

# Evaluating the Effectiveness of Requirement Patterns for Data Intensive Application Development

**Renita Raymond**

School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India  
renita.r@vit.ac.in

**Margret Anuncia Savarimuthu**

School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India  
smargretanuncia@vit.ac.in (corresponding author)

Received: 12 April 2024 | Revised: 5 May 2024 | Accepted: 14 May 2024

Licensed under a CC-BY 4.0 license | Copyright (c) by the authors | DOI: <https://doi.org/10.48084/etasr.7469>

## ABSTRACT

In the rapidly evolving landscape of data-intensive applications, the precision and clarity of software requirements play a pivotal role in these applications' development process. This research focuses on the validation of a specifically designed Transformation Requirement Pattern (TFReqPat) for data-intensive applications, such as banking, e-commerce, and healthcare. The main objective is to ascertain the completeness, correctness, and unambiguity of requirements captured using this novel pattern. Traditional approaches to requirement documentation often suffer from inconsistencies leading to the exploration of structured patterns that promise enhanced clarity and reuse. This article focuses on acceptable verification and validation procedures deployed as concrete methods to validate the adequacy of the captured requirements. Through the generation of stringent acceptance criteria, it is ensured that the documented requirements were adherent to developmental standards with fewer ambiguities. As per the proposed validation process, the correctness and completeness of the specified requirements were determined through the acceptance criteria that check for the presence of process, domain dependency, behavior, and storage details in the specifications. Accordingly, the obtained evaluation score was 93.1%, 88.5%, and 75.2% for correctness and 94.8%, 92.9%, and 76.4%, respectively, for completeness. These findings significantly underscore the suitability of the proposed pattern for data-intensive applications, marking it as a more efficient and effective methodology compared to ad-hoc reuse approaches. This article not only contributes a validated requirement pattern to the field but also highlights the importance of structured requirement documentation in enhancing software development outcomes for data-intensive domains.

*Keywords*-transformation requirement pattern; requirement validation; acceptance criteria; correctness; completeness

## I. INTRODUCTION

Requirement validation is an essential step in ensuring that the requirements are complete and consistent according to the user needs. Acceptance criteria directly validate the requirements using requirement patterns. By executing tests that align with these patterns, the team can assess whether the software behaves as expected and fulfills the specified requirements. Test execution provides critical feedback on the effectiveness of the requirement patterns and also ensures that the software meets the user expectations, which are often encapsulated in the requirement patterns [1]. Therefore, the extracted requirements, namely Transformation Requirements (TR), i.e. requirements that transform the input into different

forms for fulfilling the objective of the application, Solution Requirements (SR), i.e. requirements about the generated outcomes, Stakeholder Requirements (STR), i.e. user requirements, and Business Requirements (BR), i.e. business objectives are subjected to validation [2]. The extracted requirements are considered as an input and the requirement sentences are classified as casual and non-casual. Furthermore, the confirmed casual sentences are labelled, and their relationships are established. It is inferred that if the specifications formed are unambiguous, clear, and consistent, their correctness is guaranteed through the creation of viable acceptance criteria. Accordingly, strong acceptance test cases are generated to reflect the requirements considered for the development [3–7]. To schedule the user acceptance criteria,

appropriate test cases are to be generated. According to the research work carried out, a pattern describing the characteristics of DIA in specific to transform the requirement is created [8]. The pattern is depicted in Figure 1. Consequently, a set of user acceptance test cases are also derived to ensure that the specifications represented through the pattern are reasonable and complementing for the acquired requirements.

