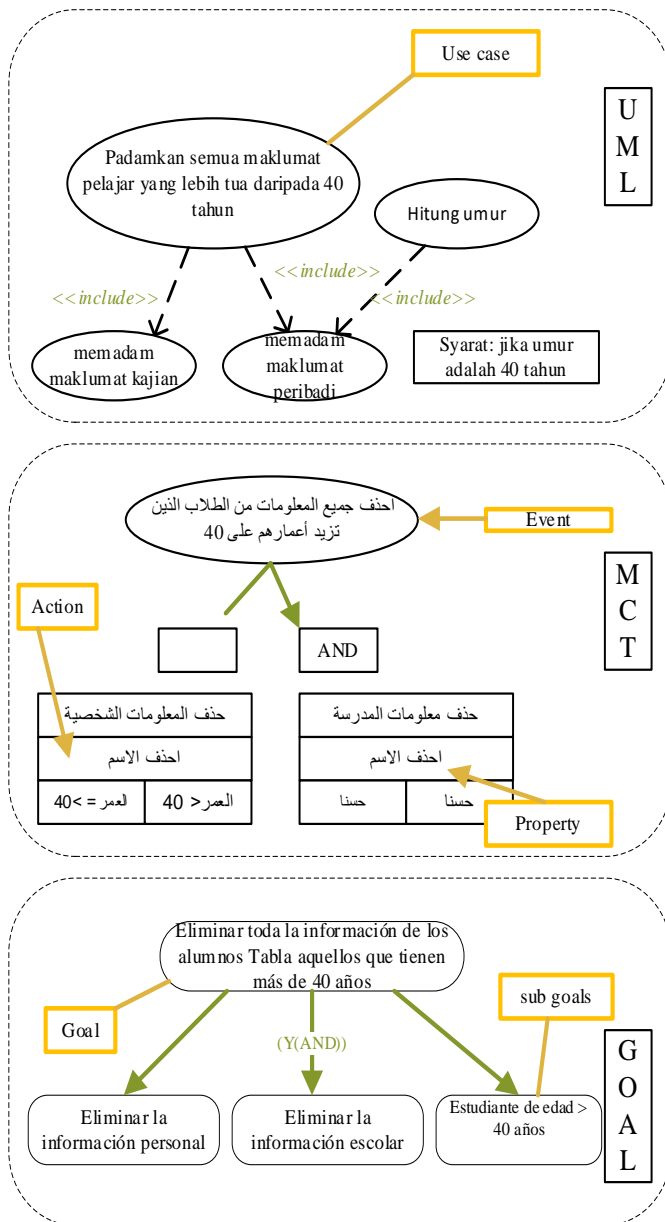


Fig. 1. Example heterogeneous requirements.

Moreover, each department is not only using different languages but also using different formalism such as department from Malaysia is using Unified Modeling Language (UML), Gulf Countries are using MERISE Conceptual Model of Treatment (MCT), and German version of requirements formalism is a goal-oriented approach. The example showed in the Fig. 1 mainly focuses on the following:

- Integration of heterogeneous requirements from a different partner into one language which is understood by all partners.
- Reasoning on requirements for identification of relationship. i.e., in Fig. 1 all partners have the same need.