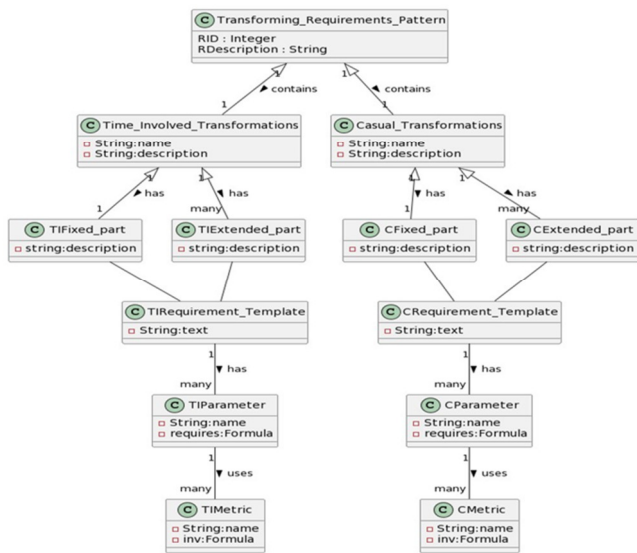


Fig. 1. Design of a TRReqPat.

The validation of a Transformation Requirement Pattern (TRReqPat) was carried out in two ways. The first is based on six experts' judgements. A questionnaire was designed in Google Forms and was given to the six experts. The questionnaire responses were quantified with a five-point Likert scale with the following values: Strongly disagree = 1, Disagree = 2, Neutral = 3, Agree = 4, and Strongly agree = 5 [8]. The questionnaire was divided into three sections:

- The pattern is relevant to the domain
- The pattern is valid with respect to data intensive systems.
- The pattern is valid with respect to requirement analysis.

The questionnaire was subjected to a reliability test to determine the level of internal consistency using Cronbach's alpha measure, which is a popular technique for assessing reliability, particularly in recent years [9]. Validating requirements for completeness and correctness is especially crucial in data-intensive applications due to their complex nature and the criticality of accurate data handling. Completeness ascertains that all necessary functionalities and conditions are defined to meet user needs and business objectives. For data-intensive applications, this involves detailing data sources, data volume, data processing, storage needs, and data security requirements. Correctness in requirements validation ensures that the requirements truly represent what the stakeholders want and are free from errors. This involves confirming that the data flows, data transformations, and integrations are accurately specified and align with business rules and regulations, especially in sectors like banking, healthcare, and e-commerce where data integrity and accuracy are paramount [8].

II. VERIFICATION FOR ACCEPTANCE

According to [2], the requirements are classified as TR, BR, SR, and STR. The extracted requirements are given as an input and the requirement sentences are categorized as casual and non-casual. After that, the confirmed casual sentences are labelled, and their relationships are established. Strong acceptance test cases are generated to reflect the requirements considered for the development. For example, a sample requirement of banking application is "When the red button is pushed or the power fails the interface does not accept further inputs and the system shuts down". The following Figures were drawn with the aid of CiRA [4]. Figures 2 and 3 indicate that the sample requirement is classified and a cause and effect graph is generated. Figure 4 illustrates the acceptance test cases of the sample requirement.



Fig. 2. Labeling of a sample requirement of a banking application.

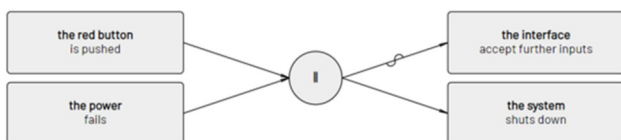


Fig. 3. Cause and effect graph of the banking sample requirement.

The sample requirement for a health care system is "The system should entail certain provisions for the administrator to be able to manage and maintain patient data records and search for patients when the patient is registered". Figures 5-7 portray the classification, cause and effect graph, and the acceptance

test cases of the sample requirement. Therefore, it is demonstrated that the captured requirements are considered for software development.

ID	Input		Output	
	the red button	the power	the interface	the system
1	not is pushed	not fails	accept further inputs	not shuts down
2	is pushed	not fails	not accept further inputs	shuts down
3	not is pushed	fails	not accept further inputs	shuts down

Fig. 4. Test cases of a banking sample requirement.

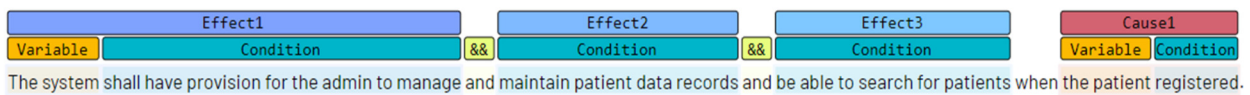


Fig. 5. Labeling of the health care sample requirement.