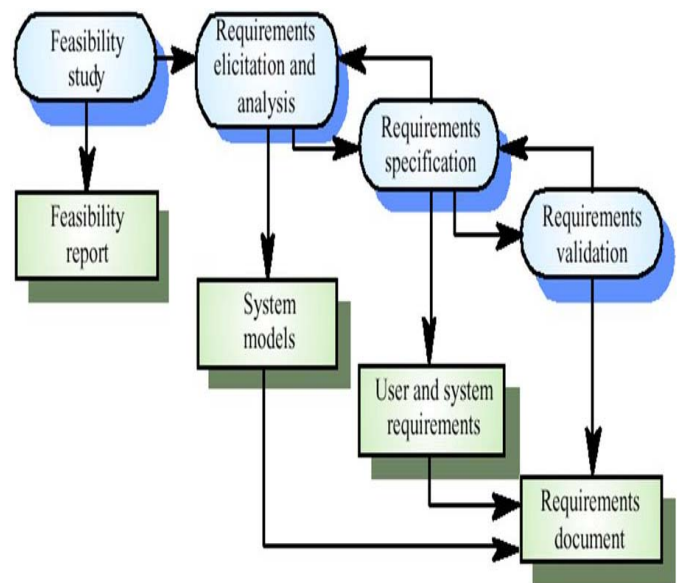


Fig. 2. Process of requirement engineering [3].

B. Ontology Engineering

This section presents ontology definition and how ontology can be used to store information captured from functional requirements.

A standard definition of ontology is coined by Gruber [13], "An ontology is an explicit, formal specification of a shared conceptualization." This definition emphasizes on two main points: formal conceptualization and automated reasoning, and designing the domain-specific ontology.

The word ontology has borrowed initially from philosophy within less than twenty years, and it means the philosophical study of nature of existence. In other words, ontology comprises recognizing the essential categories of things such as

ontology may be used to group objects as abstract or concrete, essential or existential. Although, term ontology is borrowed from philosophy though it gains substantial popularity in computer science and information science [14], [15].

Likewise, the Web Ontology Language (OWL)² is a language for modeling ontologies for the semantic web, and it is recommended standard proposed by the World Wide Web Consortium (W3C). OWL design is a more expressive language which defines the classes axiomatically and with the support consistency reasoning [16]. OWL is evolved from languages those were developed by joining two standards: European standard Ontology Interface Layer (OIL) and American standard DARPA³ Agent Markup Language (DAML). OWL allows specifying various ontology modeling paradigms such as hierarchical relationships, restrictions, modeling attributes, and associations under a well-defined semantic [17].

Numerous definitions have been proposed for ontology [13], [18], [19], domain ontology is considered as domain conceptualization of formal, consensual and referenceable regarding classes and properties. This definition focuses on three criteria that differentiate ontologies from other models used in computer science:

- **Formal:** In the context of requirements engineering, ontologies offer reasoning capabilities to diagnose inconsistency and incompleteness from requirements specification. The formal ontology conceptualization is grounded in a formal theory which checks the level of consistency and automated reasoning over the ontological concepts and individuals.
- **Consensual:** The consensual aspect allows the designer to share their models using global ontologies.
- **Referenceable:** It is the capability of ontology that each concept either class or property can be referred through a unique identifier. It helps in defining requirements semantics.

Another definition of ontology in computer science is coined by Gruber Gruber [20], "An ontology is a representation of specification of a conceptualization." Consequently, an ontology represents the conceptual model of the specific domain of interest, describing it in a declarative fashion [21].

C. *Ontology-driven Approaches Contribution in RE*

This section briefly presents state of the art studies of ontology-driven in RE. With the globalization of world in the case of the complex system, many technical experts from different fields, department, research lab and from a different part of the world participate in the project. Most of the time, they use their favorite languages to formalize the requirements of their assign part given to them [18], [22], [23].

To address the issues of heterogeneity in requirements, the study [18] proposed a pivot model to assess the user requirements role in data design repository. In the study, firstly, ontological concepts are presented relating to the formalisms of user requirement. Secondly, a proposed model is intended to integrate with different semi-formal models. For the validation of approach, a case study is an implementation using model-driven approach.

Similarly, another study [22] presents a multi-perspective framework to manage requirements traceability using ontology as a knowledge management mechanism. To generate traceability relation ontology matching is applied as a reasoning mechanism. The associations are recognized by originating semantic similarity of ontological concepts representing of requirements elements. The precision and recall are used to compare the results of traceability relations identified by the framework and manually identified by the user, as validation of approach.