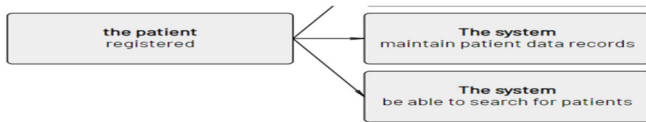


Fig. 6. Cause and effect graph of the healthcare sample requirement.

ID	Input	Output
1	the patient registered	The system shall have provision for the admin to manage and maintain patient data records and be able to search for patients
2	not registered	not shall have provision for the admin to manage and maintain patient data records and not be able to search for patients

Fig. 7. Acceptance test cases of the sample health care system.

### III. VALIDATION OF A TRANSFORMATION REQUIREMENT PATTERN

#### A. Through Experts' Opinions

Initially, the validation of a TFReqPat is based on the expert's judgement. A questionnaire was subjected to a reliability test to determine the level of internal consistency using Cronbach's alpha measure. According to [10], the correlation of variance and standard deviation can more accurately depict the Cronbach's alpha reliability coefficient than range. The greater the Cronbach's alpha score is, the more reliable the instrument will be designed to quantify the construct. The value of Cronbach's alpha should be above 0.7 to be reliable [9]. Table I displays the expert's judgement about the designed questionnaire. From Table II, as the Cronbach's alpha score is 0.82, it can be assumed that the pattern is an efficient and effective opinion. Figure 8 and Table III present the sample evaluation report and the designed questionnaire. It should be noted that most experts agreed with the survey form. Hence, the form is accepted for further evaluation.

TABLE III. SAMPLE EVALUATION REPORT

Questions	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Does reusing requirements in data-intensive systems improve the quality of requirements in it?				Yes	
Do you believe that transformation requirements play a significant role in providing business organizations insights compared to other requirement types?				Yes	
Are you of the viewpoint that a requirement pattern may aid in the creation of high-quality requirements?				Yes	

#### B. Through Requirement Quality Metrics

Requirement quality metrics are measures of how well the software requirements meet the desired standards and expectations of the stakeholders, such as clarity, completeness, consistency, traceability, testability, and feasibility [11–14].

- Completeness: the requirement should contain all the necessary information, including constraints.
- Correctness: refers to the absence of errors in the requirement statement.

Tables IV and V reveal the total score of the completeness and correctness metrics. The completeness metrics with respect to data intensive applications are:

- The requirement should have a description (D).
- It should have the functional factors (action verb) (AV).
- The description should delineate the abilities of a system (non-functional factor) (AB) (optional).
- The description should have input, process, and output either event driven or non-event driven (I/P +PR -> O/P) [precondition + event + expected result].

TABLE I. EXPERTS JUDGEMENT

E	N	SD	D	N	A	SA
E1	15	0	0	0	12	3
E2	15	0	0	2	10	3
E3	15	0	0	0	11	4
E4	15	0	0	1	12	2
E5	15	0	0	1	9	5
E6	15	0	0	0	11	4

E1 to E6: Expert id, N:No of Questions, SD: Strongly Disagree, D: Disagree, N: Neutral, A: Agree, SA: SA

TABLE II. RELIABILITY SCORE

Cronbach's alpha	Cronbach's alpha based on standardized items	No of Questions (N)
0.82	0.825	15

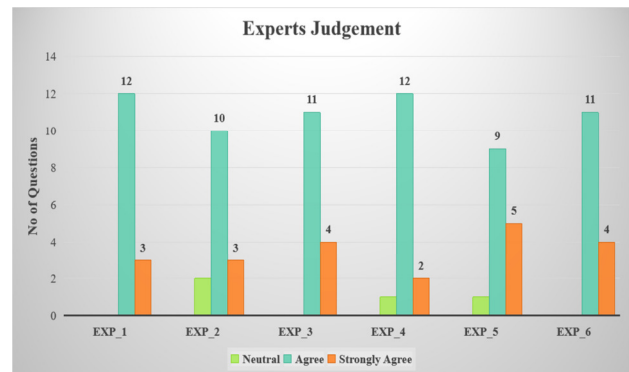


Fig. 8. Expert's judgement regarding the questionnaire.

- It should describe whether the requirement after processing stores in a permanent storage or temporary storage (PS/TS).
  - Functional factors mentioned in the requirement description should be domain dependent (DD).
  - The input and output sources should be valid (I/P, O/P).
  - The business rules of the organization should match the requirement description (BR).
- The correctness metrics with respect to data intensive applications are:
- An action verb describing the task (AV\*).

TABLE IV. COMPLETENESS SCORE

RID	Requirement	D	AV	AB	(I/P +PR -> O/P)	(PS/TS)	Total score
1	The system should provide the customers the ability to enter the payment details, to process the payment and store the payment confirmation message.	1	1	0	1	1	4
2	The system should provide the ability to automate the process of not identifying the suppliers when the stock levels are too low and store the notification message.	1	1	0	1	1	4
3	The system should offer customers and admin the chance to track the orders placed from one shipment location to the other and send the message to the customer.	1	1	0	1	1	4

TABLE V. CORRECTNESS SCORE

RID	Requirement	AV*	DD	I/P, O/P	BR	Total score
1	The system should provide the customers the ability to enter the payment details, to process the payment and store the payment confirmation message.	1	1	1	1	4
2	The system should provide the ability to automate the process of not identifying the suppliers when the stock levels are too low and store the notification message.	1	1	1	1	4
3	The system should offer customers and admin the chance to track the orders placed from one shipment location to the other and send the message to the customer.	1	1	1	0	3

TABLE VI. EVALUATION SCORE OF CAPTURED TRANSFORMATION REQUIREMENTS

Domain	T_TReq	T_CTReq	Correctness			Completeness		
			CE	PCE	LC	CM_E	CM_PCE	CM_LC
Banking	186	174	162	8	4	165	6	3
E-commerce	124	113	100	6	7	105	2	6
Health care	96	89	67	6	6	69	4	6

TABLE VII. EVALUATION SCORE IN PERCENTAGE

Domain	Correctness			Completeness		
	CE	PCE	LC	CM_E	CM_PCE	CM_LC
Banking	93.10	4.60	2.30	94.83	3.45	1.72
E-commerce	88.50	5.31	6.19	92.92	1.77	5.31
Health care	75.28	6.74	6.74	76.40	4.49	7.87

Total no of Transforming Requirements captured → T\_TReq  
 Total No of Correctly Captured Transforming Requirements through TFReqPat → T\_CTReq  
 Evaluation Score → ES\* (must be between 1 and 4)  
 Total No of Correctly Captured Transforming Requirements through TFReqPat with ES >=3 → CE  
 Total No of Correctly Captured Transforming Requirements through TFReqPat with ES >=2 → PCE  
 Total No of Correctly Captured Transforming Requirements through TFReqPat with ES <2 → LC  
 Total No of Completely Captured Transforming Requirements through TFReqPat with ES >=3 → CM\_E  
 Total No of Completely Captured Transforming Requirements through TFReqPat with ES >=2 → CM\_PCE  
 Total No of Completely Captured Transforming Requirements through TFReqPat with ES <2 → CM\_LC

According to [6], the sample transformation requirements given in Tables VI and VII, exhibit the evaluation score of the captured transformation requirements with respect to completeness and correctness. Figure 9 depicts the evaluation score of Table VII. As big data plays a major role in banking, e-commerce and health care, these applications are considered case studies [15-17]. As the approach is novel, the datasets

(requirements) are created as per the software engineering requirement specification guidelines - IEEE 830 [18]. There is no universal formula for calculating the completeness and correctness metric. However, the basic idea was captured from [19].