Furthermore, ontology in the context of software requirements is used to store information derived from requirements. Ontology is a structured way to organize information and data stores in ontologies can be accessed via queries. Software requirements specification ontology helps in capturing domain knowledge and knowledge of software under development. As software requirements specifications documents are used in all stages of development, an ontology is developed from software requirements specifications can support different development activities during the development process. These ontologies can be used to present domain knowledge in a processable format that may be helpful in test case generation to support software testing process [24], [25].

Likewise, the ontologies can be used, to reduce the adverse effects of factors such as ambiguous statements, insufficient specification, and changing requirements on requirements engineering process. The possible application of ontologies in RE process: To develop the requirements model, a paradigmatic way to write requirements, and acquisition of domain knowledge in a structured way [8], [26]. Sommerville defined that requirements specifications document is a description of the desired software characteristics specified by the customers [3].

Similarly, ontologies can be used to reduce the barrier of understanding that if machines are not recognizing the knowledge, ontology formalized that knowledge in a computer and human-readable form. It allows the user to find information based on purpose rather than syntax. A significant issue is the definition of standards to represent the underlying structures, the ontologies [17], [27]. The Resource Description Framework (RDF)⁴ allows defining taxonomies and relations between concepts. The RDF has three object types: resources, properties, and statements.

² <https://www.w3.org/TR/owl-features/>

³ <https://en.wikipedia.org/wiki/DARPA>

⁴ <https://www.w3.org/TR/rdf-concepts/>

Requirement Models	Concepts		Pivot Model				Relationships
			Actor	Requirement			
				Action	Criteria	Result	
Goal-oriented	Actor		✓				
	Goal	Task		✓			
		Metric					
		Result					
	AND/OR						✓
	Combined Relationships						✓
Use case Model	Actor						✓
	Use case	Action	✓				
		Result					
		Extension points					
		Condition					
	Generalization						
	Include						
	Extend						
MERISE	Actor		✓				
	Event						
	Treatment	Action			✓		
		Result					
		Emission rules					
	Synchronization						✓

Fig. 3. The Core notations of three requirements formalisms [12], [18], [28].

IV. PROPOSED FRAMEWORK

In this study, we present a framework to combine different user requirements in MNC as shown in Fig. 4. This study is based on the use of ontologies to amalgamate the vocabularies used to software specification in different languages. Also, this study proposed a generic model to express the software requirements statements defined in different languages. At the final stage, framework checks the consistency of the integrated requirements. As shown in Fig. 3, the framework consists of subsequent phases: (1) Each partner from different countries defines its own Local Ontology (LO), and LO will be derived from Global Ontology (GO). These ontologies dealt with the heterogeneity of different languages vocabulary. (2) Different requirements formalism, are integrated using Pivot Model, (3) Local requirements transformation into Pivot Model.

A. Ontology as Shared vocabulary using Multilanguage WordNet

Domain ontology is defined as a formal, consensual, and referenceable dictionary for classes and properties of entities. The glossary in the framework highlights that objects or

association in the ontology domain is referred to language-independent identifier. To develop a global ontology that can coordinate with multiple languages and vocabularies, we suggest using Multilanguage WordNet. The approach relies on the following assumptions.

- A global ontology stores standards ontology such as, International Electrotechnical Commission⁵ (IEC) standard, ISO standards on the targeted domain by the application to be developed.
- WordNet is a sizeable English database, which helps to identify the part of speech tags such as nouns, verbs, adjectives, and adverbs. In framework Fig. 5, WordNet acts as lexical ontology. Multilanguage WordNet⁶ module is used to access different languages.
- Each designer develops an ontology in the local language, and global ontology is also inherited into the local ontology. Requirements defined in LO later exported regarding the GO.

B. Generic Requirements Formalism using Pivot Model

We combine different languages into formal requirements we apply pivot model. The designers are free to define his concepts in LO. Though each designer's formalism may be different; The analyst used three different DESIGNS such as use case of UML, goal oriented and MCT model of MERISE method. Fig. 3 shows the formalism identified by core notations used in the requirement statements defined as set of actions, results, criteria, and the relation between requirements. By merging the core notions and metamodeling, general requirements are formalized [18], [28].

A generic model to formally defines the Pivot model:

(Actors, Requirements, Relationships) where:

- The requirements can be defined as is a set of conditions specified by an actor. It can be described as {A, R, C, T}, where:
 - A is a set of activities to satisfy a requirement.
 - R is the results obtained if the requirements are satisfied.
 - C is set of criteria or conditions to quantified results.
 - T is the type of requirements.
- Relationships can be defined as is a set of relationships among requirement: Relationship = {Requires, constrains Refines, conflicts}

Formalization detail is adopted from study [18], [28].

C. Transformation of Formalized Requirements into Pivot Model

We used transformation model techniques [29], for mapping between local formalism such use case, goal, MCT Pivot model (Fig. 5).

⁵ <https://webstore.iec.ch/publication/21869>

⁶ <http://compling.hss.ntu.edu.sg/omw/>

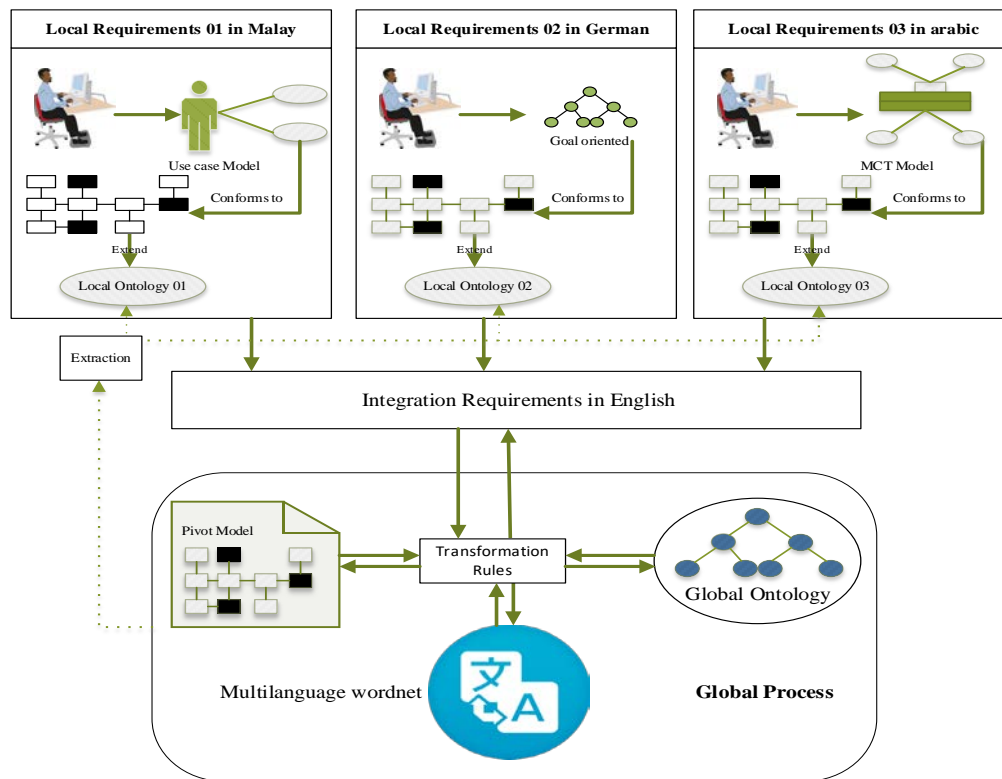


Fig. 4. Overview of proposed framework to integrate requirements.