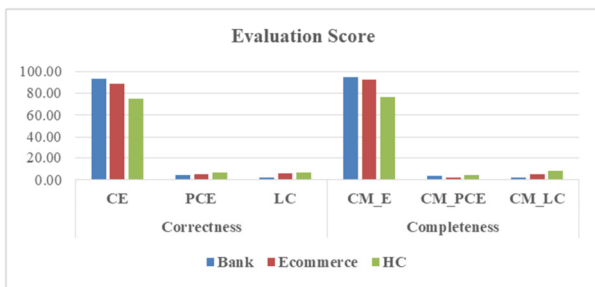


Fig. 9. Evaluation score of the captured requirements with respect to completeness and correctness.

#### IV. CONCLUSION

This article confirms the effectiveness of the novel pattern developed in order to capture the requirements of the data intensive application development. In an attempt to demonstrate the effectiveness of the proposed pattern, the validity of the specified requirements was subjected to acceptance criteria, which in turn confirmed the completeness and correctness of the specified requirement. Accordingly, the requirements were specified through the TFReqPat of the chosen domains banking, e-commerce, and health care and were evaluated for their validity through CiRA, an open source software for test case generation [4].

In addition, the requirements' validity is ensured through the acceptance criteria designed for a satisfactory requirement specification. The eminence of the specified requirement is determined by the correctness and completeness indicators. Accordingly, the outcome achieved for the considered domains of banking, e-commerce, and health care was 93.1%, 88.5%, and 75.2% for correctness and 94.8%, 92.9%, and 76.4%, respectively, for completeness. The pattern is designed based upon the generalized characteristics of the data intensive application requirements and so it can be extended, and is applicable for any processing requirements. For handling requirements associated with input and output, light modifications may have to be considered, in accordance with the characteristics of the application and domain.