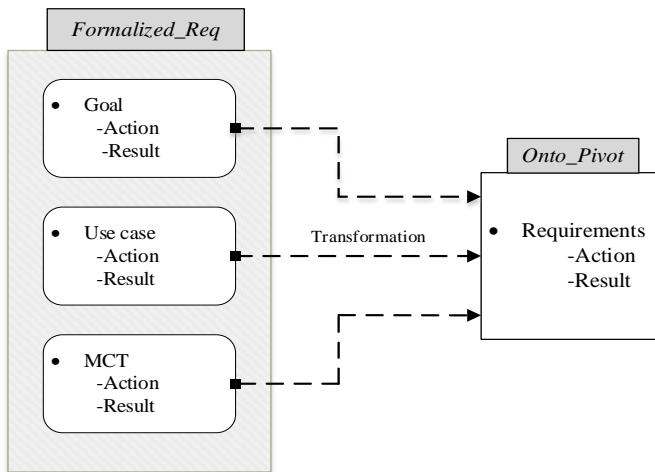


Fig. 5. Transformation of requirements.

As there are many tools and formats used to design specifications, we use ATL⁷ transformation language. ATL is a set of the model to model transformation which can integrate into Eclipse framework. It provides ways to produce a set of target models from a set of source models.

Requirements formalism is composed of three sub-formalisms as shown in Fig. 5. These instances are related to the local ontologies as described in the prior sections. The transformation rules are then implemented in these instances to translate them into Pivot model.

V. LIMITATIONS

This study does not conduct detail experiments on problem stated in Section 2. However, only presents a proposed framework to address the issue to integrate heterogenous requirement in the context of MNC.

VI. CONCLUSION AND FUTURE WORK

In this study, we discuss the problem that how heterogeneous requirements may be combined using local and global ontologies. The approach is based on general user requirement model that utilized three existing semi-formal languages: use cases of UML, MCT model of the MERISE method, and goal-oriented languages. This model is linked with local ontology and global ontology. These ontologies play the part of an amalgamated dictionary of Multilanguage Wordnet. Each designer from different locations develops the ontology using concepts and properties. By combining different user requirements in the context of MNC, our method focuses on minimizing the effect of heterogeneity of the quality of software product. In future, we will develop a system to integrate other languages.

ACKNOWLEDGMENT

The authors would like to express their deepest gratitude to Research Management Center (RMC), Universiti Teknologi Malaysia (UTM) and Ministry of Higher Education Malaysia (MOHE) for their financial support under Research University Grant Scheme (Vot number Q.J130000.2516.19H64).