#### REFERENCES

- [1] C. Wang, F. Pastore, A. Goknil, and L. C. Briand, "Automatic Generation of Acceptance Test Cases From Use Case Specifications: An NLP-Based Approach," *IEEE Transactions on Software Engineering*, vol. 48, no. 02, pp. 585–616, Feb. 2022, <https://doi.org/10.1109/TSE.2020.2998503>.
- [2] R. Raymond and S. M. Anuncia, "Identification of Data-Intensive Systems Requirements using Semantic Similarity Search," *Journal of Engineering Science and Technology Review*, vol. 15, no. 2, pp. 215–227, 2022, <https://doi.org/10.25103/jestr.152.25>.
- [3] J. Fischbach, "Why and How to Extract Conditional Statements From Natural Language Requirements," Ph.D. dissertation, University of Cologne, Koln, Germany, 2022.
- [4] J. Frattini, J. Fischbach, and A. Bauer, "CiRA: An Open-Source Python Package for Automated Generation of Test Case Descriptions from Natural Language Requirements," in *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, Hannover, Germany, Sep. 2023, pp. 68–71, <https://doi.org/10.1109/REW57809.2023.00019>.
- [5] J. Fischbach and A. Vogelsang, "Conditional Statements in Requirements Artifacts: Logical Interpretation, Use Cases for Automated Software Engineering, and Fine-Grained Extraction," in *Ernst Denert Award for Software Engineering 2022: Practice Meets Foundations*, E. Bodden, M. Felderer, W. Hasselbring, P. Herber, H. Koziolok, C. Lilienthal, F. Matthes, L. Prechelt, B. Rumpe, and I. Schaefer, Eds. Cham, Switzerland: Springer Nature Switzerland, 2024, pp. 9–43.
- [6] J. Fischbach *et al.*, "Automatic creation of acceptance tests by extracting conditionals from requirements: NLP approach and case study," *Journal of Systems and Software*, vol. 197, Mar. 2023, Art. no. 111549, <https://doi.org/10.1016/j.jss.2022.111549>.
- [7] S. Hotomski and M. Glinz, "GuideGen: An approach for keeping requirements and acceptance tests aligned via automatically generated guidance," *Information and Software Technology*, vol. 110, pp. 17–38, Jun. 2019, <https://doi.org/10.1016/j.infsof.2019.01.011>.
- [8] M. A. Savarimuthu and R. Raymond, "Transformation Requirement Pattern for Capturing Data-Intensive Applications Requirements," *Journal of Internet Services and Information Security*, vol. 12, no. 4, pp. 126–138, Nov. 2022, <https://doi.org/10.58346/JISIS.2022.14.009>.
- [9] I. Kennedy, "Sample Size Determination in Test-Retest and Cronbach Alpha Reliability Estimates," *Middle East Research Journal of Humanities and Social Sciences*, vol. 2, no. 1, pp. 17–29, Dec. 2021, <https://doi.org/10.36348/merjhss.2021.v01i01.003>.
- [10] M. Amirrudin, K. Nasution, and S. Supahar, "Effect of Variability on Cronbach Alpha Reliability in Research Practice," *Jurnal Matematika, Statistika dan Komputasi*, vol. 17, no. 2, pp. 223–230, Dec. 2020, <https://doi.org/10.20956/jmsk.v17i2.11655>.
- [11] E. Sarmiento, J. C. Sampaio do Prado Leite, and E. Almentero, "Using correctness, consistency, and completeness patterns for automated scenarios verification," in *2015 IEEE Fifth International Workshop on Requirements Patterns (RePa)*, Ottawa, ON, Canada, Dec. 2015, pp. 47–54, <https://doi.org/10.1109/RePa.2015.7407737>.
- [12] C. Kongkaew, C. N. Scholfield, T. Supapaan, C. Mann, P. Mongkhon, and S. Chanunon, "Impact of research-based learning on student knowledge and assessment in Pharmacoepidemiology: a one group pretest-posttest experimental study," *Thai Journal of Pharmaceutical Sciences*, vol. 43, no. 4, Jan. 2020.
- [13] D. Zowghi and V. Gervasi, "The Three Cs of Requirements: Consistency, Completeness, and Correctness," in *Proceedings of 8th International Workshop on Requirements Engineering: Foundation for Software Quality, (REFSQ'02)*, Essen, Germany, Apr. 2003.
- [14] D. Zowghi and V. Gervasi, "On the interplay between consistency, completeness, and correctness in requirements evolution," *Information and Software Technology*, vol. 45, no. 14, pp. 993–1009, Nov. 2003, [https://doi.org/10.1016/S0950-5849\(03\)00100-9](https://doi.org/10.1016/S0950-5849(03)00100-9).
- [15] E. Erturk and K. Jyoti, "Perspectives on a Big Data Application: What Database Engineers and IT Students Need to Know," *Engineering, Technology & Applied Science Research*, vol. 5, no. 2, pp. 850–853, Oct. 2015, <https://doi.org/10.48084/etasr.592>.
- [16] A. Alghamdi, T. Alsubait, A. Baz, and H. Alhakami, "Healthcare Analytics: A Comprehensive Review," *Engineering, Technology & Applied Science Research*, vol. 11, no. 1, pp. 6650–6655, Feb. 2021, <https://doi.org/10.48084/etasr.3965>.
- [17] Z. U. Rehman and F. A. Shaikh, "Critical Factors Influencing the Behavioral Intention of Consumers towards Mobile Banking in Malaysia," *Engineering, Technology & Applied Science Research*, vol. 10, no. 1, pp. 5265–5269, Feb. 2020, <https://doi.org/10.48084/etasr.3320>.
- [18] A. Chikh and M. Aldayel, "Reengineering Requirements Specification Based on IEEE 830 Standard and Traceability," in *Advances in Intelligent Systems and Computing*, vol. 275, A. Rocha, A. M. Correia, F. Tan, and K. A. Stroetmann, Eds. 2014, pp. 211–227, [https://doi.org/10.1007/978-3-319-05951-8\\_21](https://doi.org/10.1007/978-3-319-05951-8_21).
- [19] R. J. Halligan, "Requirements Quality Metrics: The Basis of Informed Requirements Engineering Management," presented at the 1993 Complex Systems Engineering Synthesis and Assessment Technology Workshop (CESAW '93), Calvados, MD, USA, 1993.
- [20] S. Withall, *Software Requirement Patterns*, 1st ed. Redmond, WA, USA: Microsoft Press, 2007.