⁷ <http://www.eclipse.org/atl/>

REFERENCES

- [1] B. Nuseibeh and S. Easterbrook, "Requirements engineering: a roadmap," in Proceedings of the Conference on the Future of Software Engineering, 2000, pp. 35-46.
- [2] S. Hastie and S. Wojewoda, "Standish group 2015 chaos report-q&a with Jennifer Lynch," Retrieved, vol. 1, p. 2016, 2015.
- [3] I. Sommerville, "Software Engineering, Boston, Massachusetts: Pearson Education," ed: Inc, 2011.
- [4] R. W. Selby, Software engineering: Barry W. Boehm's lifetime contributions to software development, management, and research vol. 69: John Wiley & Sons, 2007.
- [5] M. Younas, D. Jawawi, I. Ghani, and R. Kazmi, "Non-Functional Requirements Elicitation Guideline for Agile Methods," Journal of Telecommunication, Electronic and Computer Engineering (JTEC), vol. 9, pp. 137-142, 2017.
- [6] D. Gašević, N. Kaviani, and M. Milanović, "Ontologies and software engineering," in Handbook on Ontologies, ed: Springer, 2009, pp. 593-615.
- [7] J. Z. Pan, S. Staab, U. Almann, J. Ebert, and Y. Zhao, Ontology-driven software development: Springer Science & Business Media, 2012.
- [8] V. Castañeda, L. Ballejos, M. L. Calusco, and M. R. Galli, "The use of ontologies in requirements engineering," Global journal of researches in engineering, vol. 10, pp. 2-8, 2010.
- [9] K. Siegemund, E. J. Thomas, Y. Zhao, J. Pan, and U. Assmann, "Towards ontology-driven requirements engineering," in Workshop semantic web enabled software engineering at 10th international semantic web conference (ISWC), Bonn, 2011.
- [10] A. Mustafa, W. M. W. Kadir, and N. Ibrahim, "Automated Natural Language Requirements Analysis using General Architecture for Text Engineering (GATE) Framework," Journal of Telecommunication, Electronic and Computer Engineering (JTEC), vol. 9, pp. 97-101, 2017.
- [11] B. González-Baixauli, M. A. Laguna, and Y. Crespo, "Product lines, features, and MDD," in EWMT 2005 workshop, 2005.
- [12] I. Boukhari, S. Jean, I. Ait-Sadoune, and L. Bellatreche, "The role of user requirements in data repository design," International journal on software tools for technology transfer, pp. 1-16, 2016.
- [13] T. R. Gruber, "A translation approach to portable ontology specifications," Knowledge acquisition, vol. 5, pp. 199-220, 1993.
- [14] F. S. Parreiras, Semantic Web and model-driven engineering: John Wiley & Sons, 2012.
- [15] M. Hepp, "Ontologies: State of the art, business potential, and grand challenges," in Ontology Management, ed: Springer, 2008, pp. 3-22.
- [16] T. Diamantopoulos, M. Roth, A. Symeonidis, and E. Klein, "Software requirements as an application domain for natural language processing," Language Resources and Evaluation, vol. 51, pp. 495-524, 2017.
- [17] M. Ehrig, Ontology alignment: bridging the semantic gap vol. 4: Springer Science & Business Media, 2006.
- [18] I. Boukhari, L. Bellatreche, and S. Jean, "An ontological pivot model to interoperate heterogeneous user requirements," in International Symposium On Leveraging Applications of Formal Methods, Verification and Validation, 2012, pp. 344-358.
- [19] G. Pierra, "Context representation in domain ontologies and its use for semantic integration of data," in Journal on data semantics X, ed: Springer, 2008, pp. 174-211.
- [20] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing?," International journal of human-computer studies, vol. 43, pp. 907-928, 1995.
- [21] D. Sonntag, Ontologies and adaptivity in dialogue for question answering vol. 4: IOS Press, 2010.
- [22] N. Assawamekin, T. Sunetnanta, and C. Pluempitiwiriwajewj, "Ontology-based multiperspective requirements traceability framework," Knowledge and Information Systems, vol. 25, pp. 493-522, 2010.
- [23] S. J. Muhammad Irfan Marwat, Muhammad Arif Shah, Syed Zafar Ali Shah, "Towards optimization of software engineering ontologies," in Computer Applications and Information Systems (WCCAIS), 2014 World Congress on, 2014, pp. 1-6.
- [24] É. F. Souza, R. A. Falbo, and N. Vijaykumar, "Ontologies in software testing: a systematic," in VI Seminar on Ontology Research in Brazil, 2013, p. 71.
- [25] V. Tarasov, H. Tan, M. Ismail, A. Adlemo, and M. Johansson, "Application of inference rules to a software requirements ontology to generate software test cases," in International Experiences and Directions Workshop on OWL, 2016, pp. 82-94.
- [26] M. I. Marwat, S. Jan, M. A. Shah, and S. Z. A. Shah, "Towards optimization of software engineering ontologies," in Computer Applications and Information Systems (WCCAIS), 2014 World Congress on, 2014, pp. 1-6.
- [27] D. Brickley and R. V. Guha, "RDF vocabulary description language 1.0: RDF schema," 2004.
- [28] L. Bellatreche, N. X. Dung, G. Pierra, and D. Hondjack, "Contribution of ontology-based data modeling to automatic integration of electronic catalogues within engineering databases," Computers in Industry, vol. 57, pp. 711-724, 2006.
- [29] A. G. Kleppe, J. B. Warner, and W. Bast, MDA explained: the model driven architecture: practice and promise: Addison-Wesley Professional, 2003.